Journal of Theoretical and Applied Information Technology
30th April 2019. Vol.97. No 8

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A NOVEL SCALE APPROACH FOR LOAD BALANCING WITH HARDWARE AND SOFTWARE IN CLOUD AND IOT PLATFORM

[1]NARANDER KUMAR, [2]SURENDRA KUMAR
[1, 2]Dept. of Computer Science, B. B. A. University (A Central University), Lucknow, UP, India
E-mail: [1]nk_iet@yahoo.co.in, [2]kumar.surendra1989@gmail.com

ABSTRACT

In the current scenarios cloud computing is most talked innovation which offers resources from the extensive server farms. The fundamental motivation behind cloud computing is to enable customers to take the favourable position from these advancements. Cloud computing is made accessible as compensation on interest administration to the customers. Cloud administrations are "pay-per-use" over the web. It has numerous highlights that incorporate estimated administrations, accessibility, security and scalability. Among all the most and energizing component of distributed computing is Scalability which offers the customers the straightforwardness and solace to utilize the resources according to their desires and request. Versatility system could really compare to whatever else. Resources can be scaled in and scaled out when the interest and circumstance emerges that way. It may be static or dynamic in nature. In this paper, we proposed a scalability mechanism for load balancing to recourse management policies in the cloud computing. We check our proposed mechanism with the different parameters likes scalability model, measurement of bad detections, Performance heuristics, Diagnose of the Performance, production area and scalability area. We also make the various comparisons between different kinds of results. After that we simulate the results in the cloudsim and get the optimized results with the help of appropriate examples.

Keywords: *Scalability, Load Balancing, Scale In, Scale Out, Cloud Computing*

1. INTRODUCTION

Cloud computing gives nearly everything as administration to clients by its compensation per utilize display. Cloud is an administration arranged innovation, which conveys resources on client request. These assets can be any computational resource that works correspondingly to general registering assets, despite the fact that resources got to over web postponements or slacks in administrations because of inertness. Another issue is tied in with scaling this cloud condition up to its greatest accessibility, which boosts the resource usage to its limit. The disclosure and planning of resources relies upon variables like adaptability, accessibility, load adjusting, versatility and so on. These elements should be considered for making substantially more dependable framework for better cloud show as far as clients heading. In cloud, each customer can get to administrations, store, process, and get back huge volumes of data for business and household use without having any resources by pay for use technique. The qualities, for example, virtualization, no pre-venture cost, simplicity of execution, and adaptation to internal failure takes distributed computing as it were. The outrageous interest of cloud results enormous datacenters and different frameworks. Tremendous measure of intensity utilized in cooling datacenters results in overwhelming carbon radiation. Tremendous measure of intensity is spent in cooling the system frameworks of datacenters prompts high carbon discharge. Power devoured for cooling the framework coming about negative suspicions on cloud computing. Cloud is a cognizant registering of cloud that is control productive which utilizes resources for processing alongside the executives of waste. The change to cloud can be extremely valuable in lengthy time span to condition. Visit server breakdown is a pivotal concern, when servers are underutilized with out of line remaining task at hand. The real reason for server blackout is the need of the customers that their solicitations need to keep running in division.

The created software depends on disseminated design benefit arranged to distribute computing. With the advancement of PC equipment and systems administration, conveyed designs have additionally developed, particularly benefit based distributed computing has changed the customary

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

PC and concentrated stockpiling approach. It encourages handling and capacity abilities according to the necessity. IaaS gives Virtual Machines completely fulfilling the client request for as far as resources. The resources of the suppliers are normally facilitated as a server farm. The server farm is a lot of physical machines which are interconnected, virtualized, and geologically dispersed. Since the client may have diverse geographic area, a specialist co-op ought to have disseminated server farms all through the world to give administrations to the clients. In the distributed computing, the separation between datacenters prompts unfortunate system inertness, which thus prompts delay in administrations. For instance, a VM allotted in a server farm, far from client area, the client will experience the ill effects of postponed reaction because of the confinement of system resources. Moreover, if the VM is vigorously stacked it builds the reaction time to the client, contrasted with if VM is having less outstanding task at hand. Subsequently, a supplier is required to discover reasonable server farms for serving a client dependent on the client area and remaining task at hand of the server farms. Versatility of the cloud administrations is a critical concern. It ought to be straightforward to clients, so clients may take a shot at a cloud asking for administrations without fretting over how and from where they are given. In the Cloud computing condition is deals with a pool of figuring and information preparing resources that change endlessly as far as models, sizes, and setups, and are provisioned to end clients, either in a natural shape or as an administration. This pool of resources and administrations is normally disseminated and all inclusive open through the Internet. Commonplace building obstructs inside a cloud are multi-center processor based frameworks. These multi-center based frameworks interface with one another through LAN's and WAN's. Many fascinating appropriated processing frameworks of the past were worked thusly, which plan occupations on accessible pool of hosts to effectively use the accessible figuring power. Customarily LAN was interconnects between such frameworks. Applications firmly fixing to basic stages processor, interconnect and working framework was one of the obstacles to consistently convey such frameworks on geologically scattered areas. Along these lines, Cloud Computing varies from customary circulated processing framework regarding its pervasiveness [1].

2. RELATED WORK

Names Ongoing years have seen critical endeavours gone for enhancing Open Shortest Path First union speed and additionally scalability and stretching out Open Shortest Path First to accomplish consistent mix of scalable adhoc systems with ordinary wired systems [2]. By exploiting server farm organize engineering the creators plan the traffic scalability issue as a combinatorial advancement model of online VM arrangement with multi-dimensional resources imperatives. By utilizing Markov guess strategy, the ideal is proficiently gotten. Execution assessment shows that the proposed strategy accomplishes huge traffic versatility enhancement more than two regular heuristics [3]. They exhibited NSGA_SR approach that uses both target and abstract evaluations and models positioning issue as a multi target advancement and afterward fathom it with utilization of non-commanded arranging hereditary calculation. Numerical examinations, affirmed that the proposed methodology beats accessible methodologies as far as scalability and versatility with expanding number of clients and administrations. Additionally it meets to enhancement of objectives and has great dependability amid the diverse ages. Likewise it incorporates no restriction with respect to any added substance new quality characteristic, benefit or beneficial capacity [4]. They give an algorithmic reasoning on the new structure difficulties for the thick heterogeneous Cloud-RAN dependent on curved advancement. As issue sizes scale up with the system estimate and exhibit that it is basic to take one of a kind structures of plan issues and inalienable attributes of remote channels into thought, while arched improvement is fill in as an incredible resources. System control minimization and channel state data securing are be utilized as two normal guides to show the viability of curved advancement and a two-arrange structure to explain general vast scale raised streamlining issues, which is amiable to parallel usage in the cloud server farm [5]. They examine cross-layer resources designation demonstrates for CRAN to limit the general framework control utilization, fiber joins and the remote radio heads. He describes the cross-layer resources allotment issue as a blended number nonlinear programming, which together considers flexible administration scaling, RRH determination, and joint bar framing. The MINLP is anyway a combinatorial enhancement issue and NP-hard. We loosen up the first MINLP issue into an all-encompassing entirety utility expansion issue, and two diverse arrangement approaches [6]. Idea of an

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

resources package a delegate resources utilization dissemination for a gathering of hubs with comparative resources use designs that utilizes two correlative strategies to defeat the constraints of existing systems: resources use histograms to give measurable certifications to resources limits and grouping based resources collection to accomplish scalability. Utilizing follow driven recreations and information investigation of multi month-long PlanetLab follow, Resource groups can give high exactness to factual resources disclosure, while accomplishing high scalability. Resources packs are in a perfect world suited for distinguishing bunch level qualities [7]. Scalable resources portion utilizing vertical versatility to guarantee predefined QoS measurements, for example, reaction time and throughput is satisfied. QoS infringement is checked and identified utilizing fluffy on the off chance that tenets to decide the quality status. When infringement or likely infringement is identified, framework assessment is performed to distinguish conceivable mistaken conduct which at that point prompts the execution of enhanced resources distribution exercises. There situations are to show the possibility of through simulation test [8].

An approach towards an extensive investigation of different recreation based devices to test and quantify the Cloud Datacenter execution, scalability, strength and multifaceted nature. The server machines should be examined for their degree of usage regarding vitality and administration to customers in distributed computing. They have broken down different Cloud resources utilizing CloudSim, CloudReports and Cloud Analyst apparatuses. Some Simulation test results and Simulations are exhibited so as to contrast those with constant situation with bring the execution and versatility issues into our notice for future headings [9]. Plans experienced scalability, scalability issues. In ABE based access control models information will be put away in encoded frame on cloud and information proprietor will share required keys with the information customer for getting to the information. ABE plans experienced executing complex access control strategies. Key Computation is another region where the vast majority of the ABE conspired endured. They had considered various levelled property set-based encryption display which was created by expanding figure content arrangement quality set-based encryption with a progressive structure of clients. They worked with HASBE plan to defeat the restriction of scalability and scalability [10]. A Hadoop stage arrangement strategy is for different cloud frameworks with the Occopus cloud orchestrator device. A mechanized arrangement gives a simple to-utilize convenient and scalable approach to send the well known Hadoop stage with the fundamental objective to maintain a strategic distance from merchant locking issues, i.e. there is no reliance on any cloud supplier arranged and offered virtual machine picture or "discovery" Platform-as-a-Service system [11]. Fundamental to the upkeep of extensive scale frameworks is checking which takes into consideration the discovery of deficiencies, mistakes and abnormalities and the ordering of enhancement and restorative measures. Observing expansive scale frameworks is critical test requiring the low inertness development of huge volumes of information and close ongoing examination. This test is amplified by flexibility and other cloud properties which past observing frameworks don't yet completely represent. Cloud mindful observing instrument is that gives powerful, blame tolerant checking at scale. Likewise depict in detail the systems which empower Varanus to work successfully and investigate the execution of Varanus through a definite assessment [12]. Utilizing halfway reconfiguration, our equipment and programming structure virtualizes physical FPGAs to give numerous autonomous client plans on a solitary gadget. Basic segments are the administration of the virtual client characterized quickening agents, and in addition their relocation between physical FPGAs to accomplish higher framework wide usage levels. They make homogenous segments over an inhomogeneous FPGA texture to offer a reflection from physical area, size and access to the genuine equipment [13]. They present a novel system security design for distributed computing thinking about qualities of distributed computing. In particular, it right off the bat gives assurance to both outer and interne deals in distributed computing, furthermore accomplishes adaptable scalability concerning virtual middlebox stack, and after that accomplishes blame tolerant among virtual middlebox disappointment. Examinations and reenactments on our confirmation of-idea model of system security engineering for distributed computing considering attributes approve that organize security design for distributed computing considering qualities is a viable engineering with negligible execution overhead and that it very well may be connected to broad down to earth advancement in distributed computing [14].

A design is dependent on grouping virtual machines in datacenters for higher accessibility of

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

resources with enhanced versatility. Grouping causes virtual machines to reconfigure and simple planning. The resource partaking in cloud will enhance and clients get augmented outcome. After that coordinates a numerical model for clarifying the ideas of the framework. The current framework is being balanced utilizing reproduction instruments [15]. The fundamental rationality of our methodology is characterizing a vitality ideal activity routine and endeavouring to amplify the quantity of servers working in this routine. Inert and delicately stacked servers are changed to one of the rest states to spare vitality. The heap adjusting and scaling calculations likewise misuse probably the most alluring highlights of server combination components [16]. Cloud architecture is utilizes reconfigurable rationale to quicken both system plane capacities and applications. This Configurable Cloud design puts a layer of reconfigurable rationale between the system switches and the servers, empowering system streams to be programmable changed at line rate, empowering speeding up of neighbourhood applications running on the server, and empowering the FPGAs to impart straightforwardly, at datacenter scale, to gather remote FPGAs unused by their nearby servers. They sent this structure over a generation server bed and show how it very well may be utilized for both administration speeding up and arrange increasing speed [17].

The primary points is to feature the requirement for the vertical scalability administration and plan the proper cloud, virtualization layer, and working framework administrations to fuse vertical versatility in current cloud stages such that will make it monetarily and in fact proficient for the end clients to utilize cloud virtual machines as though they are utilizing their own workstations [18]. They present conceptual architecture as a premise to structuring cloud-local checking frameworks. In the Benchmarked several open source time arrangement databases cloud frameworks with some points with remaining burdens from practical mechanical applications and found that in any event KairosDB satisfies our underlying speculations concerning scalability and unwavering quality [19]. They made of scalable applications for open cloud to screen their expenses and create cost-mindful scalability controllers and present a model for catching the valuing plans of cloud administrations. That decides and assesses the application's expenses relying upon its utilized cloud administrations and their charging cycles and also assess cost effectiveness of cloud applications, investigating which application part is cost productive to reallocate and when. Scalable stage is for IoT, sent in Flexiant1, one of the main European open cloud suppliers. Cost-mindful versatility can accomplish higher application dependability and execution, while diminishing its activity costs [20]. They talked about a scheduling algorithm and auto-scaling activating techniques that investigate client persistence, a metric that evaluates the recognition end-clients have from the Quality of Service conveyed by a specialist co-op dependent on the proportion among expected and real reaction times for each demand. The methodologies help decrease costs with resources designation while keeping up apparent QoS at sufficient dimensions. Results indicate decreases on resources hour utilization by up to roughly 9% contrasted with conventional methodologies [21].

Analyse the hole between these two integral advancements and bringing Clouds and work processes together. At that point the key difficulties in supporting Cloud work processes and present our reference structure for logical work process the board in the Cloud and involvement in coordinating a logical work process the board framework—Swift into the Cloud [22]. A methodology dependent on the learning automata for the versatility of the web applications which consolidates virtual machine bunches and the learning automata so as to give the most ideal path to the scaling up and downsizing of the virtual machines. The consequences of this show the methodology has diminished the quantity of SLA violations, while it has a littler heap of scalability contrasted with alternate methodologies in such manner [23].

3. PROPOSED MECHANISM

Distributed computing scaling must be measured. On the off chance that you can't measure it, you can't promise it. The law of computational scaling gives that measurement. One of the best obstructions to applying queuing hypothesis models (regardless of whether logical or simulation) is the equivocalness of request times inside an application. Each queuing in an execution display requires a request time as an info parameter. Without the suitable queues in the model, framework execution measurements like throughput and reaction time can't be anticipated. The scalability jumps this whole issue by not requiring any low-level request time estimations as information sources.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

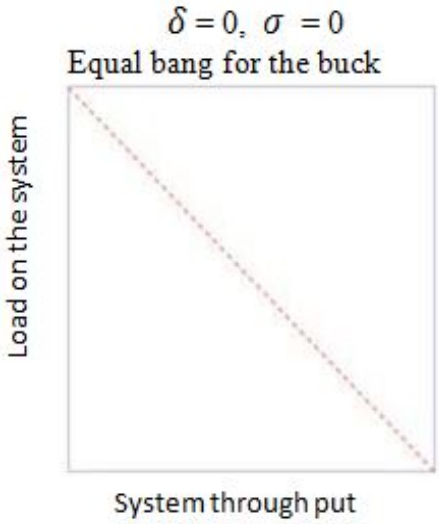

Figure 1: Equal bang for the buck

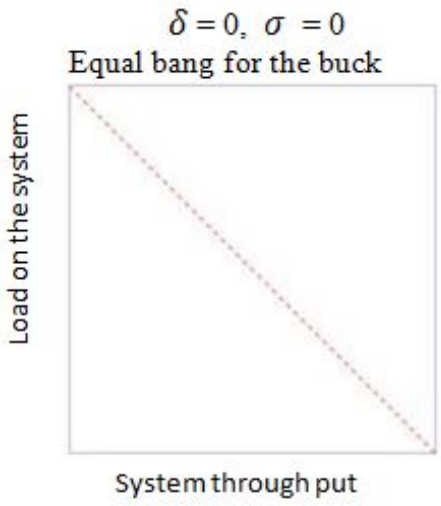

Figure 2: Cost of sharing resource

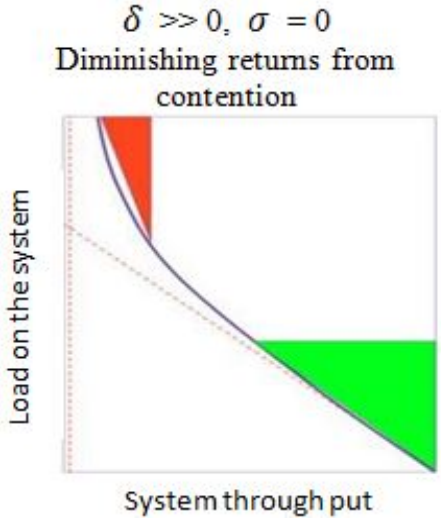

Figure 3: Diminishing returns from buck

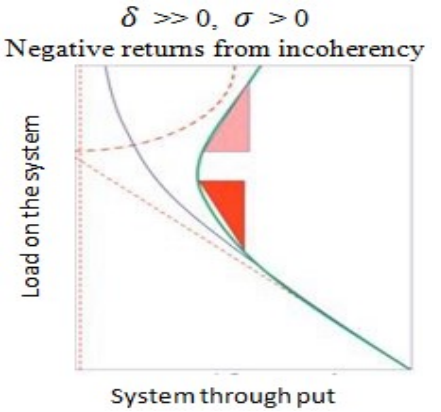

Figure 4: Negative returns from contention incoherency

Journal of Theoretical and Applied Information Technology
30th April 2019. Vol.97. No 8

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Every one of these commitments can be consolidated into an equational model to register the relative limits:

$$A(P) = \frac{\lambda P}{1+\delta(P-1)+\sigma P(P-1)} \text{.....(1)}$$

The relative limit F(P) is the standardized throughput at each progressive load N, and consequently, any throughput can be communicated as A(P) = F(P) A(1). The heap may be instigated either by P virtual clients e.g., created by a heap testing and clients on a generation stage.

The three coefficients, δ,σ,λ in eqn. (1) can be identified respectively with the three Cs:

3.1 Concurrency

It can likewise be translated as either: the slant related with straight rising scalability, i.e., the line A(P) = λP in Fig. A when δ = σ = 0 the greatest throughput feasible with a solitary load generator, i.e., A(1) = λ.

3.2 Contention

(with extent δ) because of sitting tight or queuing for shared resource.

3.3 Coherency

(with extent σ) because of the postponement for information to end up reliable by temperance of point-to-point transfer information between resources that are appropriated.

3.3.1 Software Scalability

The quantity of load and user (P) is increased on settled equipment. For this situation, the quantity of users goes about as the autonomous variable while the processor stays settled over the scope of client stack estimations.

3.3.2 Hardware scalability

The quantity of processors (P) is augmented in the equipment while keeping the user stack per processor settled. For this situation, the users executing per processor (e.g., 100 clients for every processor) is expected to continue as before for each additional processor. For instance, on a 32 processor stage you would apply a load of P = 3200 clients to the test stage.

Eq. (1) discloses to us that application and hardware scalability are opposite sides of a similar coin: something not for the most part perceived. A non-zero estimation of σ is related with estimated throughput that goes retrograde, i.e., diminishes with expanding burden or configuration of the platform.

The allocated coefficients in eq.(1) at that point give a gauge of the client stack at which the most extreme scalability will happen on the off chance that D:

$$\sqrt{\frac{(1-\delta)}{\sigma}} \text{.....(2)}$$

The λ parameter assumes no job in eq. (2). That is on the grounds that it finds the situation of the most extreme in the scalability bend on the x-pivot, though λ scales the y-hub esteems. The relating most extreme throughput Amax is found by substituting P_{max} into eq. (1).

$$A_{max} = A(P_{max}) \text{.....(3)}$$

Scalability in Fig B and C (above) compare to σ = 0, which implies that P_{max} happens at limitlessness. Put in an unexpected way, the versatility bend essentially approaches a ceiling at 1/ δ.

4. EXPERIMENTS OF THE SCALABILITY MODEL

From the proposed mechanism we get the better scaling approach and better performance in the cloud environments. We can test the various parameters and analyze it in the different types of problems.

4.1 Experiments of the scalability model

Scalability in eq.(1) permits you take an arrangement of load estimations at any rate about six information focuses and from those information decide how your informational collection will scale under bigger client loads than you might have the capacity to create in your test.

Virtual users P	script/hours A(P)	realcap F(P)	efficiency F/P	inverse P/F	linearity P-1	deviation (P/F)-1
1	65	1	1	1	0	0
18	996	15.35	0.85	1.17	17	0.17
36	1652	25.46	0.71	1.41	35	0.41
72	1853	28.55	0.4	2.52	71	1.52
108	1829	28.18	0.26	3.83	107	2.83
144	1775	27.35	0.19	5.27	143	4.27
216	1702	26.23	0.12	8.24	215	7.24

Trendline Perabola	Perameters Co-efficient	Super Parameter	Serial Values	Users	Predicted F(P)	Capacity Modeled	Measured
a	8.001245875	δ	0.0171	1	1.00	64.90	65
b	0.0171	K	0.0001	18	13.70	889.16	996
c	0.0000	N*	111	36	21.22	1377.19	1,652
				72	27.51	1785.30	1,853
				108	28.83	1871.31	1,829
				144	28.34	1839.24	1,775
				216	25.79	1673.94	1,702

Figure 5: The excel data for large user load

Journal of Theoretical and Applied Information Technology
30th April 2019. Vol.97. No 8

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

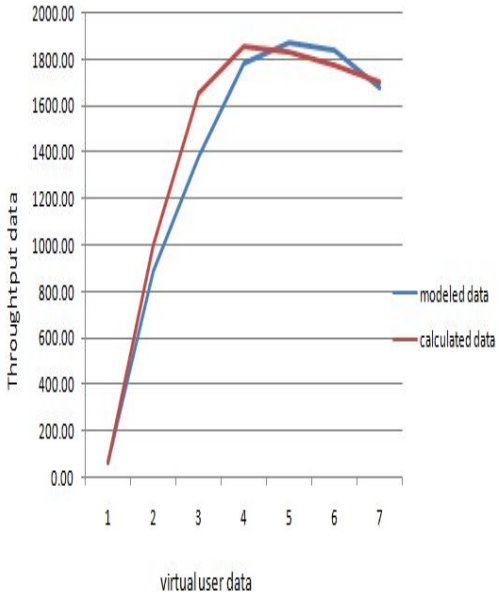

Figure 6: the scaling larger user loads data results load

4.2 Measurements of bad Detection

In the Eq. (1) isn't an as much as expected. It can't anticipate the beginning of broken estimations. At the point when the information separate from the model that does not naturally make the model off-base. You have to quit apportioning and discover why.

4.3 Performance heuristics

The overall sizes of δ and σ parameters let you know individually, regardless of whether dispute impacts or coherency impacts are in charge of poor versatility.

4.4 Diagnose of the Performance

What makes simple to apply additionally restricts its analytic capacity. On the off chance that the parameter esteems are poor, you can't utilize it to disclose to you what to settle. All that data is in there okay, yet it's packed into the estimations of those two less parameters. Notwithstanding, other individuals e.g., application developers, the frameworks planner, may effectively distinguish the issue once the all inclusive law has disclosed to them they have to search for one.

4.5 Production area

Applying the scalability to execution information gathered from creation situations with blended remaining tasks at hand is a momentum region of research. The primary issue is deciding the fitting free factor, e.g., P clients or procedures, not needy factors like use ρ(P). At that point you just need A(P) information as the needy variable to relapse against. Those A(P) qualities ought to be resolved from information that are gathered amid an estimation window with as couple of huge transient impacts as could be expected under the circumstances, i.e., near unfaltering state.

4.6 Scalability area

The thought is, rather than simply considering a straightforward bend fit to the information focuses, let the information fall over various regions whose limits are characterized by an arrangement of fitted scalability parabola curves. The limits characterize zones as demonstrated as follows. Additionally, these zones can be specifically indentified with sequential impacts and this can be colossally useful to applications designers searching for approaches to change their code to enhance execution.

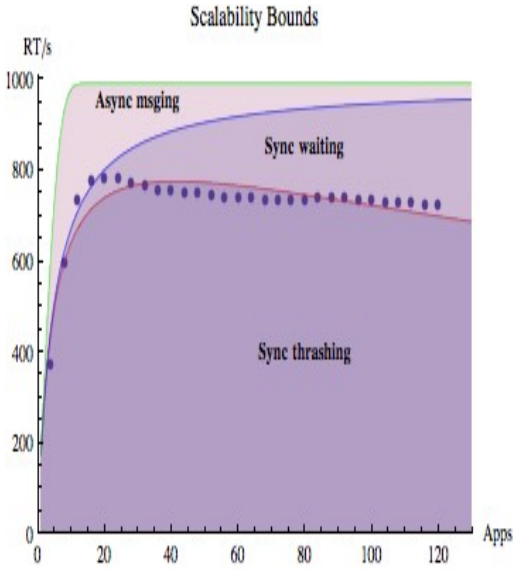

Figure 7: Measured scalability in the dots

Precedent, the above figured plot demonstrates the deliberate scalability (in dots) of the tool used WebSphere MQ V6.0 nonpersistent and sized 2KB messages (in Rnd Trips/sec) as an element of the expanding number of driving Apps. The hued regions navigated by the information (in dots) demonstrate the sort of synchronous queuing impacts in charge of the clear loss of scalability above about P=15 Apps.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

5. RESULT AND DISCUSSION

The workload attributes were demonstrated and settings characterized the remaining tasks at hand were carrying on arbitrarily with exponential conveyances for all the span of the simulation. The simulator was kept running for 7 hours, which was observed to be adequate for gathering measurements to prepare the BN model. In this analysis are the subtleties of the gathered insights which were gathered at customary interims of 60s and encouraged into the BN learning process [24]. They have distinguished two key scalability highlights of utilizations. First is workload parallelism, which depicts to what degree an application can parallelize its outstanding task at hand so as to use expanded CPU. Amid the execution of an application, its outstanding task at hand parallelism can be estimated by the quantity of strings in the application that are dynamic and gaining ground. In a day and age, the higher the remaining task at hand parallelism is the more advancement the application can make whether gave more resource. An application with higher workload parallelism will in general show higher execution versatility on both physical machines (on account of less inert time) and virtual machines (due to less unused CPU time) [25].

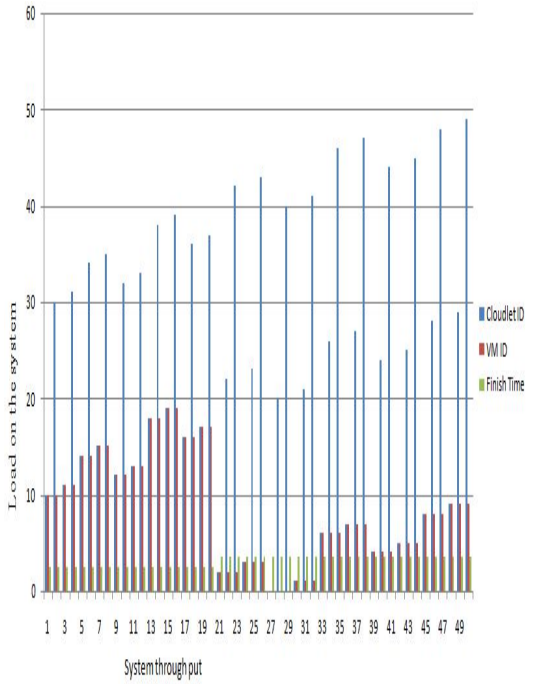

Figure 8: Loads on the system and throughputs

To comprehend the potential for accomplishing better scalability with enhanced framework plans, we evaluated the resources usage efficiencies that could accomplish if the VM migration could bolster the VM to use its time cut as completely as could reasonably be expected. We chose different sort, in light of the fact that right off the bat we need to concentrate on enhancing the distribution of CPU time, rather than diminishing the overhead of Virtual CPU switches, besides there is space to additionally enhance their scalability.

The estimation depends on profiling on the physical server. For every assignment period amid the execution, we gather the CPU time use. On the off chance that the use is higher than the part of the CPU time that a VM can get. We expect that, with an all around planned VM relocation, the VM can drain the time cut apportioned to it and accomplish a productivity of 100% when the calculation is executed on the VM. Else, it doesn't have enough calculation to exhaust the time cut apportioned to the VM. Hence, in the period, the productivity is the proportion among use and part of CPU time designated to a VM. The assessed resources usage effectiveness is the normal proficiency amid the execution. In the our research work we take five datacenters, 50 CloudLets, 50 VMs, 2GB RAM, dual-core machine and simulation run over the cloudsim and get the scaling results which are shows in the table (1).

6. CONCLUSION

Scalability has pursued a turned line being connected to centralized servers, disseminated frameworks, back to centralized computers and back to a "concentrated" Cloud which is dispersed and heterogeneous, however observed as a solitary element by edge gadgets getting to the Cloud through institutionalized interfaces so as to execute administrations. Scaling capacities have, in this manner, wandered among horizontal and vertical scalability, for the dominant pattern to frameworks structure and usage. In the proposed scalability mechanism shows the optimized load balancing in the cloud environment as well as we also shown some of the most prominent examples of Cloud-enabled scalability at the different Cloud computing environments. We proposed the scalability model and measured it in the different parameters as well as mathematically. A Scalable load of the systems are shows optimized results. Scalability is offered in a transparent manner for the end user (either a service provider or a service consumer). For the future purpose we would like take various other

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

parameters like quality of services, service level agreements and reduce the energy consumptions in the cloud environments.

REFRENCES:

[1] M, H, Jamal., A, Qadeer., W, Mahmood., A, Waheed., J, J, Ding, "Virtual Machine Scalability on Multi-Core Processors Based Servers for Cloud Computing Workloads", *IEEE International Conference on Networking, Architecture, and Storage, Hunan*, 2007, pp. 90-97.

[2] M. Goyal, E. Soperi, G. Baccelli, S. A. Choudhury, H. Hosseini, K. Trivedi, "Improving Convergence Speed and Scalability in OSPF: A Survey", *IEEE Communications Surveys & Tutorials*, Vol. 14, No. 2, 2012, Pp. 443-463.

[3] T. Ma, J. Wu, Y. Hu, W. Huang, "Optimal VM placement for traffic scalability using Markov chain in cloud data centre networks", *Electronics Letters*, Vol. 53, No.9, 2017, pp.602-604.

[4] A. J. Ahani, L. M. Khanli, "Cloud service ranking as a multi objective optimization problem", *The Journal of Supercomputing*, Vol. 72, No. 5,2016, pp.1897-1926.

[5] Y. Shi, J. Zhang, K. B. Letaief, B. Bai, W. Chen, "Large-scale convex optimization for ultra-dense cloud-RAN", *IEEE Wireless Communications*, Vol. 22, No.3, 2015 pp. 84-91.

[6] J. Tang, W. P. Tay, T. Q. S. Quek, "Cross-Layer Resource Allocation With Elastic Service Scaling in Cloud Radio Access Network", *IEEE Transactions on Wireless Communications*, Vol. 14, No.9, 2015, pp. 5068-5081.

[7] M. Cardosa, A. Chandra, "Resource Bundles: Using Aggregation for Statistical Large-Scale Resource Discovery and Management", *IEEE Transactions on Parallel and Distributed Systems*, Vol. 21, No. 8, 2010, pp. 1089 – 1102.

[8] G.Y. Chan, H. M. Khan, F. F. Chua, "Resource Scalability as Preventive and Remedial Measures for Cloud Service Violation", *International Conference on Future Internet of Things and Cloud*, 2018, pp. 428-435.

[9] M. F. Alii, O. A. Batarfil, "Abul Bashar. "A simulation-based comparative study of Cloud Datacenter scalability, robustness and complexity", *IEEE International Conference on Intelligent Computing and Information Systems*, 2015, pp. 547-551.

[10] R. Aluvalu, L. Muddana, "Access control model with enhanced flexibility and scalability for cloud", *International Conference on Green Computing and Internet of Things*, 2015, pp. 355-359.

[11] R. Lovas, E. Nagy, J. Kovács, "Cloud agnostic Big Data platform focusing on scalability and cost-efficiency", *Advances in Engineering Software*, Vol. 125, 2018, pp. 167-177.

[12] J. Ward, S. Barker, "Cloud cover : monitoring large-scale clouds with Varanus", *Journal of Cloud Computing: Advances, Systems and Applications*, 2015, Vol. 4, No.1, pp. 1-28.

[13] O. Knodel, G. Paul, R. Spallek, G. Rainer, "Virtualizing Reconfigurable Hardware to Provide Scalability in Cloud Architectures", *International Conference on Advances in Circuits, Electronics and Micro-electronics,* 2017, pp. 32-38.

[14] J. He, M. Dong, K. Ota, M. Fan, et al, "NetSecCC: A scalable and fault-tolerant architecture for cloud computing security", *Peer-to-Peer Networking and Applications*, Vol. 9,No. 1, 2014, pp.67-81.

[15] V. Chavan, P. R. Kaveri, "Clustered virtual machines for higher availability of resources with improved scalability in cloud computing", *International Conference on Networks & Soft Computing*, 2014, pp. 221-225.

[16] A. Paya, D. C. Marinescu, "Energy-aware Load Balancing and Application Scaling for the Cloud Ecosystem", *IEEE Transactions on Cloud Computing*, Vol. 5, No. 1, 2017, pp.15-27.

[17] A. M. Caulfield, E. S. Chung, A. Putnam, et al, "A cloud-scale acceleration architecture", *IEEE/ACM International Symposium on Microarchitectur*, 2016, pp. 1-13.

[18] M. Kaiiali, "Designing a VM-level vertical scalability service in current cloud platforms: a new hope for wearable computers", *Turkish Journal of Electrical Engineering & Computer Sciences*, Vol. 25, No. 4, 2017, pp. 2555- 2566.

[19] T. Goldschmidt, A. Jansen, H. Koziolek, J. Doppelhamer, H. P. Breivold, "Scalability and Robustness of Time-Series Databases for Cloud-Native Monitoring of Industrial Processes", *IEEE International Conference on Cloud Computing*, 2014, pp. 602-609.

[20] D. Moldovan, H. L. Truong, S. Dustdar, "Cost-Aware Scalability of Applications in

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Public Clouds", *IEEE International Conference on Cloud Engineering*, 2016, pp. 79-88.

[21] M. D. Assunc, C. H. Cardonha, M. A. S. Netto, R. L. F. Cunha, "Impact of user patience on auto-scaling resource capacity for cloud services", *Future Generation Computer Systems*, Vol. 55, No. C, 2016, pp. 41-50.

[22] Y. Zhao, Y. Li, I. Raicu, S. Luc, W. Tian, H. Liu, "Enabling scalable scientific workflow management in the Cloud", *Future Generation Computer Systems*, Vol. 46, 2015, pp. 3-16.

[23] M. Fallah, M. G. Arani, M. Maeen, "NASLA: Novel Auto Scaling Approach based on Learning Automata for Web Application in Cloud Computing Environment", *International Journal of Computer Applications*, Vol. 113, No. 2, 2015, pp. 18-23.

[24] A. Bashar, "Autonomic scaling of Cloud Computing resources using BN-based prediction models", *IEEE International Conference on Cloud Networking*, 2013, pp. 200-204.

[25] J. Shan, W. Jia, X. Ding, "Rethinking Multicore Application Scalability on Big Virtual Machines", *IEEE International Conference on Parallel and Distributed Systems*, 2017, pp. 694-701.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 1 The Simulated Allocation Results

Cloudlet ID	STATUS	Data center ID	VM ID	Time	Start Time	Finish Time
10	SUCCESS	4	10	2	0.5	2.5
30	SUCCESS	4	10	2	0.5	2.5
11	SUCCESS	4	11	2	0.5	2.5
31	SUCCESS	4	11	2	0.5	2.5
14	SUCCESS	5	14	2	0.5	2.5
34	SUCCESS	5	14	2	0.5	2.5
15	SUCCESS	5	15	2	0.5	2.5
35	SUCCESS	5	15	2	0.5	2.5
12	SUCCESS	5	12	2	0.5	2.5
32	SUCCESS	5	12	2	0.5	2.5
13	SUCCESS	5	13	2	0.5	2.5
33	SUCCESS	5	13	2	0.5	2.5
18	SUCCESS	6	18	2	0.5	2.5
38	SUCCESS	6	18	2	0.5	2.5
19	SUCCESS	6	19	2	0.5	2.5
39	SUCCESS	6	19	2	0.5	2.5
16	SUCCESS	6	16	2	0.5	2.5
36	SUCCESS	6	16	2	0.5	2.5
17	SUCCESS	6	17	2	0.5	2.5
37	SUCCESS	6	17	2	0.5	2.5
2	SUCCESS	2	2	3	0.5	3.5
22	SUCCESS	2	2	3	0.5	3.5
42	SUCCESS	2	2	3	0.5	3.5
3	SUCCESS	2	3	3	0.5	3.5
23	SUCCESS	2	3	3	0.5	3.5
43	SUCCESS	2	3	3	0.5	3.5
0	SUCCESS	2	0	3	0.5	3.5
20	SUCCESS	2	0	3	0.5	3.5
40	SUCCESS	2	0	3	0.5	3.5
1	SUCCESS	2	1	3	0.5	3.5
21	SUCCESS	2	1	3	0.5	3.5
41	SUCCESS	2	1	3	0.5	3.5
6	SUCCESS	3	6	3	0.5	3.5
26	SUCCESS	3	6	3	0.5	3.5
46	SUCCESS	3	6	3	0.5	3.5
7	SUCCESS	3	7	3	0.5	3.5
27	SUCCESS	3	7	3	0.5	3.5
47	SUCCESS	3	7	3	0.5	3.5
4	SUCCESS	3	4	3	0.5	3.5
24	SUCCESS	3	4	3	0.5	3.5
44	SUCCESS	3	4	3	0.5	3.5
5	SUCCESS	3	5	3	0.5	3.5
25	SUCCESS	3	5	3	0.5	3.5
45	SUCCESS	3	5	3	0.5	3.5
8	SUCCESS	4	8	3	0.5	3.5
28	SUCCESS	4	8	3	0.5	3.5
48	SUCCESS	4	8	3	0.5	3.5
9	SUCCESS	4	9	3	0.5	3.5
29	SUCCESS	4	9	3	0.5	3.5
49	SUCCESS	4	9	3	0.5	3.5