Scheduling Algorithms and Scalability Issues in Cloud Computing: A Survey

Dr. V. Muthumanikandan
Assistant Professor(Sr.)
School of Computer Science and Engineering
Vellore Institute of Technology, Chennai, Tamilnadu, India -600127

Abstract **-** Distributed computing manages various types of virtualized assets, subsequently planning places a significant job in distributed computing. In cloud, client may utilize several thousand virtualized assets for every client task. Thus manual planning is anything but a plausible arrangement. Strategic planning to a cloud situation empowers the utilization of different cloud administrations to help system execution. Cloud Computing is made available as pay on demand service to the clients. Cloud services are "pay-per-use" over the internet. It has many features that include measured services, availability, security and scalability. In this paper, a survey of cloud computing is presented, highlighting some of its issues and challenges and important concepts of scalability along with scheduling algorithms used in cloud. The aim is to create a better understanding of the immediate concerns that need to be taken care for a better scalable cloud environment in cloud computing.

Keywords **-** Cloud Computing, Scalability, Scheduling, Vertical Scalability, Horizontal Scalability

1. INTRODUCTION

Every on in the IT Industry has an opinion on what is cloud computing. Cloud computing helps in to increase the speed with which applications are deployed, increase innovation and lower costs, all which increasing business agility. Cloud computing has become the ability to use application on the web that store and protect data while providing a service –anything including email, sales department and tax preparation. Virtualization is a key feature of cloud computing, IT organizations have understood for years that virtualization allows them to easily create copies of existing environments and sometimes involving multiple virtual machines-to support test, development and staging activities.[2] [3]

Cloud computing offers three main delivery models which are Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS). Clouds in cloud computing is of several types based on the scalability and pooling up of the resources .Types are public, private, and hybrid clouds. Public clouds are availed to the general public in a pay-as-you-go manner and they are owned by the cloud provider. Private clouds are operated only for a business or an organization and they are controlled by that organization.

2. SCHEDULING ALGORITHMS

The following scheduling algorithms are established in the area of grids, clouds and workflows and these algorithms have been summarized in table with the scheduling parameters.

2.1 An Energy Efficient Scheduling Algorithm based on Private Cloud

Hybrid energy efficient scheduling application[1] based on pre- power techniques and least load first algorithm developed. Since private clouds have some coptional characteristics and special requirements, it is still a challenging problem to effectively schedule virtual machine requests onto computer nodes, especially with multiple objectives to meet. Special tow problems of virtual machine scheduling are discussed. Pre power technique is used to reduce the response time and it uses idle threshold value. Least load first algorithm is used to balance workloads when the data centers are running on power mode.

2.2 A Scheduling algorithm for private cloud

A hybrid energy efficient scheduling algorithm was proposed using dynamic migration.This paper is based on[1], but the disadvantage found is that by using the threshold value, powering down a busy node is not possible/feasible. Hence, a normal range set for the left limit is used. It uses power up command/instruction, to wake the hubs(sleep nodes) also the idle nodes.

2.3 Energy Efficient in Cloud Computing Environment

Proposed[10] a near optimal scheduling policies that exploits heterogenic across multiple data centers for a cloud provider. A number of efficiency factors like energy costs, carbon emission rate, workload and CPU power load efficiency which changes across different data center counting on their location agricultural design, and management system were considered.

2.4 Providing Power- Aware Cloud Resources for Real Time Services

This paper explore power-aware provisioning of virtual machines for real-time services[4].Energy consumption in a data center is a critical issue in cloud computing .Three Power-aware VM provisioning schemes proposed. Lower-DVS, Advanced-DVS, and Adaptive-DVS.A real time cloud service framework where each real time service request is modeled as RT-VM in resource broken have developed. This proposed approach is

- To model a real-time service as a real-time virtual machine request.
- To provision virtual machines of data centers using DVFS schemes.

3. STATIC/DYNAMIC

In static booking, all planning data about undertakings is accessible previously so the execution calendar of each undertaking is figured before executing any errand. It is successful for applications that have fixed requests. In addition, in static planning, the purchaser settles on concurrence with the cloud supplier for administrations and the cloud supplier readies the required assets before the beginning of required assistance. So the execution schedule of task may change as per the user demands[12]. Space-Shared approach is utilized for both VM booking also, task booking. Since each VM requires two centers, only one VM can be doled out profoundly in a explicit time. So VM2 can't run and utilize the centers until VM1 wraps up. Likewise, each undertaking facilitated inside the VM requires one center, so T1 and T2 will run all the while T3 and T4 will hold up until T1 and T2 are finished. The equivalent occurs for undertakings running in VM2. Fig.1 describes this case.

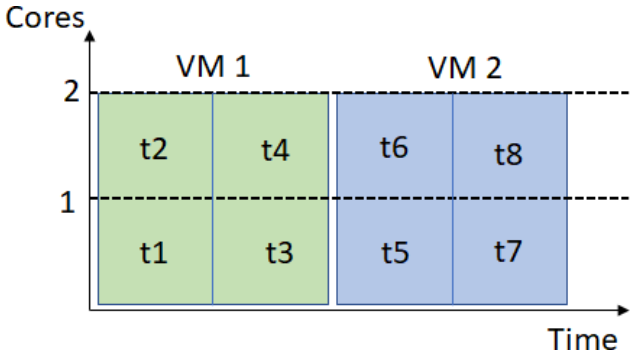

Fig. 1 Space Shared for VM and Tasks

Time-shared arrangement is utilized for VM planning, however space-shared arrangement is utilized for task planning. Thus, VM1 and VM2 shares a period cut of each center. At that point each cut will be allocated just one undertaking while others will hold up until those undertakings are finished.Fig.2 presents this case.

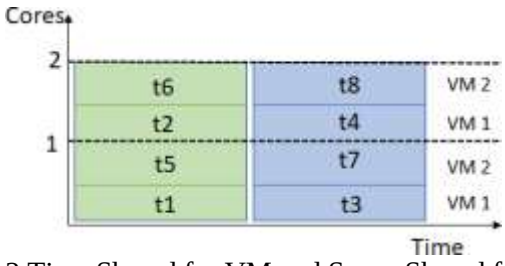

Fig. 2 Time Shared for VM and Space Shared for Tasks

Errand booking centers around mapping undertakings to suitable VMs proficiently. In view of the undertaking reliance, errands can be delegated free or ward undertakings. The free errands have no conditions with different undertakings and furthermore, have no need request should be followed during booking process. In any case, the needy undertakings have priority request dependent on conditions among the undertakings what's more, should be followed during the planning procedure. Booking subordinate undertakings is called Workflow Scheduling.[13]. Fig. 3 presents this case

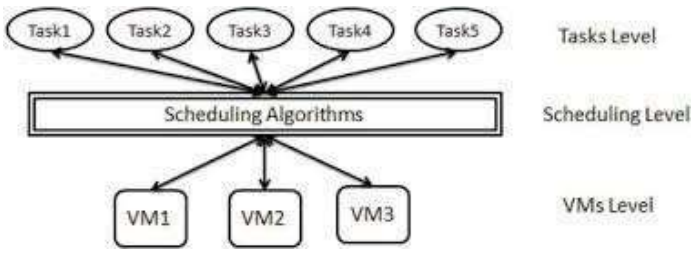

Fig. 3 Task Scheduling Algorithms

3.1 Scheduling of Scientific workflows employing a Chaos genetic algorithm

A Meta heuristic algorithm [6]based on Genetic Algorithms proposed. In a grid environment, number of challenges are

- Sources are shared(competition)
- Scheduler is not in control of resources
- Number of available resources are constantly changing and so on.

By using the characteristic of chaotic variable in scattering the solutions among the whole search space and thus avoids the precipitate convergence of the solution and produces better results within a shorter time. Investigation of scheduling workflows considering the QoS constraints(user budget, deadline)has done.

3.2 An Ant Colony Algorithm for Balanced Job Scheduling in Grids

Here proposed a balanced ANT colony algorithm [7]which uses pseudo random proportional rule to balance the whole system load while completing all the job roles at hand as soon as possible according to the environment status. Current scientific problems are very much complicated and require huge computing power and storage space. To utilize the grid resources efficiently ,balanced and colony algorithm is proposed by balancing the workload as well as minimizing the make span.

3.3 Job Scheduling Algorithm Based on Dynamic Management of Resource Provided by Grid Computing Systems

An algorithm of job scheduling and Dynamic adjustment of nodes loading with a grid system proposed[8]. Within a distributed computing systems, request of proposing randomly received from the system's users. A good planning of these request assumes their assigning towards available processors, so that all request should be solved as soon as possible. Considering the resources sharing in grid systems, a job scheduling algorithm which also includes dynamic load balancing is proposed. The distribution of first come, first served(FCFS) with a robin mechanism of the execution node is proposed.

3.4 Evolution of Gang Scheduling Performance and Cost during a Cloud Computing System

An efficient job scheduling algorithm[9] for time sharing proposed. This paper includes the study performance of a distributed cloud computing model, based on Amazon Elastic Compute Cloud(EC) architecture and to revise, study and estimate both the performance and the overall cost of two foremost gang scheduling algorithms. It utilizes the concepts of virtual machines act as the computational units of the system. The proposed system implement for adding and removing virtual machines from the system depending on the system load at any specific time. Job routing, Job scheduling has done, adaptive FCFS and largest job first served are used.

3.5 Heterogeneity aware resource Selection &Scheduling in the Cloud

A metric of share in a heterogeneous cluster to realize a Scheduling scheme that achieves high performance and fairness proposed[15]. The heterogeneity of the environment should be developed alongside performance and cost-effectiveness. The data analysis must report for heterogeneity of the situation and workloads. It must provide fairness among jobs when multiple jobs share the cluster. Hence architecture to allocate assets(resources) to a data analytics cluster within the cloud proposed.

3.6 Embracing market arranged Scheduling approaches for Cloud Computing

By considering the 2 degrees of provisioning(time and cost) two approachs were proposed[11]. It utilizes least expensive as set type which is known as small computational unit . The paper deals with how scheduling policies inside the middleman(broker) can take the advantage from resource supplied by IaaS providers additionally to the local schedulers to urge the use of application completed by the requested deadline and provided budget.

4. ACO BASED SCHEDULING ALGORITHMS

Ant Colony Optimization(ACO) metaheuristic is propelled by the conduct of genuine ants finding the most brief way between their provinces and a wellspring of food. postulation and was initially called subterranean insect framework. While strolling in the midst of their province and the food source, ants leave pheromones on the manners in which they move. The pheromone force on the sections increments with the quantity of ants going through and drops with the vanishing of pheromone. As time passes, small ways draw more pheromone and this way, pheromone force causes ants to perceive littler ways to the food source. ACO techniques are valuable for taking care of discrete advancement issues that need to discover ways to objectives. It has been effectively applied for taking care of voyaging sales rep issue, multidimensional rucksack issue, work shop booking, quadratic task issue, planning of errands in matrix and cloud and some more. The initial move toward any difficult arrangement utilizing ACO is to outline framework to the given issue.

5. GA BASED SCHEDULING ALGORITHMS

GA represents a populace put together streamlining technique based with respect to an analogy of the development procedure saw in nature[13]. In GA, every chromosome (individual in the populace) speaks to a potential answer for an problem and is made out of constant qualities. The underlying populace is taken arbitrarily to fill in as the beginning stage for the calculation. A fine work is characterized to check the appropriateness of the chromosome for the earth. Based on wellness esteem, chromosomes are chosen and hybrid and transformation tasks are performed on them to create off springs for the new populace. The wellness work assesses the nature of every posterity. The procedure is rehashed until adequate posterity are made.

A large portion of the booking calculations have focused on a couple of destinations, while proposed a resistant hereditary calculation for work process planning, which considered five goals and tackled imperative fulfillment issue related with task booking limitations. In the wake of utilizing twofold point hybrid and transformation, each arrangement that damaged the limitations was inoculated, along these lines remedying the inadequate qualities. Wellness work is determined by first isolating the destinations, which are to be limited and the targets to be amplified and afterward adding them after standardization.

6. PSO BASED SCHEDULING ALGORITHMS

Molecule Swarm Optimization (PSO) is a developmental computational method presented by Kennedy and Eberhart in 1995 spurred by social conduct of the particles. Every molecule is aligned with position and speed and travels through a multi-

dimensional inquiry space. In every cycle, every molecule alters its speed dependent on its best position and the situation of the best particle of the whole populace. PSO consolidates nearby hunt strategies with worldwide pursuit techniques attempting to adjust investigation and misuse. PSO has picked up prevalence because of its straightforwardness and its handiness in expansive scope of utilizations with low computational expense.

PSO was initially created for ceaseless enhancement issues. So it ought to be reengineered to take care of discrete enhancement issues, for example, booking. Little Position Value (SPV) rule is one of the colossally utilized procedures for this reason while utilizing 1 × n vector encoding for PSO particles. In , Integer- PSO procedure is utilized which outflanks the SPV when there is enormous distinction in the length of the undertakings and in the handling speed of assets.

6.1 Improved Cost-Based Algorithm for Task Scheduling

Proposed associate degree improved cost-based programming algorithmic rule[9] for creating economical mapping of tasks to offered resources in cloud. The improvisation of ancient activity based mostly cost accounting is projected by new task programming strategy for cloud surroundings wherever there is also no relation between the overhead application base and also the approach that completely different tasks cause overhead value of resources in cloud. This programming algorithmic rule divides all user tasks looking on priority of every task into 3 completely different lists. This programming algorithmic rule measures each resource value and computation performance, it conjointly Improves the computation /communication magnitude relation.

6.2 Performance and Cost evaluation of Gang Scheduling

Distributed ADPS with Job Migrations and Starvation Taking care of, projected a constabulary booking calculation with work movement and starvation taking care of during which designing equal occupations, effectively applied within the regions of Grid and Cluster process[12]. the amount of Virtual Machines(VMs) accessible at any second is dynamic and scales as indicated by the requests of the employments being overhauled. The antecedently mentioned model is focused through recreation thus on investigate the exhibition and usually speaking expense of Gang programming with relocations and starvation coping with. Results feature that this booking system may be adequately sent on Clouds, which cloud stages may be appropriate for HPC or superior endeavor applications.

6.3 An Optimistic Differentiated Job Scheduling System in Cloud computing

Proposed[11] associate degree isolated transcription count with non-preemptive would like lining model for practices performed by cloud client within the cloud calculation condition. during this procedure one internet application is formed to try to to some activity like one in all the report moving likewise, downloading then there's would like of helpful occupation transcription count. The Qos requirements of the distributed registering client and also the most outrageous benefits of the disseminated calculation organization supplier area unit cultivated with this estimation. Dynamic Resource Assignment on the premise of credibleness resource providing agent, whereas the user can assign applicable quantity of resource through resource demand agent to the

roles required to be done. In CDA mechanism, resource providing agent, resource demand agent and data serving agent correspond to the vendor, the client and also the arbiter within the auction severally. The arbiter is chargeable for organizing the auction and assembling market info.

At any unit of time throughout the auction, the vendor and also the customer supply their own value to the arbiter, and also the arbiter can match the resource transactions supported each side' tariffs and provides a median value for each sides.

6.4 Optimized Genetic Algorithm

Hereditary algorithmic rule (GA) what is additional, worldwide arrangement house search area unit the 2 outstanding supported Map/Reduce model in cloud, thus on fell each the all out period and also the traditional time of trip execution, scientists add an added well-being to boost the GA. that's the upgraded hereditary calculation with double well-being (DFGA), that has 2 well- being capacities. DFGA calculation utilizes the backhanded cryptography strategy for asset—task. The length of body is that the amount of sub errands. moreover, the estimation of each quality on the body is about the quality range that is appointed to the sub task on this space. Instatement is to provide a SCALE range of body, that features a length of M, and also the value scope of quality is associate degree irregular range in [WORKER]. Among them, M represents the entire range of sub assignments, and employee is that the amount of assets. There area unit 2 well-being capacities, one is that the all out time of occupation running on each single virtual machine and also the alternative is that the traditional time.

6.5 *A molecule swarm enhancement based heuristic for planning work process applications in distributed computing frameworks*

Meta heuristics technique dependent on molecule swarm enhancement proposed[10] . In framework condition, client applications may cause huge measure of information recovery and execution costs when they are booked considering just the execution time. Added to that improving the execution time, the expense emerging from information move between assets just as execution costs should likewise be considered, and centering to limit the all out execution cost of utilizations on assets. PSO's capacity to discover close to ideal answers for mapping all the errands in the work process to the given arrangement of PC assets. It considers both calculation and correspondence cost, if the asset cost builds PSO limits the most extreme complete expense of relegating all assignments to assets.

6.6 *Workflow Algorithms in Cloud Environment*

A review of different work process booking calculations has done[15]. Past work process booking calculations doesn't consider unwavering quality and accessibility factors. Accordingly required a work process booking calculation that can improve accessibility and dependability in cloud condition. The principle motivation behind a work process the board framework (WfMS) is to help the definition, execution, enlistment and control of business forms. Three significant segments in a work process establishment motor are the work process booking, information development and deficiency the executives. Because of the diminished exhibition looked in matrices, there is a need to actualize work processes in cloud.

6.7 Cloud-DLS :Dynamic trusted Scheduling for Cloud Computing

A trust dynamic level scheduling algorithm[14] named Cloud-DLS proposed. Because of the characteristic of cloud computing, obtaining trustworthiness in computing resources is difficult. Novel Bayesian method based cognitive trust model, trust relationship models of sociology is being used. The paper provides focus on Cloud computing.

6.8 Improved Ant Colony Algorithm

The embodiment of employment booking is to settle on a way of dynamic mixture of plus with moderately nice execution among all the plus portion techniques[7][8]. From the purpose of read of essential thinking, increased insect state calculation is entirely applicable for plus designation in cloud atmosphere. because the whimsicality of insect state calculation is big, it's effortlessly caught in near ideal arrangement and moderate combination. during this manner, explore laborers gift GA, that contains a capability of fast moreover, capricious worldwide inquiry, to every emphasis procedure of insect state calculation. this could hugely quicken the speed of assembly and guarantee the exactitude of the primary calculation. for each plus requester, distributed computing administration cluster ought to provides a genuinely good mixture of undertakings and assets. In improved subterranean insect settlement calculation, at the time, the weather influencing the plus state is depicted by scenario, and also the booking procedure will get unstartling outcomes primarily and speedily. In cloud state of affairs, take ACS (Ant Colony System) calculation model addicted to ACO calculation as an example, the progression of employment booking method addicted to ACO the improved ACO calculation addicted to associate broad cloud registering reenactment stage. it had been contrasted and also the Cooperative effort (RR) calculation and also the initial ACO calculation. For the foremost half, improved ACO calculation takes less time and contains a higher effectiveness than alternative 2 calculations.

7. SCALABILITY

With cloud computing, the calculated resources can be changed according to the fluctuating requirements of the client, thus avoiding underutilization and excessive use of resources while maintaining the high quality of the hosted service. This function is called elasticity and is the basis of the usage calculation model. Therefore, customers only pay when infrastructure resources are needed. Cloud computing is also useful from the perspective of a cloud provider, because more users can be served with the same service. Tools that automatically provide changes and modifies the amount of used resources are called "auto-scaling services" .Although auto-scaling has shown considerable potential for cloud computing, it also brings unique challenges that need to be addressed

Lack of research for automated scaling at the service level. Automatic scaling includes various cloud service models. However, most studies only focus on the level of infrastructure. Automatic scaling at the service level is important because the service runs on a number of connected VMs and the quality of the service depends on how the automatic scaling processes resources for that VM. Service level metrics such as For example, transactions in time units must be mapped to system level metrics such as CPU usage, network speed, and disk I / O. Insufficient tools for monitoring and aggregating metrics at the platform level and service level to support auto-scaling decisions. There is a lack of knowledge to show the relationship between auto-scaling and

quality attributes involving security availability, reliability. For example, DoS attacks causes an auto-scaling service to scale out the system unnecessarily and thus increase operation cost.

7.1 Cloud Scalability

One main advantage of using the cloud computing paradigm is its scalability. This supports long-term strategy and business requirements and differs significantly from resilience. This is a mechanism where customers dynamically mobilize their resources such as hardware and software applications when need and situations arise.. Cloud computing allows clients or cloud vendors business to easily scale up or scale down their IT requirements as and when required. This will allow clients to support their business growth without expensive changes to the existing systems. Auto scaling reduces the eclient's manual involvement and an intervention thus minimizes the possibility of client's errors, provides aautomation to the resources, increases the speed and reduces the laborer costs. Provisioning the resources automatically by implementing Auto scaling mechanism making the cloud first choice for the different e-commerce organizations residing on it.

7.2 Cloud Scalability Types

7.2 1 Horizontal Scalability (Scaling out)

Horizontal scaling is the process of involving more instances and resources to an application, service or system. For instance, a software as a service vendor who includes instances whenever average of users per cpu exceeds 50 and excludes instances when users per cpu falls to or below 40.

7.2.2 Vertical Scaling

The process of moving to huge instance or upgrading resources. For example, a webpage getting executed on a 8-CPU virtual machine that is deployed again to a 16-cpu machine. Many IaaS platforms have tools and technologies that makes the process easy that can be accomplished in mins or secs.

7.2.3 Auto Scaling

Scaling is performed using an API automatically. For instance, a settlement process for a bank that horizontally scales itself depending on how many trades it is processing.

7.2.4 Side -by -side scaling

It is the process of adding instances for different purpose on demand. For example, a firm that adds development and test instances of a service as required by the project[16].

7.2.5 Global Scaling

Scaling an infrastructure(service) to execute and run in different geographical locations. For instance, a content delivery network that delivers videos from different geographically distributed data centers so that videos are served to customers from a data center that is close to them[17].

7.2.6 Proactive scalability

This includes a schedule for infrastructure changes based on an estimated demand diagram. This scalability can be configured this way by cloud management tools. To meet the needs in the morning with the minimum infrastructure available and then to reduce capacity again until noon, this strategy is not intended to increase demand, but is based on certain programs.

7.2.7 Reactive scalability

In this strategy, the infrastructure reacts according to the changes in demand by adding or reducing capacity.

8. BENEFITS OF CLOUD SCALABILITY

8.1 Performance:

One main advantage of scalability in the cloud is that it improves performance. Scalable architecture has the ability to handle the traffic bursts and heavy workloads that will happen with business growth.

8.2 Cost-efficient:

You can allow grow your business without making costly changes in the current setup. This deducts the cost implications of large storage making cloud's scalability, cost effective.

8.3 Easy and Quick:

in cloud Scaling up or scaling out is simple; you can include additional VMs with a few clicks and after the payment process, the additional resources are made available without any delay.

8.4 Capacity:

Scalability makes sure that with the continuous business growth the storage space in cloud expands. Scalable cloud computing systems provides data growth requirements[18]. With scalability, you don't have to worry about storage size.

8.5 Flexibility:

The use of Cloud computing helps users to be more flexible – both in and out of the workplace. Employees can access folders using web-enabled notebooks, smartphones, laptops. The ability to share documents and other folders in the Internet simultaneously can also support both internal and external collaboration. Many employers are now using "bring your own device (BYOD)" policies. In this way, cloud enables us to use mobile technology.

8.6 Impact of Scalability on managed data centers

Due to highly scalable nature , many organizations are now managing with using managed data centers where there are cloud experts trained in maintaining and scaling shared, private and hybrid clouds[19]. Cloud computing helps in easy and quick allocation of resources in a monitored environment where overloading is never a concern as long as the system is managed properly. From small companies to large enterprise companies, managed data centers are often an option for your business.

8.7 Scalabillity Levels

Scalability is one major benefit of the cloud paradigm[20]. It differentiates clouds from advanced outsourcing solutions. But, some unresolved issues must be addressed before automated scaling of applications is done. Some core initiatives towards scalability in cloud environments are as follows

8.8 Server Scalability

Infrastructure as a Service (IaaS) clouds which are most available, collaborate and work with individual Virtual Machine (VM) management primitives—such as elements for adding or removing VMs—but lack mechanisms for treating applications as single entity or for managing relationships among application modules[21]. For instance, relationships between any VM is not to be concerned about, ordered deployment of VMs containing software for different application tiers deployment time, so the database is deployed firstly to receive IP and the web server details to configure and connect it. Application providers takes care of only applications, and not virtual infrastructure terms.

8.9 Scaling of the network

Networking over virtualized resources is done in two ways: overlay networks and TCP/IP virtualization and Ethernet virtualization. User traffic separation is not appropriate for application scalability: Scalability is often achieved by over-provisioning resources to meet this demand increase.

8.10 Scaling of the platform

Clouds like IaaS provide an easy way to take control over resources to application providers. IaaS clouds need application developers or system administrators to install and configure the software stack that meets the need of application components. In contrast, PaaS (Platform as a Service) clouds offer ready-to-use execution environments for applications. So, while using clouds such as PaaS, developers should focus on programming their components and not spend much time in environment setup steps. since, because PaaS clouds may experience high usage PaaS providers must be able to scale execution environments accordingly.

9. CLOUD SCALABILITY ISSUES

The actions scalability are done by adding more restrictions in cloud computing for providing better service to customers and checking out both systems will work correctly or not after the addition of new features. Many problems arise after adding new restrictions. Scalability issues only arise when an organization adds additional requirements to make it easier for customers. When scalability issues arise, many web application problems are cloud-based virtual space. The problem of scalability can be divided into two types: the first type is horizontal scalability and the second is vertical scalability. Horizontal scalability can be determined by adding virtual events or repeating virtual appearance when a web application is under heavy load[22].

At present, the load balancing method is used to load web applications, and in many cases this method is a cost-effective solution for load balancing and increased productivity. As a rule, scalability is in favour of service providers and cloud services. So, if you face an increase in costs, scalability should not be used. It must be said here that this system has poor scalability. Also, transparency is more important when it comes to scalability for users. For example, users can store their data in the cloud without knowing where or how is kept when they use them.

9.1 Solutions

Scalability issues seen when scaling conversions to zoom in and out. To fix scalability issues and fix them first, the notification feature is used when constraints are added or removed. Therefore, alarm notification or management must be available. The best way to solve scalability problems when using load balancers. Load balancing is used to balance and manage all programs for traffic load applications. There are many sources of load balancing techniques, each related to a particular source application program. Also keep in mind that all information in the cloud is shared equally. Scalability is thus carried out according to user requirements. This increases the scalability of cloud computing. For the benefit of downscaling, initially, the removed restrictions will be discussed whether or not their removal will affect. Sometimes some memories are deleted, but on the other hand, they are used by some users and will lead to big problems. After completing the text editing, the paper is ready for the template. Duplicate the template file and use the naming convention prescribed by your conference for the name of your paper. In this new file, highlight all of the file contents and import your prepared file(text). Now you are ready to style your paper; using the scroll down window on the left of the MS Word Formatting toolbar[23].

CloudHealth can be very valuable for effectively managing many scalability elements in one or more clouds. CloudHealth determines which resources are right for size (a long-term measure to optimize costs and efficiency), and allows you to use policy-driven automation to generate signals for resources that need to be increased and reduced. This process effectively leads to the seamless management of your scalable resources.

One of the main advantages of CloudHealth over other cloud management platforms is that companies can manage many elements of scalability better, regardless of whether their assets are in the cloud or during infrastructure upgrading. Data can be collected to provide better visibility into their assets and make more informed decisions.

Creating a cloud network which offers maximum scalability potential level is entirely possible by applying a more 'diagonal' solution. By applying the best solutions present in both horizontal and vertical scaling, you will reap both the benefits. Once servers reach the threshold of no growth, you should start cloning them. This allows you to have a consistent architecture when including new components, software, apps and users. For most individuals, issues happen due to lack of resources not the inherent architecture of their cloud itself. A more diagonal approach should help you deal with the current and growing demands that you are facing[24].

9.2 Diagonal Scaling

Diagonal scaling is a combination of horizontal and vertical server scaling, where components are updated and added to the server at a critical point, and the server is then replicated in the current configuration. This provides the most efficient scaling mechanism in terms of cost and performance. In practice, server computing power is increased by increasing the number of processor cores, main memory, and disk storage. After the server's computing power peaks, or adding components is no longer profitable, a similar server is added to the tree to increase it horizontally.

This mechanism is highly effective for maximizing on performance and for increasing and improving throughput indefinitely. The infrastructure designed for only diagonal scaling, which involves a smooth path from a very small load server(such as a development machine) to a server setup which will withstand high-usage from many simultaneous requests[25].

9.3 Best Practices for Maximizing Cloud Scalability

The way and speed of which resources can be allocated, moved, and stopped has been revolutionized with the cloud. When one of your services or applications sees a surge or drop in use, you can dynamically provision your cloud to scale it however needed instantly. This alone may be a huge win for each area of your business, and paired with a robust plan, preparation and vigilance, the scalability of your cloud can help rocket your business to new heights. There are four ways to help you get the most out of your cloud in terms of scalability[26].

9.4 *Employ auto scaling*

Many cloud providers offer automatic scaling, which can be used to better manage resources and distribute workload balances appropriately. Automatic scaling is defined as the ability to increase or decrease based on special conditions automatically. This way, you can make sure that the right number of cases is always available to handle the burden on your application.

Automatic scaling defines IT-specific guidelines or milestones that automatically trigger the creation of new instances or expansion of existing instances, so that there is no constant monitoring of the traffic and resources used by each application. You can use multiple rules for the same service or application to add or subtract based on the event-based policies that you create. For example, if you recognize that an application always sees high use when dark(night) and low use in the morning, you will create a schedule-based policy that scales up the number of nodes in the evening and back down the next day. In addition, you shall create auto scaling triggers for those events you don't anticipate. This simulates the service replication scheme for the 17 different volumes of service load. On every service load, (1) conventional service system with (2) service replication scheme in terms of average response time are compared

9.5 *Use load balancing*

Load balancers are another automatic scaling option that distributes cross-node loads to maximize resources. Load balancing receives all incoming application traffic and then acts as an usher to find the best example for each incoming request that makes the best use of available resources. For example, if you are dealing with top users or resource consumption, load balancing tries to distribute your load among all available nodes to balance unused resources. Load balancers usually also continue to monitor the health of each instance to ensure that only traffic is sent to a healthy instance, and can also move loads that are considered heavy to a particular node, rather than looking for nodes that are less dense[27].

9.6 Employ containers with container orchestration

Containers - and container orchestration systems - are fast becoming a popular way to create a more scalable and portable infrastructure. Containers share a single core, but are isolated from their environment, which, unlike the whole machine, limits the problem to one container. Containers require less resources and offer more flexibility than virtual machines because, for example, they can share operating systems and other components. As a result, containers on the platform can function in the same way and can therefore be easily and quickly moved between nodes.

The beauty of containers is the ability to provide a large number of identical application examples, which, combined with low resource consumption, make containers a great way to scale up certain micro services. Container orchestration systems such as Docker Swarm, Amazon ECS and Kubernetes offer automated container management and coordination and enable automated services such as automatic placement (similar to load balancing) and automatic replication (similar to automatic scaling) for easier resolution on container stacks for scales. Containers are not ideal for every application. It is therefore important to evaluate your current application to determine which is suitable for containerization.

9.7 Test, test, test again

Your cloud environment could also be scalable – but it is that the application you would like to scale is ready to do so? Testing for scalability may be a crucial part of growing and will be done continuously to prevent bottlenecks later on. Make sure you add extra time at the end of your application development cycle to test for scalability to ensure you don't stumble across major issues when scaling that application later on[28].

9.8 Turn to Auto Scaling Services.

Automatic scaling is a special approach to dynamic scaling in the context of cloud services (ie scaling, where computing power is adjusted to the network load volume). In particular, service users with automatic scaling (the best known being Amazon Web Service, Google Cloud Platform and Microsoft Azure) receive additional virtual machines (from which connections can be automatically disconnected) if necessary, because traffic and query intensity are calmed. With services like that, you consume as much server performance as you need. This is a very profitable option compared to physical scaling if you have to buy expensive products and keep waiting for the hardware.

9.9 Microservices containers, clusterization

You can utilize resource-efficient, performance boosting strategies that wrap services into containers, then accumulate these containers into clusters[29]. The clusterization is trailed by characterizing contents that either add lacking assets(resources) to minimize the dedication of resources to avoid the excess.

9.10 Implement Caching

During horizontal scaling, a simple memory cache cannot be implemented for several nodes simultaneously, so it must be optimized. Specifically, memory such as Memcached or Redis can be used for the combined distribution of cached data between application iterations. These tools work according to different algorithms, so data caching is reduced. The cache is also well protected from replication and storage errors. When using a cache repository, it is very important to avoid situations where iterations of different applications require cached data at once. As such, with a correct approach, caching can help your systems get a **cloud scaling ability** to handle intensive loads and achieve an optimal output.

9.11 Employ CDN Services

CDN is a remote physical computer network that sends content to service users. In other words, it is distributed memory and cache usage. In general, another way to CDN is the best when web services, websites, or complete applications are targeted at a consumer audience spread across various countries. The price of a CDN depends directly on the amount of data traffic sent through the service[30].

Alternatively, CDN can be an unprofitable solution if your TA, despite its wide territorial distribution, has localizations with the concentration of certain users. I.e., suppose that about **60%** of your TA is based in the USA, **30%** in London, and the rest **10%** are scattered all over the planet. In such a case, using CDN will be a rational decision only for the latter 10% (whereas other locations will require new servers to be installed).

9.12 Assuring High Scalability in Cloud Computing

To execute scalability assuring schemes, there is cost involved, like the additional CPU and memory cost. The scalability obtained through schemes like these should be proportional to the cost spent for these schemes and cost-effectiveness must be considered during scalability assurance. Services should along with their SLAs include the quality specified level. Services should not undergo QoS degradation which includes acceptable scalability. Scalability assurance schemes must make sure that services fulfill the constraints of meeting the minimal limit of their QoS attributes two effective software-oriented schemes is obtained: service replication and service migration.

9.13 Service Replication

A technique which uses clone services which are presently executing on the other nodes to improve/increase the service load without influencing operations (activities) in progress. Replicated services secure extra assets(resources) provided by the new nodes to large service load. Also, service replication(duplication) improves service scalability and decreases QoS degradation thereby handling more heavier service loads. For a case study, a service load as a variable was set. The service load is the number of service invocations per unit time. We set 500ms as the unit time. That is, if ten invocations occur within 500ms, then the service load is 10.

9.14 Service Migration

This is a scheme which introduces a service on an alternative node when a node cannot provide high QoS due to hardware issues or software issues. After this process, the migrated service performs the usual role as it did on the unstable node, and the unstable node is removed from the list of service nodes. Table 2 shows the result for service migration. Service load,Scalability assuring scheme appliance, variables, are usual as the service replication simulation. In this, a service is migrated to a closer node from users[31]. To make this possible, they assume that the response time is proportional to the distance. Therefore a service is migrated to the closest node in terms of the response time. The integration of cloud along with Software Defined Networks(SDN) is possible with respect to the interfaces available. The various link failure issues are addressed which can help in analyzing the cloud integration[32].

9.15 Improving Performance

To improve scalability by adding new controllable switches to the network, with several new devices added to the network. This step is a successful effort to reduce usage costs and increase scalability. On the other hand, the cost of replacing it is still a big problem. A model was proposed that provides cash from customers, it was a successful effort to increase scalability and reduce the cost of SaaS (Software as a Service), but with cloud scalability in Regarding PaaS and IaaS is still a big problem. SIS (Scalability Enhancement System): SIS is a combination of load balancing, load analyzer, and cache. Load analysis; Analyze all requirements that users play *on the system directly or through SaaS. It monitors the* requested data repeatedly and writes the requested element repeatedly to cache 1 or cache 2 depending on the interface, regardless of whether they are connected to SaaS or IaaS. The next time the same item is requested, it will be served efficiently by the appropriate cache. Also, if one of the caches overflows, the load balancing mode is activated and the excess data is moved to another cache, resulting in quick access to the requested items repeatedly.

10. CONCLUSION

The paper discussed about the scheduling algorithms, their usage and concepts in it. It also discussed the concepts and issues of scalability in cloud computing. The survey helps to analyze the complete scenario in terms of scheduling algorithms and scalability issues in cloud computing.

REFERENCES

[1] Jiandun Li, Junjie Peng, Zhou Lei, Wu Zhang, “An Energy- efficient Scheduling Approach Based on Private Clouds”, Journal of Information & Computational Science, april 2011.

[2] NIST Definition of Cloud Computing v15, csrc.nist.gov/groups/SNS/cloud-computing/cloud-def v15.doc

[3] Baomin Xu , Chunyan Zhao,Enzhao Hua,Bin Hu,“Job scheduling algorithm based on Berger model in cloud environment”, Elsevier publications, march 2011.

[4] Kyong Hoon Kim, Anton Beloglazov, Rajkumar Buyya, “Power-aware Provisioning of Cloud Resources for Real-time Services”, ACM Publications, December 2009.

[5] JiandunLi,Junjie Peng,ZhouLei,WuZhang,“AScheduling Algorithm forPrivateClouds”, Journal of Convergence Information Technology, Volume6, Number 7, July 2011.

[6] Golnar Gharoonifard, Fahime Moeindarbari, Hossein Deldari, Anahita Morvaridi, “Scheduling of scientific workflows using a chaos-genetic algorithm”, Elsevier publications, 2010.

[7] Ruay-Shiung Chang, Jih-Sheng Chang, Po-Sheng Lin, “An ant algorithm for balanced job scheduling in grids”, Future Generation Computer Systems, june 2008.

[8] JiandunLi,JunjiePeng,ZhouLei,WuZhang,“AScheduling Algorithm for Private Clouds”, Journal of Convergence Information Technology, Volume6, Number 7, July 2011.

[9] Loannis A.Moschakis, Helen D. Karatza, “Evaluation of gang scheduling performance and cost in a cloud computing system”, Springer publications, 2010.

[10] Suraj Pandey, LinlinWu, Siddeswara Mayura Guru, Rajkumar Buyya, “A Particle Swarm Optimization-based Heuristic for Scheduling Workflow Applications in Cloud Computing Environments”.

[11] Muhammad Asad Arfeen, Krzysztof Pawlikowski, Andreas Willig, "A Framework for Resource Allocation Strategies in Cloud Computing Environment"

[12] Y. Chawla and M. Bhonsle, "A study on scheduling methods in cloud computing," Int. J. Emerg. Trends Technol. Comput. Sci., vol. 1, no. 3, pp. 12–17, 2012.

[13] T. Ma, Y. Chu, L. Zhao, and O. Ankhbayar, "Resource Allocation and Scheduling in Cloud Computing: Policy and Algorithm," IETE Tech. Rev., vol. 31, no. 1, pp. 4–16, Jan. 2014.

[14] WeiWang,GuosunZeng,DaizhongTang,JingYao,"Cloud- DLS: Dynamic trusted schedulingfor Cloud computing", Expert Systems with Applications, 2011.

[15] Anju Bala, Inderveer Chana, "A Survey of Various Workflow Scheduling Algorithms in Cloud Environment", 2nd National Conference on Information and Communication Technology (NCICT) 2011, International Journal of Computer Applications publications.

[16] Maram Mohammed Falatah, Omar Abdullah Batarfi,"CLOUD SCALABILITY CONSIDERATIONS",International Journal of Computer Science & Engineering Survey (IJCSES) Vol.5, No.4, August 2014.

[17] Ab Rashid, Dr.D.Ravindran,"SURVEY ON SCALABILTY IN CLOUD ENVIRONMENT",International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)Volume 5, Issue 7, July2016.

[18] L.Arockiam, A. Stanisals,"SCALABILITY ISSUES AND RESEARCH CHALLENGE IN CLOUD COMPUTING", ACM SIGCOMM Computer Communication Review, Vol. 39, 2009, pp. 50-55.

[19] Thamararai Selvi, V.Praba, Mahesh Arumugam,"SCALABILTY ISSUES IN CLOUD COMPUTING"(Explained about scalability issues in cloud computing based on several factos)(2012 Fourth International Conference on Advanced Computing (ICoAC).

[20] Luis M.Vaquero,Luis Rodero Merino, Rajkumar buyya,"DYNAMICALLY SCALING APPLICATIONS IN THE CLOUD"(ACM SIGCOMM Computer Communication Review.

[21] Uranus Kazemi, Reza Boostani,"Analysis of Scalability and Risks in Cloud Computing"International Journal of Academic Research in Computer Engineering Print ISSN: 2476-7638 and Online ISSN: 2538-2411Vol. 2, No. 1, Pages.24-33,May2018.

[22] Hanieh Alipour, Yan Liu, Abdelwahab Hamou-Lhadj," Analyzing Auto-scaling Issues in Cloud Environments".

[23] Nishant Agnihotri," EVALUATING PAAS SCALABILITY AND IMPROVING PERFORMANCE USING SCALABILITY IMPROVEMENT SYSTEMS".

[24] Jian Wu ; Qianhui Liang ; Elisa Bertino ," Improving Scalability of Software Cloud for Composite Web Services" 2009 IEEE International Conference on Cloud Computing.

[25] Chao-Rui Chang ; Meng-Ju Hsieh ; Jan-Jan Wu ; Po-Yen Wu ; Pangfeng Liu ,"HSQL:A HIGHLY SCALABLE CLOUD DATABASE FOR MULIT-USER QUERY PROCESSING". . In this paper, we create a new, distributed B-tree column indexing scheme for HBase, which can support indexing for non-row-key columns, as well as parallel B-tree search in large data table. 2012 IEEE Fifth International Conference on Cloud Computing.

[26] Aakash Tyag," A Review Paper on Cloud Computing" International Journal of Engineering Research & Technology (IJERT)ISSN: 2278-0181Published by, www.ijert.orgVIMPACT - 2017 Conference Proceeding.

[27] Marram Mohammed, Omar Batarfi,"Cloud Scalability Considerations"(International Journal of Computer Science & Engineering Survey (IJCSES) Vol.5, No.4, August 2014.

[28] RS.M.Lakshmi, Santhi Sri Kurra,Nirupama Mundukar," A STUDY ON SCALABILITY OF SERVICES AND PRIVECY ISSUES IN CLOUD COMPTING "(ICDCIT 2012:DISTRUBUTED COMPUTING AND INTERNET TECHNOLOGY pp 212-230.

[29] Manasaf Alam, Kashish Ara Shakil, "RECENT DEVELOPMENTS IN CLOUD BASED SYSTEMS:STATE OF ART"(IJCN, vol. 3, no. 5,pp. 247-255, 2011.

[30] ThepparitBanditwattanawong,PutchongUthyopas,"IMPORVING CLOUD SCALABILITY, ECONOMY AND RESPONSIVNESS WITH CLIENT SIDE" 2013 10th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology

[31] Luis Soares, Jose Pereiea,” Improving the Scalability of Cloud-Based Resilient Database Servers”, DAIS 2011: Distributed Applications and Interoperable Systems pp 136-149.

[32] V. Muthumanikandan and C. Valliyammai, "A survey on link failures in software defined networks," 2015 Seventh International Conference on Advanced Computing (ICoAC), Chennai, 2015, pp. 1-5, doi:10.1109/ICoAC.2015.7562808.

Dr. V. Muthumanikandan B.E., M.E., Ph.D., is working as a Senior Assistant Professor in the School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, India. He is serving as a Placement Coordinator for PG departments. He received his B.E, M.E and Ph.D. degree in Computer Science and Engineering discipline. His areas of interests include Networking, Cloud Computing, Software Defined Networking and Network Function Virtualization. He published many papers in reputed journals, conferences and book chapters. He published two patents. Acting as a reviewer for many international journals. An active member in various professional bodies like CSI, IAENG and IRED.