International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-8 Issue-12, October 2019

Testing Methodologies for Cloud Performance

Abey Jacob, Cyril Raj

Abstract—Cloud environment basically offers Software as a Service (SaaS), Infrastructure as a Service (IaaS), and Platform as a Service (PaaS). Here we describe the testing process employed for performance testing. Though new tools for testing cloud are emerging into the market, there are aspects which are suited for manual testing and some which can be speeded up using automatic testing tools. This paper brings out the techniques best suited to test different features of Cloud computing environment. The authors also try to bring out (recommend) broad guidelines to follow while setting up a cloud environment to reduce the number of bugs in the system.

Index Terms—Cloud Computing, Testing tools.

I. CLOUD COMPUTING

Cloud Computing is becoming center stage with its promise to provide the means for BigData storage and analysis
The National Institute of Standards and Technology (NIST), created under US tof Commerce defined Cloud computing as a model for enabling ubiquitous, easy, on-demand network access to a pool of configurable and shared computing resources (e.g., services, networks, applications, storage, and servers) which can be quickly stitched and offered without much effort or association with supplier. Mainly cloud offerings have five features (Self-service by on-demand basis, High Bandwidth access to network, sharing of resources, High elasticity, and Steady performance). The commonly available three cloud offerings are (Infrastructure as a Service, Software as a Service, Platform as a Service) and four installation types (viz, Public cloud, Hybrid cloud, Private cloud and Community cloud).
From implementation perspective, Cloud computing is a pool of network connected resources in which large groups of remote servers are interconnected for processing the data on sharing basis, data retention at a common places, and accessing the resources by online. [http://en.wikipedia. org/wiki/Cloud_computing]. Cloud consists of virtualized data centers and network interconnectivity infrastructure. Hence 'Testing a Cloud Computing Environment' implied to test the fundamental components which inhabit the data center for security, functionality, and performance, accompanied with data center testing and complete services [Testing the Cloud: Definitions, Requirements, and Solutions, Whitepaper, IXIA, August 2011].

Revised Manuscript Received on October 30, 2019.
***** Correspondence Author
JACOB&, Principal technical Officer with Centre for Development of Advanced computing (CDAC), Bangalore - India.
Dr. V CYRIL RAJ, Join Registrar (E&T) at Dr.M.G.R University, Chennai - India.

In other words, we need to basically test the routers, switches, gateways, firewalls, storage systems, virtual hosts and also the cloud specific factors such as ramp-up time, elasticity, quality-of-service etc.
Cloud is basically delivery of services over the internet. So in addition to the tests required for web services, we need to perform tests related to Infrastructure / Platform / Network/Software as mandated by the system chosen by the end user.

A. Cloud

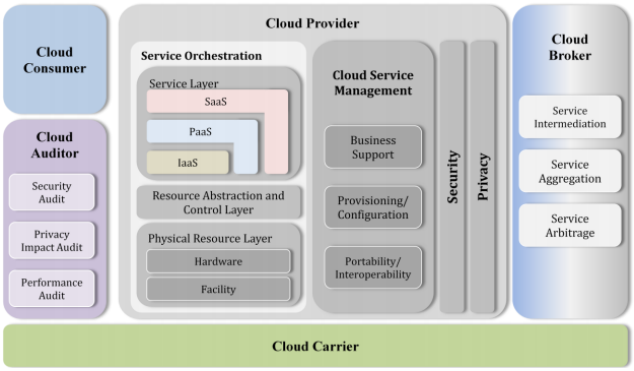
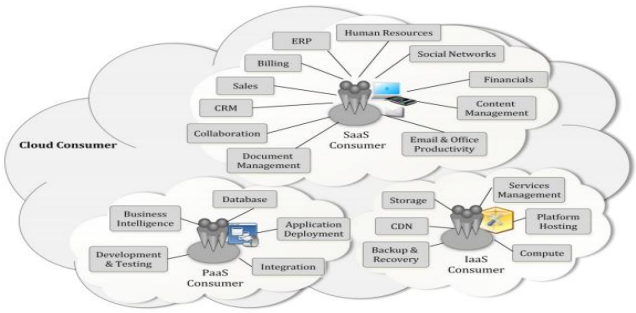


Fig 1.NIST Cloud reference Architecture

Five major actors are defined in Figure 1 [2] viz., cloud broker, cloud carrier, cloud end user, cloud service supplier, and cloud auditor. Here everyone is an associate degree operator (an individual or associate degree organization) which engages during dealings/method and/or performs jobs in cloud computing. The table below briefs the list of actors outlined with this cloud computing reference design.

Actor	Definition
Cloud Consumer	A person or organization that maintains a business relationship with, and uses service from, *Cloud Providers*.
Cloud Provider	A person, organization, or entity responsible for making a service available to interested parties.
Cloud Auditor	A party that can conduct independent assessment of cloud services, information system operations, performance and security of the cloud implementation.
Cloud Broker	An entity that manages the use, performance and delivery of cloud services, and negotiates relationships between *Cloud Providers* and *Cloud Consumers*.
Cloud Carrier	An intermediary that provides connectivity and transport of cloud services from *Cloud Providers* to *Cloud Consumers*.

B Type of Services



Based on types of usage, cloud offers [3] Platform as a Service (PaaS), Software as a Service (SaaS) and Infrastructure as a Service (IaaS).
Consumer is principal stakeholder of the cloud computing service and deputize as a member or institute that engaged as a professional connection with and utilizing their service through a cloud vendor. Based on the Service Level Agreement (SLA), users request the appropriate service from service giver and utilize the service. This SLA specifies the technical performance requirements like standard of service, security, corrective measures for performance failures etc which has to be fulfilled by a cloud provider.

II. CLOUD TESTING

Individual components at each layer need to be tested; and then Integration Testing should be conducted to ensure quality of the Cloud computing system.
Specifically with respect to Cloud computing environment, testing should be carried out for cloud-specific factors such as elasticity. The SPEC Open Systems Group (OSG) Cloud working group has been investigating application workloads suitable for benchmarking. It lists four key metrics relevant in the context of cloud benchmarking. The main aspects to be tested in Cloud Computing environment from the users view point include the following

- Elasticity
- Provisioning interval, or the lag between when a resource is requested and when it is actually available.
- Agility or the ability of the provider to track the needs of the workload.
- Scale-up or the improvement in response times with increased amount of resources.
- Elastic speed-up or the improvement in performance of tasks as an additional resource is added on real time basis.
  - Throughput
  - Response time
  - Variability

Describe the various items that required to be tested in our cloud computing.

- Clear Box
- Behavioral
- Functional
- Load
- Performance
- Network

Network interconnectivity infrastructure level comprises of the routers, switches, gateways and firewalls. Table1 gives the component, parameter to be tested and available testing tools.

III. DEFINITION OF PERFORMANCE TESTING

Various illustration and explanation of performance testing exist. In IT industry, performance testing can be summarized as:

- The technical examination to decide or to prove the load handling capability, speed, dynamicity, and / or steadiness in performance of the system under test
- Evaluations used to figure out bottlenecks or performance issues or successfulness of a system under specific load or operational environments.
- Quantitative (e.g. load testing) or qualitative (e.g. reliability, scalability) testing.

Normally Performance testing is executed in closed situations, such as scientific lab or in a duplicate or simulation of a production environment. Performance testing is also bothered with identifying performance issues, and forms part of the process of optimizing and adjusting to attain the required performance target.
The basic function of performance testing is to minimize the level of failure connected with system performance risks. The main objective of performance testing is the controlling of peril and this type plan of actions are highly pointed towards controlling the perils.

IV. FORMS OF PERFORMANCE TESTING

The different forms of performance testing conducted are: Load-, tortune-, spike-, endurance /soak-, scalability-, configuration- and isolation testing.

BENEFITS OF PERFORMANCE TEST

One of the main principles of performance testing is to eradicate business risk by identifying performance issues in key business systems before going live.
In this project we have illustrated actual benefits by identifying the soft limits and configuration settings which otherwise would not have been picked up during functional testing as user load was required to identify these configuration limits. This exposed a prominent potential problem if utilizing the full 100 Mb network connections wherein the response time was slow which in turn resulted in application to time-out (fail).
Since the system is used by outsiders responding to vacancies, hence if they were undetected the reputational damage would have been very massive.

V. PERFORMANCE CHALLENGES IN CLOUD

Migrating various services to the cloud assure to render with more agility at a lower cost − but at the same time with greater complexity and many risks associated along the way. Given below are the five critical hurdles and how to over the same when operating a private or hybrid cloud.

1. FINDING SUITABLE APPLICATIONS FOR CLOUD AND PLANING FOR A FRUITFUL MIGRATION?

Every application is not suitable for the cloud. One part of an associate in nursing application is cloud-ready whereas different elements are not. We need to spot the best appropriate applications and associated elements for migration, figure out main issues such as chattiness and latency which are amplified in the cloud, and create a performance baseline that you just will take a look at against once migration. With a transparent image of service dependencies and infrastructure usage, we will be able to produce a list which will guarantee an entire and successful migration.

2. PERFORMANCE – HOW TO SHOOT THE SERVER-RELATED ROOT CAUSES WHEN PERFORMANCE ISSUES CROP UP? IF YOU DON'T KNOW THE PHYSICAL SERVERS ON WHICH YOUR APPLICATION IS RUNNING ON.

In dedicated domain, we typically use infrastructure metrics and events to figure out performance problems. But concluding the application performance from tier-based statistics becomes challenging – if not impossible – when applications shared the allocated physical resources dynamically. In order to manage application performance within the cloud, you need a real-time topological map of service delivery across all tiers. Since the landscape is usually dynamic, it's necessary that the dependency map is dynamically created and mechanically updated for each single dealings and repair instance.

3. CHARGEBACK – HOW TO KNOW HOW MUCH CPU YOUR APPLICATION IS CONSUMING IN ORDER TO CHOOSE AN APPROPRIATE CHARGEBACK MODEL OR VERIFY YOUR BILLS?

IT needs a new standard for computing the resource consumption in order to transition from a resource-focused cost-center to a business-service-focused profit-center. But unfortunately traditional chargeback and APM tools are not collecting the aresource utilization per transaction to compute the cost on business-aligned and chargeback paradigms. A solution is very much needed in cloud, which can monitor the consumption for each and every service across multiple applications and tiers. This will also help us to accurately find the cost of services and hence to decide about an appropriate chargeback scheme, tune the applications and infrastructure for higher resource usage at a less cost.

4. NOT ALIGNED WITH THE BUSINESS – HOW TO ENSURE THAT SERVICES ARE ALLOCATED ACCORDING TO BUSINESS PRIORITY?

Clouds provide us new degrees of dynamic resource allocation. But we need to ensure that SLAs within the cloud are met. We should also be able to determine the assignment of resources based on measurements of real end-user performance and a precise read of whether extra resources will really eradicate SLA risks. In order to make it doable, we have to obtain a clear cut picture of resource utilization at the transaction level and business intelligence about the effect of each infrastructure tier on performance. Providing supported business preferences becomes additionally important as cloud architectures transition to a dynamic auto-provisioning model.

5. OVER-PROVISIONING – HOWEVER ARE YOU ABLE TO RIGHT-SIZE CAPABILITY AND FORESTALL OVER-PROVISIONING THAT UNDERCUTS ROI?

IT infrastructure sharing is an additional economical associate in nursing efficient – forward you've got a correct image of usage of resources for every service, and understanding of how that allocation affects SLA compliance and the capability to designate the allocation of resources. In cloud for making intelligent decisions about provisioning, a complete history of all transaction instances including precise resource utilization metrics and SLAs are essential. And with a associate in nursing correct image of resource consumption for every business dealings, cloud owners can accurately plan future capacity requirements.

VI. PERFORMANCE TESTING IN CLOUD: A PRAGMATIC APPROACH

Before concluding perceptibility of cloud, one have to examine the computing variables obtainable in market, why then Cloud has niche market hold. Realistically, it's not helpful in making any judgment in binary terms as commerce will normally want to collect and select the flavors that can make sensation for them. Additionally, there are questions of quality while evaluating the features makes sense for profession. In professional and scientific phrase below are few notes suggesting the merits, the restrictions and the problems faced by the Cloud computing. Coincidentally the merits or the potential are also referred as flaw in some places but they are merely related to the provision created by the service providers and the facilities to support.

Adequate availability of Services makes organizations suspicious of the Cloud computing. Ironically publically existing cloud computing offerings are stable with better standards. According to "Berkeley View of Cloud Computing" users are considering Google Search as the basic version of net connectivity: while searching in Google and if it's not available to render the required service, then proclaim that network is not available. End user expects more or less same convenience from new providers that are tough to try to do.

APPROACH

Cloud based load generators
Cloud based test environments
Cloud based test environment management
Parameters to consider while doing the testing (tool type)
Platform support
Ease of installation
Ease of scripting
Ease of use
Data management
Stability
Real time monitoring
Real time load monitoring
Dynamic load monitoring
IP Spoofing
Bandwidth throttling

Cloud solutions inclined to possess rendering profiles that area unit serviceable in existence, reckoning on the request feeding to cloud in any explicit instance. When I record the performance on cloud-based processes -- some that area unit Input-Output connected, some that aren't -- i buy results that change indiscriminately throughout the day. In fact, they seem to possess the pattern of a really uptight method. Explicitly, the job or system is troubled to get virtual machines, which find difficulty in obtaining physical machines. This "jitter" isn't in the slightest degree unplanned, depends on the other users or processors, who are accessing the same hardware in same moment.

Of course, in a very cloud computing surroundings, you can spin up as many instances of computing resources as you need. As you utilize a lot of instances, any variation in the performance of a single instance is masked by the sheer number of instances. Moreover, as you spin up instances, they usually reside in several physical machines,

that conjointly lessens resource rivalry and on the average keeps payloads' performance on par with one another.

Still, the particular performance of your cloud system across several instances depends for the most part on however well it has been designed. Providers differ significantly in their cloud architecture and design prowess.

The variability performance solely becomes a problem once individuals ought to suffer through associate degree I/O-intensive and/or chatty application wherever inputs and screen writes area unit perceptibly scattered. Alternatively, it may happen when the performance varies more on the slow side, and large processes -- such as huge database transformations and writes that occur in daily runs -- don't take place at optimal times. But that is once individuals (customer or IT admins) appear to most care concerning performance.

Any affordable cloud supplier offers service-level agreements (SLAs) that outline performance commitments like interval. Most of the time, cloud services do not fall below those SLA commitments after you average their performance across on a daily basis or every week. They're just, well, variable.

If that variability is an issue, you can ask for a dedicated physical instance -- what used to be called "hosting" -- which many cloud providers offer as part of virtual private clouds? However, your prices go far after you get dedicated resources. Or you will simply keep the process in-house if variability in performance is really a hindrance to productivity. Again, this can be usually not as efficient as employing a cloud service and ready to accept the inherent shifts.

Perhaps what we tend to all really want to try to to is learn to share. I'm unsure the cloud jitters area unit aiming to escape anytime shortly.

Benefits	Limitations	Challenges
Speed to Value	Availability of a Service	Data Governance
Cost Reduction	Data Lock-In	Manageability
CapEx Avoidance	Data Confidentiality and Auditability	Reliability & Availability
Greener IT	Data Transfer Bottlenecks	Virtualization Security
Highly Automated	Performance Unpredictability	Monitoring
More Mobility	Scalable Storage	Latency
Flexibility	Bugs in Large-Scale Distributed Systems	Connectivity
Scalability	Scaling Quickly	Regulatory compliance
Allows IT to Shift	Reputation Fate Sharing	Privacy, Legal
Focus	Software Licensing	Open source, Open standards
Easy Implementation		Sustainability and siting

VII. HOW TO CHOOSE A CLOUD TESTING SOLUTION

Most of the cloud load testing solutions enables user to make use of the functionality of cloud in some way, but few enable the user to follow the best practices mentioned here and at large on the offer that load testing intended. Even though we can drive at a maximum speed in highway than service road, the type of vehicle driving makes a big difference in how speedy and safely the vehicle reaches the designated point. Similarly, load testing with the cloud offers have more advantage than traditional load testing, but the testing suits used are more pertinent to the excellence of the conducted tests.

While looking for cloud validation solution, tester must keep in mind the following points:

- The extent of integration of this solution with the cloud
- Does this help us to experiment sensible tests?
- Does this support the unified tests, irrespective of firewall settings?
- Ease of usage of the solution, or requires expert learning and has to be configured.
- Does this include full pledged investigation and decision-making modules to help the members to make use of these results?
- Can the solution able to extend supports to the technologies used for building the application?

VIII. CLOUD PLATFORM INTEGRATION

If tester choosing a non-integrated method with multiple cloud platforms, he may end up in handling many time-permitting tasks on own. Primarily, have to find out how every platform are victimization works, which includes the restriction and its limits. Secondly, tester has to create, validate, and sustain own virtual machine images.

Some solutions that can be integrated with the cloud alter and speed up the pace required to use the cloud infrastructure in load testing. This solution can provide either one or more of the following benefits over other non-integrated possibilities:

Preconfigured images for fast provisioning: The infrastructure can be set up in minutes.

Security in simplified manner: Requisite protections including firewall, certificates, and encryption are setup by default.

Better scalability: Major load testing solution vendors have worked out with cloud service vendors to permit users of their own tools to create additional virtual machines (for the sake of load testing) than the default offerings.

Multiple service providers with a unified interface: tester can have the liberty of taking extreme benefit of the cloud in minimum time, if provisioning and charge details can be hiding by Load testing solutions.

Test launching in advanced stage: Saving time and energy by shaping and propel load creators within the cloud is directly possible from the load testing access mechanism.

Observation & results Generation: Clear cut results from every geographical area units concerned within the take a look at are obtainable for investigation.

But in general, one limited solutions embody each one of those integration strengths. Many of the solutions comes in some place in the band between very limited or without integration and full-pledged integration with multiple cloud service providers.

REALISTIC TESTS

In spite of testing from the cloud in several instances, is a lot more practical than testing within workplace, simply operating to the cloud isn't sufficient to confirm the foremost practical tests. End users usually might have a less information measure than a testing load generator in an exceedingly cloud information point. A much lower affiliation, the $64000 user may need to be patient than the load generator to transfer the entire information required of an online webpage or stubs. The two vital impressions for this are:

The measured response times with cloud is nearly boundless information measure area unit is higher than the end users. It may mislead the tester to conclude into wrong way, by interpreting that users may get a satisfactory response time, where actually opposite in reality.

The load generator will increase the count of established interconnections with the server, because the average, interconnection time for end users are lengthier than that of load generator. This may lead to a point where the server refuses additional connections under load in a unexpected manner.

While selecting a tool for load testing, it's always better to look for solutions that can offer a feature with a bandwidth simulation, which can restrict bandwidth to make sure that users are downloading the data from internet/web application in a sensible way. This potential is especially necessary while testing applications via mobile, because this type of device mostly operates at a lower bandwidth than PC and laptops.

Likewise, have to explore tools/software to parallelize requests. Currently almost all web browsers have the capability to parallelize HTTP requests, since they can extract web page's static resources. This concurrent request needs multiple connectivity with the server and can elaborate the times to respond. Load testing solutions which cannot handle multiple concurrent requests are not feasible to produce the faithful and sensible tests for web applications.

UNIFIED LAB TESTING AND CLOUD TESTING

Companies which are using either lab testing or cloud testing are at a disadvantage. So companies have to adopt variety of tools for such activities.

The solution that can enable the lab testing will help the engineers to verify the performance of any application at in-house, before its hosting in Internet. This will help to diagnose and rectify the performance related issues at an earlier stage in development lifecycle. This will also help to lower the cloud costs by helping the developers to perform in-house performance tests on readily available hardware platform.

Importantly, single solution which supporting test labs and cloud testers, to reuse the same codes for both kinds of tests, can save time and replication of work. Reusing the codes may also helpful in pinpoint performance issues that may visible in cloud testing and may not be visible during in house testing. Lastly, this type of single hand solution can be helpful in reduce the license fee and familiarization costs, and can edged the testers to sharpen their skill set for conducting the load test on both ways.

USAGE ADVANTAGES

Performances testing, which are happening by the end of the development lifecycle, are mostly executed under tight time schedules. Any delay in initial phases including requirement gathering or development stages of a project can be crucially affected the time availability for tester to perform their jobs. This will increase the stress to hand over the results as soon as early. In such situation a tool with difficult to configure and use are ruled out.

During the development and execution phases of performance tests (it can be at in-house or over the cloud) many important attributes has to consider improving testers efficiency, including the support for: Effortless initiating the recording of virtual user profile (mostly in single click).

Detailing advanced behaviors with the help of a user interface, complemented by the ability to use a scripting language for more complex cases.

Dynamic parameters handled automatically. It consists of grouping of related rules for the known frameworks for servers. Typically, this solution will progressively identify and xecute user specified attributes to the application.

Sharing of common script among profiles of multiple users (login or logout transactions).

Comparison of results: Driving through the end results to determine whether the change is any specific application or infrastructure made drastic changes like more time-consuming and cumbersome without a dedicated comparison tool.

This are some of the usable features that can leverage testers job in an efficiently way. This is not a full list, rather can be consider as a baseline of least required capabilities for an efficient load testing solution.

IX. ANALYSIS, MONITORING, SCHEDULING & REPORTING

Collecting the virtual user profile and rebuilding it to collect the outcomes is just the starting of a successful performance test. Special tools have to be used for analyzing this data, (if possible real time data collection), reason for the issues, and make the action to be taken.

On the run analysis will help to identify and apprehend the problems in real time basis. With this, the tester need not to have waited to complete the test in order to identify the problems, to correct it, and can start again the testing. If this test is happening in assembly line, on time analyzing may be helpful in terminate the testing, without affecting the glitch to end users.

To find the basic origin of an issue, an extensive observing system is required. With the help of a fine tuned performance monitors and alert based system having programmable threshold levels can make it easy to define and analyze the counters. For a non interrupting solution which is easy to programmed, always choose tools that can mange remotely without human interaction.

If your company used to perform regression tests on regular basis - and even if it doesn't – tester can perform these tests automatically via keying the scripts through the command line to finish functional testing. Frequently scheduling load tests with self created reports may be helpful to companies to identify the performance degradation as early as it occurs, to analyze and to precise.

Last, report generation is vital and crucial for informing the findings to other team members and higher authorities.

If any changes has to be made to reports, it's advisable to provide choices to open the tools which can assist other formats, like Word, PDF, XML, and HTML for easy incorporation with other systems.

SUPPORT FOR WEB TECHNOLOGIES

For testing any Siebel applications or any application developed using Oracle Forms, Real-Time Messaging Protocol (RTMP), Adobe Flex, AJAX push technologies or Microsoft Silverlight, we have to use load testing tool having built-in support for these technologies. In absence of this specialized support, it might be quiet tough or may not be possible to carry out the performance testing of such applications.

The load testing solution chosen has to cater necessary assistance for authentication methods deployed by the applications, irrespective of Basic, Digest, NTLM, or Kerberos. Other hand, it's difficult to set up a virtual user profile which can tests the application like a tester would use it.

X. TESTING METHODS & TOOLS

Many standard non-cloud testing tools can be re-used for performance, stress and benchmarking of Cloud Computing Environment. The commonly using performance testing tools like Win runner and Load runner from HP is a good candidate for performance evaluation under normal circumstance with limited number of virtual users.

No	Name of tools and uses		
	Name of tools	Type of test	
1	SOASTA Cloud Test	performance	Load
2	Xamarian test Cloud	performance	CPU
3	App Thwack	performance	Simulation
4	BlazeMeter	performance	Load
5	Jenkins Dev@Cloud	performance	Elasticity
6	Nessus	performance	Compliance
7	LoadStorm	performance	Scalability

Final thoughts on performance metrics: There is no "single right answer" for a performance dashboard; which works for a project and team might not work for another project. My first question would be: Who is the customer? Engineers can need metrics to assist tweak performance, managers will want to spot problems early and executives will just want to know that everything is OK. (Or, if not, who is working on it.)

The core issues with any dashboard are the audience, the problem to be solved, the effort to get the data and the data's value. Historically, I've found that the data that is most valuable to the customer is the hardest to gather. I have to be terribly careful with machine-controlled measures to form positive the info} is having access to helpful information.

There is a balance here between providing an excessive amount of data and not enough. On one hand, too much information can make figuring out the status a bit like looking for a needle in a hay stack, while too little runs the risk that you don't include something important and relevant.

XI. CONCLUSION

The paper discussed the testing methodology used for testing various aspects of IaaS, PaaS and SaaS. It describes the popular kinds of bugs that appear in the Cloud Ecosystem and the recommends some guidelines to reduce the number of bugs in the system. Popular validation software used for cloud validation is described and their short comings for testing a cloud computing system are identified. The effort required to build new tools for testing the cloud .vs. the practical implication on testing an operational cloud is yet quite debatable.

ACKNOWLEDGMENT

We are extremely thankful to Executive Director – CDAC Bangalore, Group Heads and Ms. Mangala, Senior Director for their valuable guidance, support and encouragement.

REFERENCES

1. Yatendra Singh Pundhir , "Cloud Computing Applications And Their Testing Methodology", Bookman International Journal of Software Engineering, Vol. 2 No. 1, Jan-Feb-Mar 2013
2. http://www.disa-apps.com/Services/DoD-Cloud-Broker/~/media/Files/DISA/Services/Cloud-Broker/nist-cloud-ref-architecture.pdf
3. http://clds.sdsc.edu/sites/clds.sdsc.edu/files/wbdb2012/papers/WBDB2012Paper40Jost.pdf]:
4. Report on Cloud Computing to the OSG Steering Committee, SPEC OSG Cloud Computing Working Group, to appear 2012.
5. Michael Lynch, Thomas Cerqueus, Christina Thorpe, "Testing a cloud application: IBM SmartCloud inotes: methodologies and tools", International Workshop on Testing the Cloud, IISTA, July 2013, Switzerland.
6. William Jenkins, Sergiy Vilkomir, Puneet Sharma, "Framework for Testing Cloud Platforms and Infrastructures ", 2011 International Conference on Cloud and Service Computing
7. Dr. Rahul Malhotra, Prince Jain, "Testing Techniques and its Challenges in a Cloud Computing Environment", The SIJ Transactions on Computer Science Engineering & its Applications (CSEA), Vol. 1, No. 3, July-August 2013
8. Eric Bauer, Randee Adams, "Reliable Cloud Computing – Key Considerations", http://www2.alcatel-lucent.com/ techzine/reliable-cloud-computing-key-considerations/
9. Sue Poremba, "Open Source Cloud Computing Testing Tools", Cloud Expo, 25 April 2013, http://www.Cloud computingexpo.com/node/2625872
10. Hanyu Pei ; Beibei Yin ; Min Xie, "Dynamic Random Testing Strategy for Test Case Optimization in Cloud Environment" 2018 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW). Pages: 148 - 149
11. Sheng-Jen Hsieh; Shyan-Ming Yuan; Guo-Heng Luo; Hsiao-Wei Chen "A flexible public cloud based testing service for heterogeneous testing targets" The 16th Asia-Pacific Network Operations and Management Symposium 2014, Pages: 1 - 3
12. Pillutla Harikrishna ; A. Amuthan, "A survey of testing as a service in cloud computing", 2016 International Conference on Computer Communication and Informatics (ICCCI) 2016, Pages: 1 - 5
13. Wei Wang ; Ningjing Tian; Sunzhou Huang; Sen He; Abhijeet Srivastava; Mary Lou Soffa; Lori Pollock, "Testing Cloud Applications under Cloud-Uncertainty Performance Effects", IEEE 11th International Conference on Software Testing, Verification and Validation (ICST) 2018, Pages: 81 - 92
14. HongHui Li; XiuRu Li; Huan Wang; Jie Zhang; ZhouXian Jiang, "Research on Cloud Performance Testing Model", IEEE 19th International Symposium on High Assurance Systems Engineering (HASE) 2019, Pages: 179 - 183
15. Nitin Naik, "Connecting google cloud system with organizational systems for effortless data analysis by anyone, anytime, anywhere", IEEE International Symposium Systems Engineering (ISSE) 2016, Pages: 1 – 6

AUTHORS

ABEY JACOB had completed B.E (ECE) and M.Tech (CSE) and at present working as a Principal technical Officer with Centre for Development of Advanced computing (CDAC), Bangalore - India. He is currently pursuing PhD in cloud computing, at CSE Department of Dr.M.G.R University, Chennai - India. His areas of interest are Grid Computing, Cyber Forensic and Cryptography. He has published couple of papers in National and International Journals.

Dr. V CYRIL RAJ had completed M.E, PhD and presently working as Join Registrar (E&T) at Dr.M.G.R University, Chennai - India. Formerly he is the Dean (E&T) & HoD for Computer science department. He has guided more than 20 PhD students and his areas of interest are Bioinformatics Data Mining, Grid Computing, Cloud & Mobile Computing, Robotics etc. He had published a lot of papers in International and National level journals and also authored many books.