



A Study on Simulation Tools in Cloud Computing

K. Arun Kumar^{*1} **Dr. R.P.Ram Kumar**^{*2} **Kunchala Little Flower**^{*3}

^{*1}Assistant Professor Department of Computer Science & Engineering, MREC-(A)

^{*2}Professor Department of Computer Science & Engineering, MREC-(A)

^{*3}Assistant Professor Department of Computer Science & Engineering, KL University(AP)

Abstract—today, distributed computing has turned into a promising worldview that goes for conveying figuring assets and administrations on request. The reception of these administrations has been quickly expanding. One of the fundamental issues in this setting is the manner by which to assess the capacity of cloud frameworks to give the coveted administrations while regarding the QoS requirements. Experimentation in a genuine situation is a difficult issue. Truth be told, the monetary cost and the time required are high. Likewise, the examinations are not repeatable, on the grounds that various factors that are not under control of the analyzer may influence exploratory outcomes. Accordingly, utilizing reenactment structures to assess cloud applications is favored. This paper introduces a study of the current simulation instruments in distributed computing. It gives likewise a basic and similar investigation of the considered instruments. At last, it emerges a noteworthy test to be tended to for additionally inquire about.

Keywords—Cloud computing, simulation tools, comparative analysis.

I. INTRODUCTION

Distributed computing has risen as another innovation which gives a lot of processing and information stockpiling ability to its clients. It expects to defeat numerous issues emerging from the fast development of endeavors and the development of their information. Truth be told, the open space for the capacity of data on a PC can't meet the present needs. What's more, upkeep expenses of equipment have expanded [2].

At present, cloud conditions are making utilization of virtualization advancements. Rather than running projects on PCs, everything is facilitated in the cloud. The last is a model empowering an on-request organize access to a mutual pool of configurable figuring assets (stockpiling, applications, administrations, and so forth.). It offers benefits that can be ordered in three classes [21]: programming as an administration (SaaS), stage as an administration (PaaS) and foundation as an

administration (IaaS). The clients can expend these administrations in view of a Service Level Agreement (SLA) which characterizes their required Quality of Service (QoS) parameters, on a "pay-as-you-go" premise.

In recent years, the number of companies moving into the cloud has increased considerably. Then, it is necessary to evaluate the performance levels of cloud systems. The experimentation in a real environment is not advisable. In fact, the deployment of a cloud system generally requires the use of many hardware resources, network resources, storage resources, etc. In addition, with a real cloud system, the evaluation of some critical scenarios and failure is difficult to achieve. Also, repeating experiments are impossible. Moreover, performing experiments with a real cloud system needs to have certain knowledge of networking fundamentals, cloud resource management, cloud security, etc. Furthermore, the financial cost and time required by these experiments are often very high.

At that point, a more feasible answer for beat these issues is to utilize reenactment instruments that can assess the execution of cloud applications before conveying them in a genuine setup. The simulation innovation has turned out to be notable in cloud industry and foundation. It offers a free domain that can copy the conduct of a genuine cloud condition [4]. Actually, utilizing a reenactment approach, investigations can be effectively controlled and rehashed. In fact, they require less push to plan and test explore situations. Not at all like a genuine situation, tests utilizing a reproduction instrument can be imitated rapidly and the outcomes can be replicated effortlessly.

By the by, choosing the correct test system to utilize needs an intensive investigation of the accessible devices. Past endeavors to overview recreation instruments for distributed computing can be found in the writing [22]. In this overview paper, we endeavor to offer a refreshed perspective of this theme. We give an outline of the current reenactment apparatuses in the cloud. Additionally, we display a



profound near examination of these instruments in light of various properties. Besides, the paper presents another test that must be tended to.

The remainder of this paper is structured as follows. In section II, we give a description of the existing cloud simulation tools. In section III, we present a rich discussion and a comparative study of these simulators based on various criteria. Finally, the last section concludes and highlights an emerging research challenge to address in our future work..

II. OVERVIEW ON THE EXISTING CLOUD SIMULATION TOOLS

In this segment, we exhibit an itemized investigation of cloud test systems proposed in the writing. We start by exhibiting the prominent test system CloudSim and its expansions. From that point onward, we present alternate test systems.

A. CloudSim and its expansions :

CloudSim: CloudSim [6] is an outstanding cloud computing test system based upon GridSim. It has been created in the CLOUDS Laboratory at the University of Melbourne. It gives a toolbox to displaying and mimicking the conduct of many cloud parts, for example, virtual machines (VMs), server farms and asset provisioning administrations. Additionally, it can speak to various sorts of mists (open, private, crossover and multi-cloud conditions). CloudSim is an occasion driven recreation apparatus, in other words, all segments of the reenactment keep up a message line and produce messages, which they go along to other en-tities. It can instantiate numerous server farms which comprises of capacity servers and physical host machines. These machines have numerous VMs executing a few assignments (named cloudlets). CloudSim can perform reproductions of allotting and executing a workload on a cloud framework [7]. The correspondence stream among center CloudSim substances is appeared in Figure 1:

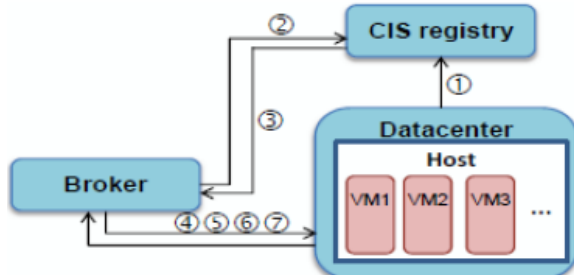


Fig. 1. Flow of communication among CloudSim entities

1) At the beginning of a simulation, each data center entity registers its information in the Cloud Information Service (CIS) registry.

2) The data center broker (DCB) acting on behalf of users queries the CIS registry for the information of data centers.

3) The CIS registry responds by sending a list of the available data centers.

4) The DCB requests the characteristics of the available data centers (DC).

5) The DCB asks the concerned DC to create the required VMs.

6) Once the VMs are created, the DCB sends cloudlets to DC for execution. If cloudlets finish their execution, a message is sent to the DCB.

7) At the end of the simulation, the VMs will be destroyed. CloudSim enables modeling and simulation of a large scale infrastructure. In addition, it is easily extensible. In fact, it helps to develop new scheduling algorithms and resource distribution policies.

Extensions: Despite its focal points, CloudSim has a few constraints. Along these lines, many endeavors have been made to add to it new functionalities in light of various necessities. We refer to a while later some of these expansions.

- **Network CloudSim:** It is an expansion of CloudSim test system proposed by Garg et al. [11]. It underpins the demonstrating of summed up applications, for example, work processes, internet business and web applications. Additionally, it acquaints new ideas with demonstrate an inward system of a server farm. It has two levels of planning. The first is at the host level. It comprises in planning assignments on VMs. The other is at the VM level where genuine applications are executed.

- **Federated CloudSim:** It is a structure which stretches out CloudSim to reenact a few unified cloud situations [19]. In addition, it includes bundles for testing SLAaware booking calculations.

- **Dynamic CloudSim:** It is a test system based on CloudSim. It acquaints models with catch inhomogeneity in the execution of computational assets, vulnerability and dynamic changes of VMs execution and also disappointment amid undertaking execution [3].

- **Teach Cloud:** CloudSim does not have a graphical UI (GUI) which enables the figuring out how to procedure of the understudies. Along these lines, another expansion called "TeachCloud" was created [13]. It fills in as an instructive toolbox, utilized by understudies to lead analyzes in a cloud figuring framework. It empowers assembling and executing redid arrange topologies. Additionally, it contains



new modules identified with SLA limitations and business process administration (BPM).

- **FTCloud Sim:** It presents new improvements over the CloudSim fundamental stage to display and reenact the dependability upgrade components [33]. It gives an extensible interface which causes specialists to actualize new components effortlessly. It includes new modules that can trigger disappointment occasions to test the execution of every instrument.

- **Elastic Sim:** It is another expansion of CloudSim test system for work process applications [5]. It plans to assess the execution of booking and asset provisioning calculations. It underpins asset runtime auto-scaling and displaying of stochastic assignment execution time. Moreover, it offers a GUI which represents the booking comes about. This GUI finds the points of interest and downsides of the proposed calculations.

- **Cloud Analyst:** It is a test system in light of CloudSim [32]. It plans to assess the execution of largescale disseminated applications on cloud. These applications can have high workloads that are topographically dispersed over various server farms. Also, CloudAnalyst offers a GUI with a specific end goal to arrange any topographical dispersed framework, for example, setting equipment parameters. It produces recreation brings about the type of outlines and tables.

- **Cloud Reports:** It presents numerous improvements over CloudSim's structure [29]. It gives a GUI which offers numerous functionalities. Right off the bat, it permits torun numerous reenactments in the meantime. Also, it can create reports with itemized data and fare recreation information. These data are identified with asset use costs, vitality utilization, execution time, and so on.

- **CEPSim:** It is a test system of cloud-based Complex Event Processing (CEP) frameworks [12]. It expands CloudSim with an application show in view of coordinated non-cyclic diagrams (DAGs) which is utilized to speak to persistent questions which process quick floods of information.

The execution of these inquiries can be mimicked in many sorts of cloud condition including open, private and multi-cloud situations. Moreover, this apparatus can be redone by making different booking procedures and administrator arrangement.

- **CDOSim:** This [10] instrument is an augmentation of CloudSim test system which can recreate the SLA infringement, reaction times and expenses of a CDO (cloud sending alternative). It can reenact application models that take after the Knowledge Discovery Meta-Model (KDM).

Likewise, it empowers cloud clients to think about the cost and proficiency of a cloud arrangement with those of alternate arrangements.

- **CloudSimEx:** The significant commitment of CloudSimEx [16], is the expansion of CloudSim to mimic MapReduce applications. The last are displayed as an occupation made out of "outline" and "diminish" assignments. Likewise, it can run various examinations in parallel, in various JVM forms. CloudSimEx acquaints different highlights related with online frameworks, movement Latency issues, and so on.

B. Green Cloud:

It is an open-source apparatus which is intended for recreating a server farm in distributed computing [18]. It is an extensionof the outstanding test system in PC organizing, NS-2. GreenCloud catches points of interest of the vitality devoured by the parts of server farm (switches, servers and connections) and also parcel level correspondence designs in reasonable setups. It can likewise break down the heap circulation through the system.

C. Cloud Sched:

It is another reproduction device proposed by Tian et al. [30]. It gives a stage to displaying and assessing the execution of a few planning arrangements in IaaS layer. In addition, it offers graphical and literary yields after recreation.

D. MDC Sim:

It is a discrete occasion test system created at the Pennsylvania State University by Lim et al. [20]. It empowers clients to demonstrate particular equipment qualities of various segments of a server farm like servers, correspondence connects and switches. The entire reenactment demonstrates is designed in three layers: a correspondence layer, a portion layer and a client level layer. The point is to model and copy genuine stack from the correspondence conventions to the applications.

E. iCan Cloud:

Nunez et al. [23] have built up the recreation stage "iCanCloud". This test system can foresee the exchange offs amongst cost and execution of a given arrangement of utilizations executed in a particular arrangement. It gives a GUI to planning and running the analyses. Besides, it permits parallel execution of one trial more than a few machines. Additionally, it bolsters reproduction of combined cloud conditions, that contain internetwork esources from both open and private areas.



F. *secCloud Sim*:

It is based over iCanCloud test system [25]. It gives the essential highlights of security, for example, validation and approval. The proposed security layer comprises of two modules.

The first permits to verify clients and utilizations administrations of cloud in a recreated domain. The second module characterizes authorization sets of rights which contrast starting with one client then onto the next as indicated by their necessity.

G. *GroudSim*:

It is a test system created by Ostermann et al. [24] at the University of Innsbruck. It expects to recreate the execution of logical applications in a computational network or cloud. It concentrates on IaaS administration and it can be stretched out to help extra models. Additionally, it gives a few highlights to reproducing complex situations.

H. *DC Sim*:

It is a reenactment structure which enables a large number to mimic virtualized server farms conveyed on IaaS cloud [31]. It bolsters sharing of workload between numerous VMs that are running multi-level applications. Besides, it can assess control utilization and SLA infringement of server farm administration framework. DCSim can be effectively stretched out to execute new highlights and functionalities.

I. *Sim IC*:

It is a discrete occasion test system in light of the SimJava bundle. It expects to recreate a between cloud office where various mists team up [27]. What's more, it bolsters reproduction of heterogeneous and processing situations which are liable to ongoing imperatives. Besides, it can recreate a few topologies and substances for IoT (Internet of Things) situations.

J. *SPECI*:

It is a reproduction apparatus which can anticipate the conduct and execution of vast server farms [28]. It additionally expects to test disappointment and recuperation instruments. SPECI is made out of two bundles. The first is committed to the server farm format and topology. The second one contains mimicking parts for performing tests.

K. *PICS*:

It is a recreation toolbox that intends to assess the cost and execution of different open IaaS arrangements [17]. Actually, it offers abilities to assess diverse sorts of assets, charging models and execution vulnerability. What's more, it can reproduce numerous arrangements of assets

administration, for example, the level and vertical cloud asset scaling, work booking strategies, and so forth.

III. DISCUSSION

All through this review paper, we set forward a delegate outline of the real endeavors of recreation devices accessible in distributed computing. One of the fundamental issues of specialists is to pick the satisfactory test system for their examination. Indeed, there are cloud test systems which are expected for a particular reason. Table I exhibits an endeavor to think about the already portrayed test systems in light of various criteria. In the accompanying, we depict the characteristics on which we have played out the similar investigation.

- **Platform:** It speaks to the fundamental stage used to actualize the test system. As delineated in the table, around 57% of the test systems have been stretched out from the outstanding test system CloudSim.

- **Language:** The dialects used to execute the test systems are identified with the stages. A large portion of the existing test systems utilize Java dialect. Others are composed in C++. Additionally, there are a few instruments which utilize mixes of dialects like GreenCloud and MDSCSim.

- **Availability:** This model demonstrates whether an apparatus is business or open source and accessible to download. We take note of that 79% of the recreation apparatuses outlined in the table are open source. The remaining are not accessible or business.

- **Graphical help:** The accessibility of GUI for test systems encourages clients to play out the reenactments in a basic and proficient way. It permits to characterize recreation parameters, for example, the administration of VMs. Additionally, by utilizing a graphical interface, clients can execute or drop reenactment and create graphical or literary reports. As showed in the table, the first CloudSim did not bolster any graphical interface whatsoever. At that point, numerous augmentations have been proposed to give GUI highlights, for example, CloudAnalyst and CloudReports.

- **Communication demonstrate:** It decides the correspondence way between elements of the test system. For instance, GreenCloud has a full correspondence demonstrate. Truth be told, it actualizes a full TCP/IP convention reference demonstrate. In any case, CloudSim has a restricted correspondence display which does not bolster TCP/IP. The correspondence between and inside server farms happens by message passing.

- **Correspondence demonstrate:** It decides the correspondence way between elements of the test



system. For instance, GreenCloud has a full correspondence demonstrate. Truth be told, it actualizes a full TCP/IP convention reference show. Be that as it may, CloudSim has a restricted correspondence show which does not bolster TCP/IP. The correspondence between and inside server farms happens by message passing.

Vitality demonstrate: This measure lets us know whether the test system enables clients to demonstrate the vitality or not. A vitality utilization display expects to think about the effectiveness of planning calculations regarding vitality.

A few test systems perform just harsh estimation on control utilization (like MDCSim). Organization show: A cloud league intends to run cloud applications on heterogeneous mists. The help of organization show implies whether a test system enables clients to demonstrate combined cloud applications SLA bolster: This paradigm decides whether the test system can guarantee the necessities expressed in the SLA. In view of SLA parameters, asked for assets are allocated to clients. A few test systems can reproduce SLA infringement, for example, DCSim, TeachCloud and Green- Cloud.

Cost show: This quality lets us know whether a recreation apparatus has a module to display costs and decide the cost of the utilized administrations. Distributed computing receives pay-as-you-go demonstrate where clients are charged based on their utilization.

Parallel tests: The model "Parallel examinations" implies the capacity to consolidate more than one machine to cooperate keeping in mind the end goal to process assignments

IV. CONCLUSION AND RESEARCH CHALLENGES

To outline, in this paper we have examined and analyzed the well known cloud test systems in the writing. Based on this examination and assessment, we have called attention to the failure of the current test systems to manage changes of applications at runtime. Such necessities require another exploration commitment.

In our continuous work, we plan to beat this impeding of existing test systems. We intend to give an application demonstrate which portrays dynamic changes amid execution in view of a few principles. The difference in application display amid execution essentially needs to upgrade a current test system. will depend on our past work [9] which concentrate on work processes which have turned into a successful route to the advancement of logical applications.

REFERENCES

- [1] M. Adams, "Dynamic Workflow," in *Modern Business Process Automation*. Springer, 2010, pp. 123–145.
- [2] V. Arutyunov, "Cloud computing: Its history of development, modern state, and future considerations," *Scientific and Technical Information Processing*, vol. 39, pp. 173–178, 2012.
- [3] M. Bux and U. Leser, "DynamicCloudSim: Simulating heterogeneity in computational clouds," *Future Generation Computer Systems*, vol. 46, pp. 85 – 99, 2015.
- [4] R. Buyya, R. Ranjan, and R. N. Calheiros, "Modeling and simulation of scalable Cloud computing environments and the CloudSim toolkit: Challenges and opportunities," in *HPCS*. Leipzig, Germany: IEEE, June 21-24 2009, pp. 1–11.
- [5] Z. Cai, Q. Li, and X. Li, "ElasticSim: A toolkit for simulating workflows with cloud resource runtime auto-scaling and stochastic task execution times," *Journal of Grid Computing*, pp. 1–16, 2016.
- [6] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. F. De Rose, and R. Buyya, "CloudSim: A Toolkit for Modeling and Simulation of Cloud Computing Environments and Evaluation of Resource Provisioning Algorithms," *Software: Practice and Experience*, vol. 41, no. 1, pp. 23–50, Jan. 2011.
- [7] R. N. Calheiros, R. Ranjan, C. A. F. D. Rose, and R. Buyya, "Cloudsim: A novel framework for modeling and simulation of cloud computing infrastructures and services," *Grid Computing and Distributed Systems Laboratory, Tech. Rep.*, 2009.
- [8] W. Chen and E. Deelman, "WorkflowSim: A Toolkit for Simulating Scientific Workflows in Distributed Environments," in *e-Science*. Washington, DC, USA: IEEE Computer Society, 2012, pp. 1–8.
- [9] F. Fakhfakh, H. H. Kacem, and A. H. Kacem, "A provisioning approach of cloud resources for dynamic workflows," in *CLOUD*. IEEE, 2015, pp. 469–476.
- [10] F. Fittkau, S. Frey, and W. Hasselbring, "Cdosim: Simulating cloud deployment options for software migration support," in *MESOCA*, Trento, Italy, September 24 2012, pp. 37–46.
- [11] S. K. Garg and R. Buyya, "NetworkCloudSim: Modelling Parallel Applications in Cloud Simulations," in *UCC*, 2011, pp. 105–113.
- [12] W. A. Higashino, M. A. M. Capretz, and L. F. Bittencourt, "Cepsim: A simulator for cloud-based complex event processing," in *International Congress on Big Data*. New York City, USA: IEEE, June 27 - July 2 2015, pp. 182–190.
- [13] Y. Jararweh, Z. Alshara, M. Jarrah, M. Kharbutli, and M. N. Alsaleh, "Teachcloud: a cloud computing educational toolkit," *IJCC*, vol. 2, no. 2/3, pp. 237–257, 2013.
- [14] Y. Jararweh, M. Jarrah, M. Kharbutli, Z. Alshara, M. N. Alsaleh, and M. Al-Ayyoub, "Cloudexp: A comprehensive cloud computing experimental framework," *Simulation Modelling Practice and Theory*, vol. 49, pp. 180–192, 2014.
- [15] J. Jung and H. Kim, "MR-CloudSim: Designing and implementing MapReduce computing model on CloudSim," in *ICTC*, October 2012, pp. 504–509.
- [16] P. Kathiravelu, "An elastic middleware platform for concurrent and distributed cloud and mapreduce simulations," *CoRR*, vol. abs/1601.03980, 2016.
- [17] I. K. Kim, W. Wang, and M. Humphrey, "PICS: A public iaaS cloud simulator," in *CLOUD*. New York City, USA: IEEE, June 27 - July 2 2015, pp. 211–220.



National Conference on Convergence of Emerging Technologies in Computer Science & Engineering (CETCSE-2K17)

- [18] D. Kliazovich, P. Bouvry, and S. Khan, "GreenCloud: a packet-level simulator of energy-aware cloud computing data centers," *Supercomputing*, vol. 62, no. 3, pp. 1263–1283, 2012.
- [19] A. Kohne, M. Spohr, L. Nagel, and O. Spinczyk, "FederatedCloudSim: a SLA-aware federated cloud simulation framework," in *Middleware*. Bordeaux: ACM, 2014, pp. 1–5.
- [20] S. Lim, B. Sharma, G. Nam, E. Kim, and C. R. Das, "MDCSim: A multi-tier data center simulation platform," in *International Conference on Cluster Computing*, New Orleans, Louisiana, USA, August 31–September 4 2009, pp. 1–9.
- [21] P. Mell and T. Grance, "The NIST Definition of Cloud Computing," NIST, Gaithersburg, MD, Tech. Rep. 800-145, September 2011.
- [22] P. J. Mudialba, "A Study on the Fundamental Properties, Features and Usage of Cloud Simulators," in *PlatCon*. IEEE, 2016, pp. 1–5.
- [23] A. Nez, J. Vazquez-Poletti, A. Caminero, G. Casta, J. Carretero, and I. Llorente, "iCanCloud: A Flexible and Scalable Cloud Infrastructure Simulator," *Journal of Grid Computing*, vol. 10, no. 1, pp. 185–209, 2012.
- [24] S. Ostermann, K. Plankensteiner, R. Prodan, and T. Fahringer, "Groudsim: An event-based simulation framework for computational grids and clouds," in *Euro-Par*. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 305–313.
- [25] U. U. Rehman, A. Ali, and Z. Anwar, "seccloudsim: Secure cloud simulator," in *FIT*, December 2014, pp. 208–213.
- [26] J. Son, A. V. Dastjerdi, R. N. Calheiros, X. Ji, Y. Yoon, and R. Buyya, "CloudSimSDN: Modeling and Simulation of Software-Defined Cloud Data Centers," in *CCGrid*, 2015, pp. 475–484.
- [27] S. Sotiriadis, N. Bessis, N. Antonopoulos, and A. Anjum, "Simic: Designing a new inter-cloud simulation platform for integrating large-scale resource management," in *AINA*, March 2013, pp. 90–97.
- [28] I. Sriram, "Speci, a simulation tool exploring cloud-scale data centres," in *CloudCom*. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 381–392.
- [29] T. Teixeira Sa, N. R. Calheiros, and G. D. Gomes, *CloudReports: An Extensible Simulation Tool for Energy-Aware Cloud Computing Environments*. Springer International Publishing, 2014, pp. 127–142.
- [30] W. Tian, Y. Zhao, M. Xu, Y. Zhong, and X. Sun, "A toolkit for modeling and simulation of real-time virtual machine allocation in a cloud data center," *IEEE Transactions on Automation Science and Engineering*, vol. 12, no. 1, pp. 153–161, January 2015.
- [31] . Tighe, G. Keller, J. Shamy, M. Bauer, and H. Lutfiyya, "Towards an improved data centre simulation with dcsim," in *CNSM*, October 2013, pp. 364–372.
- [32] B. Wickremasinghe, R. N. Calheiros, and R. Buyya, "Cloudanalyst: A cloudsim-based visual modeller for analysing cloud computing environments and applications," in *AINA*, 2010, pp. 446–452.
- [33] A. Zhou, S. Wang, C. Yang, L. Sun, Q. Sun, and F. Yang, "FTCloudSim: support for cloud service reliability enhancement simulation," *Journal of Web and Grid Services*, vol. 11, no. 4, pp. 347–361, 2015.

