A Study on Cloud Platform Testing Analysis and Validation Using Fuzzy with Public Cloud Characteristics

*A.Vanitha Katharine
Associate Professor,
PSNA College of Engineering and Technology,
Dindigul, Tamil Nadu, India.

Abstract: *Software product development life cycle has software testing as an integral part Software Testing is a challenging activity for many software engineering projects and it is one of the five main technical activity areas of the software engineering lifecycle that still poses substantial challenges Depending on the testing method employed, software testing can be implemented at any time in the development process. Traditionally most of the test effort occurs after the requirements have been defined and the coding process has been completed, but in the agile approaches most of the test effort is on-going. Cloud computing has opened up new opportunities for software testing, which provides unlimited resources with scalability, flexibility and availability of distributed testing environment. It reduces the execution time of testing of large applications and lead to cost-effective solutions It frees companies from large capital expenditures for creating test environments, and pay-as-you-go models mean need not to pay for idle test environments. Besides certain challenges cloud can attain its efficiency by taking care of parameters like network traffic, disk storage and RAM.*

Keywords: *Software testing, cloud testing, Fuzzy, TaaS, Saas, IaaS, PaaS,*

1. Introduction

Testing requires expensive dedicated infrastructure and resources that were only used sporadically which scrutinizes the application's performance, reliability, speed, security and functionality. Since, business applications are growing in complexity, it is somewhat difficult for organizations to build and maintain in-house testing facilities that imitate real-time environments. This is where cloud testing has emerged as a fresh approach to testing where cloud computing environments are leveraged to simulate real-world user traffic by significantly decreasing costs [1]. A combination of pay peruse, lower cost and upfront capital expenditure elimination[2] is offered by cloud testing than that of conventional test environment. At the same time, new set of challenges are introduced by cloud testing especially in public cloud which includes lack of standards and data security [8].

A traditional test environment restricts Time-to-market[3] since it can be delay prone and time consuming. Single application can take weeks or months which leads to delay release. Forward thinking companies due to on-demand provisioning starts the process[18], Time-to-market is often restricted by traditional test environments. That’s because creating on-premise test environments can be time-consuming and delay-prone. It can typically take weeks or months to set up a simple test environment for a single application, delaying its release. Ondemand provisioning jumpstarts the process for forward-thinking companies, since testing resources required to meet time-tomarket demands already exist in the cloud and can be provisioned instantaneously Despite advances in formal methods and verification techniques, a system still needs to be tested before it is used.

Testing remains the truly effective means to assure the quality of a software system of nontrivial complexity [13], as well as one of the most intricate and least understood areas in software engineering [19]. With cloud-based testing, organizations no longer need to worry about finding servers, procuring licenses for programs and testing tools and installing them. Service providers give testers access to scalable and ready-to-use virtual labs with a library of operating systems,

test management and execution tools, middleware and storage necessary for creating a test environment that closely mirrors the real environment.

Testers can run existing applications and virtual machines with minimal or no rewriting and utilize pools of virtualized infrastructure to scale up the test environment within minutes [4]. There are some challenges associated with cloud testing such as security, lack of standards, infrastructure, usage, planning and test data[5][6]. With cloud-based testing, organizations no longer need to worry about finding servers, procuring licenses for programs and testing tools and installing them. Service providers give testers access to scalable and ready-to-use virtual labs with a library of operating systems, test management and execution tools, middleware and storage necessary for creating a test environment that closely mirrors the real environment. Testers can run existing applications and virtual machines with minimal or no rewriting and utilize pools of virtualized infrastructure to scale up the test environment within minutes.

Thus the goal is to use a mathematical fuzzy model to increase the performance of testing in cloud by handling the user defined parameters network traffic, disk storage and RAM in the cloud server. As a part of infrastructure resource, cloud testing can attain its efficiency by taking care of the parameters like network traffic, Disk Storage and RAM speed[7]. Added to these parameters, limitations of different kinds of clouds also can be taken into consideration. Mainly public cloud and private cloud is taken into consideration.

Thus a mathematical fuzzy model is used to test the limitations of public and private clouds which results testing in public clouds are more efficient. Takagi-sugeno fuzzy model is more realistic and reliable model for dynamic system than Malthus model, verhurst model and harvesting model[25].

In this work sugeno model is tested in statistical system. The below figure 1 indicates that testing and application development rank second (57%) as the most likely workload to be put into the cloud after Web sites (61%)[2]. Hence the cloud platform provides cost effective solution to the software testing, still it poses the some challenges.

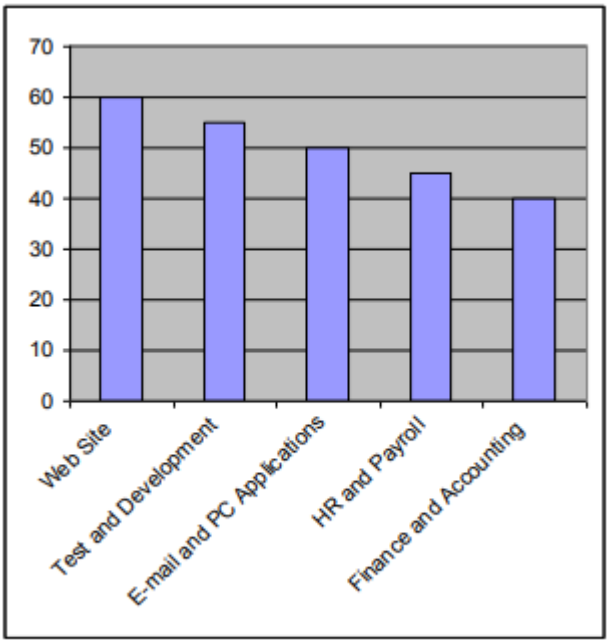

Figure1: TOP Applications in Cloud

2. Literature

This section briefly reviews the published research work relating to software testing using Cloud Platform. Vengattaraman et al. in [8] propose their initial work on modeling of cloud based application environment for software testing by focusing On-Premises Applications over clouds. Its major objective is to present the relationships between different application services over clouds and external consumer services. But it does not address how to use this model in cloud testing. Yang Yang at al. [9] discussed that software testing can be conceptualized as a service rather than being viewed as a sequential line of responsibility in software development.

In their view, TaaS has two key aspects: (1) a service to developers, and (2) a service to end users. Their paper discusses software testing as a service from software quality assurance perspectives. Yu, et al., [12] defined a 5-layer TaaS

framework based on cloud infrastructure services, including:1) test tenant and test service contributor layer, 2) test task management layer, 3) test resource management layer, 4) test layer (and it has three components: testing service composition, testing service pool, and testing task reduce), and 5) test database layer.

This paper proposed an automated testing platform TaaS on a cloud. This platform adopts cloud computing technique to build the elastic resource infrastructures, and provide various kinds of testing services to testing users. Candea [10] identified three categories of testing services: TaaSD for developers, TaaSH for end users and TaaSC as certification service. They argue that with a pricing model, TaaS can be operated as a public service and as a business, targeting at the ”long-tail” small business companies.

Liviu Ciortea et al. in [11] introduce Cloud9, a cloud based testing service that promises to make high quality testing fast, cheap, and practical. Cloud9 is the first parallel symbolic execution engine to run on large shared- clusters of computers, and its test harness uses the aggregate memory and CPU resources based on compute utilities like Amazon EC2. The paper reports their initial prototype results. In addition, some initial cloud-based test experiments are reported. Banzai et al [12] from University of Tsukuba in Japan developed “D-Cloud”, is a dedicated simulated test environment built upon Eucalyptus, an open-source cloud infrastructure providing similar functionalities as Amazon EC2.

It uses QEMU, an open-source virtual machine software, to build virtual machine for simulating faults in hardware including disk, network and memory. Parveen et al [53] migrates JUnit test framework to Hadoop platform. JUnit test cases are created as independent Hadoop MapReduce jobs. The map() function receives test jobs as testname; testcommand> pair.

At each node, the command is executed as a process. The reducer gets < testname; testresult> from each map and combines all the results. Experiments shows that a 150-node cluster can produce 30x improvement compared with sequential test executions on a local computer. Moreno et al [14] proposed a new distributed testing architecture for simulating parallel jobs.

This framework contains two types of nodes - Master and Slave. Master is unique identified, which is responsible for distribution, synchronization and management

of all slave nodes. Master is started with a given test. It waits for enough slaves to connect to it, then sends every slave corresponding tasks.

During the execution, the master controls the execution sequence of slaves to guarantee that all tasks in a step start at the same time. The slaves run testing tasks and store test results locally, including nodes' states collected by daemon thread at each node. At the end of the test, master recollects every slave's results, analyzes and then generates statistics and graphs for analyzing test execution. Baride, et al [15] proposed a cloud-based approach for mobile application testing where infrastructure services are used to simulate diversified mobile devices, hardware configurations, heterogeneous application platforms, and complex dependencies.

Cloud computing is a comprehensive solution that delivers IT as a service. It is an internet-based computing solution where shared resources are provided like electricity distributed on the electrical grid [9].

Computers in the cloud are configured to work together and the various applications use the collective computing power as if they are running on a single system. With the advent of cloud computing, resources are used as an aggregated virtual computer.

This gives the flexibility to allocate the resources on demand. Before dwelling to cloud testing, we need to understand the cloud model. Cloud model is composed of three broad service model (SaaS, PaaS, IaaS) and four deployment model [10] (Public, Private, Community, Hybrid Cloud) as shown the below figure 2.

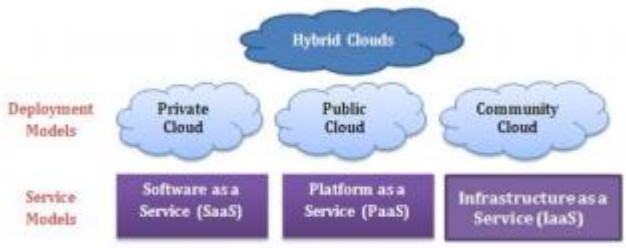

Figure 2: Cloud Computing

3. Proposed Work

Fuzzy logic depends upon the context that cannot be defined precisely and is a form of knowledge representation suitable for notions. It is a Problem-solving control system methodology. We move to fuzzy due to the following reasons. Conceptually fuzzy logic is easy to understand[22], Fuzzy reasoning mathematical concepts are simple. Fuzzy logic does not need any far-reaching approach, it in an intuitive approach. It need not start any thing from scratch and flexible with any system and easily layer on more functionality.

Even on careful inspection most things are imprecise when looked closely but fuzzy logic tolerates on imprecise data[23]. Rather than tacking it onto the end fuzzy reasoning builds this understanding into the process. Fuzzy system can be created to match any set of input-output data, nonlinear functions of arbitrary complexity can be modeled in fuzzy logic.

Fuzzy logic toolbox software[24] has a technique like Adaptive Neuro-Fuzzy Inference Systems(ANFIS), which is an adaptive technique which ease the above process. Neural networks is in direct contrast with fuzzy logic which generates opaque data taking training data as input with impenetrable models, fuzzy logic rely on the experience of people who already understand the system thus built on top of the experienced experts.

A fuzzy system does not replace conventional control methods in turn blend with the conventional control techniques. It simplifies their implementation and augments them in many cases. Fuzzy logic is the basis for human communication thus based on natural language[23].

Many other statements about fuzzy logic are underpinned by this observation, because it is easy to use and built on the structures of qualitative description used in everyday language. A fuzzy logic system (FLS) is defined as the nonlinear mapping of an input data set to a scalar output data[4].

Fuzzy logic depends upon the context that cannot be defined precisely and is a form of knowledge representation suitable for notions. It is a Problem-solving control system methodology. We move to fuzzy due to the following reasons. Conceptually fuzzy logic is easy to understand[22], Fuzzy reasoning mathematical concepts are simple. Fuzzy logic does not need any far-reaching approach, it in an

intuitive approach. It need not start any thing from scratch and flexible with any system and easily layer on more functionality.

Even on careful inspection most things are imprecise when looked closely but fuzzy logic tolerates on imprecise data[23]. Rather than tacking it onto the end fuzzy reasoning builds this understanding into the process. Fuzzy system can be created to match any set of input-output data, nonlinear functions of arbitrary complexity can be modeled in fuzzy logic.

Fuzzy logic toolbox software[24] has a technique like Adaptive Neuro-Fuzzy Inference Systems(ANFIS), which is an adaptive technique which ease the above process. Neural networks is in direct contrast with fuzzy logic which generates opaque data taking training data as input with impenetrable models, fuzzy logic rely on the experience of people who already understand the system thus built on top of the experienced experts.

A fuzzy system does not replace conventional control methods in turn blend with the conventional control techniques. It simplifies their implementation and augments them in many cases. Fuzzy logic is the basis for human communication thus based on natural language[23].

Many other statements about fuzzy logic are underpinned by this observation, because it is easy to use and built on the structures of qualitative description used in everyday language. A fuzzy logic system (FLS) is defined as the nonlinear mapping of an input data set to a scalar output data[4]. A FLS as in consists of four main parts: fuzzifer, rules, inference engine, and defuzzfier.

A membership function is used to quantify a linguistic term[24], which are used in the fuzzification and defuzzification steps of a FLS, to map the non-fuzzy input values to fuzzy linguistic terms and vice versa. With a condition and a conclusion a fuzzy rule is a simple IFTHEN rule. if the When network traffic in high the performance is low and when the network traffic is low the performance is high. Membership function is constructed to control the output variable based on rule.

To look into a better performance one more input RAM is considered along with network traffic and disk storage. When RAM speed is high then performance of system is also high. Combining this with network traffic, disk storage based on AND rule new rules can be framed.

show the performance range of disk storage. These added benefits are not found in private clouds, which attempt to mimic the public-cloud idea behind the firewall.

"While the public cloud addresses all sources of variability, the private cloud can address only a subset," For example, industry variability cannot be addressed by a private cloud, while growth variability can be addressed only to a limited degree if an organization pools all its internal resources in a private cloud [14].

Thus a mathematical fuzzy model is developed using certain attributes such as performance, reliability & security, administrative control, market, capital benefits, TCO benefits, System availability and scalability[17] with which public cloud is preferred than that of private cloud. Computer performance is characterized by the amount of useful work accomplished by a computer system compared to the time and resources used.

Depending on the context, good computer performance may involve one or more of the following: Short response time for a given piece of work High throughput (rate of processing work), Low utilization of computing resource(s), High availability of the computing system or application Fast (or highly compact), data compression and decompression, High bandwidth / short data transmission time. Reliability and security means trustworthy computing with data security.

Apart from security, these managed service providers give you disaster recovery and data backup plans so that intense situations like cloud collapse or data crash could be handled or managed with ease and less efforts. Apart from security, these managed service providers give disaster recovery and data backup plans so that intense situations like cloud collapse or data crash could be handled or managed with ease and less efforts [23].

4. Results and Discussion

This section highlights the some of the key research issues [21]. Define the characteristics of an application under test and the types of testing done on the application. Parveen and Tilley [20] highlights the characteristics of an application under cloud testing are test case dependency, the operating environment within which to carry out testing and the ability of an application’s interface to be programmed.

The types of testing identified to be appropriate for testing in the cloud are unit testing, high volume automated testing and performance testing. Evaluate whether

certain testing infrastructure in the cloud really helps to meet a specific performance attribute. SOASTA, a cloud testing provider, has for almost two years been providing performance testing of web applications in the cloud [16].

However, many other applications as well as the cloud itself need to be tested for various performance attributes e.g. response time, speed and throughput. So a suitable approach to address this issue is by running a test bed which would encourage researchers to explore different aspects of performance testing, reflect on the experiences encountered in the process and thereafter give recommendations based on the results.

Validate the quality of cloud tested applications at all levels. Quality is sometimes a highly subjective attribute, varying due to different end-user expectations. So the researchers have to develop a testing methodology to verify and validate the quality of overall testing in the cloud. Monitoring and managing the software testing processes. With the growing number of cloud providers, customers have a wide range of choice from where to get cloud services.

ssuming a customer chooses to acquire the services from more than one cloud testing provider, it would be efficient to have some methods, tools and facilities for monitoring and managing the software testing processes from the different providers in an “all-in-one” fashion.

Management of test data The management of test data is a delicate issue. In order for effective testing to take place, some testing tasks depend highly on the actual customer or production data. In some cases, due to rules and regulations, customers are prohibited from supplying confidential or production data to third parties. A solution to this problem may be the development of new models or algorithms that would generate almost “identical” test data to facilitate productive testing results

There are some challenges associated with cloud testing [7][13]. Security: Security in the public cloud is still a major concern. Cloud test is based on the internet. So it may come up to the situation about the leakage of private information, internet suspending, and service provider may suddenly announce disruption of service due to a maintenance window, slow internet speed, virus attack and so on. Procedures are being developed to improve security and performance in the public cloud.

Lack of standards: Presently, there are no universal/standard solutions to integrate public cloud resources with user companies' internal data center resources. Public cloud providers have their own architecture, operating models and pricing mechanisms and offer very little interoperability. This poses a big challenge for companies when they need to switch vendors. Infrastructure: Some cloud providers offer only limited types of configurations, technology, servers and storage, networking and bandwidth, making it difficult to create real-time test environments. Usage: Improper usage of cloud-based test environments can increase costs. Planning: Testing teams should rigorously plan their test environments, from utilization periods through disassembly.

They should also be aware of the associated expenses, such as cost of encrypting data, before putting testing in a cloud environment, since these requirements will consume additional CPU and memory. It's important to monitor utilization of cloud resources to avoid over-usage and over-payment. Test Data: The management of test data is a delicate issue. In order for effective testing to take place, some testing tasks depend highly on the actual customer or production data.

In some cases, due to rules and regulations, customers are prohibited from supplying confidential or production data to third parties. So the test data should be doubly scrutinized for testing in cloud. A fuzzy logic is a precise logic of imprecision and approximate reasoning. More specifically, fuzzy logic may be viewed as an attempt at formalization/mechanization of two remarkable human capabilities[26].

Fuzzy logic is much more than a logical system. It has many facets. The principal facets are: logical, fuzzy-settheoretic, epistemic and relational. Most of the practical applications of fuzzy logic are associated with its relational facet[20]. Fuzzy inference system also known as fuzzy rule-based systems of fuzzy models are schematically[21].

Two major types of fuzzy rules exist, namely, Mamdani fuzzy rules and Takagi-Sugeno fuzzy rules. The fuzzy model proposed by Takagi and Sugeno is described by fuzzy IF-THEN rules which represents local input-output relations of a nonlinear system. The main feature of a TakagiSugeno fuzzy model is to express the local dynamics of each fuzzy implication (rule) by a linear system model.

The overall fuzzy model of the system is achieved by fuzzy "blending" of the linear system models. The input for the model is taken as the parameters of local,

private and cloud To enhance testability of autonomic services, TSaaS was proposed so that each service will expose both production and test environment to external users. Test functions (such as specification, execution, configuration and reporting) are exposed as API services. King, et al.,[5], [6] applied autonomic computing concepts to testing of adaptive systems, called autonomic self-testing (ATC). The technique was then migrated to the cloud platform [7], called TSaaS, so that services that are hosted on remote cloud platform can expose their test support APIs for partner providers.A self-test harness is developed to manage testing workflow and activities. It monitors changes or updates on hosted services, utilizes necessary infrastructure services, and invokes TSaaS supporting services to validate the changes.

Test operations exposed as supporting services include test setup, input, assertions, and teardown operations. These services are provided for cloud partners during the development, testing and maintenance of tailor-made cloud applications and services. They can also be used for design, build, and deployment of automated tests across administrative domains. TaaS concept was initially introduced by Tieto in Denmark in 2009. Software Testing as a Service (TaaS) is a model of software testing used to test an application as a service provided to customers across the Internet.

By eliminating the need to test the application on the customer's own computer with testers on site, TaaS alleviates the customer's burden of installing and maintaining test environments, sourcing and (test) support. Using TaaS can also reduce the costs of testing, through less costly, on-demand pricing.

4. Conclusion and future work

Conventional software testing incurs high capital cost such as expenditure on hardware, software and its maintenance to simulate user activity from different geographic locations. In case of applications where rate of increase in number of users is unpredictable or there is variation in deployment environment depending on client requirements, cloud testing is more effective. So cloud testing is becoming a hot research topic in cloud computing and software engineering community. The major contributions of this paper is a comparative study on

traditional testing with cloud testing and its challenges New fuzzy mathematical model which provides unlimited resources with scalability[13], flexibility and availability of distributed testing environment is created for cloud computing which opens new opportunities for software testing. Software testing performance can be increased in cloud by increasing the working of cloud parameters namely network traffic, disk storage and RAM. Compared with conventional testing methods, cloud testing emphasizes more on system testing and online testing. This is due to the novel design and development methods imposed by cloud computing. This is still an emerging research area with many open problems. This paper highlights the recent cloud testing architecture, tools and research issues. This will really serve the foundation for the new researchers and students those who really interested in software testing using cloud.

5. References

[1]http://mcahelpline.com/tutorials/testing/testing.p df //abt testing

[2]http://www.cognizant.com/insights/perspectives/ taking-testing-to-the-cloud//abt cloud tes

[3]http://healthcare-elp.blogspot.in/2011/11/cloudcomputing-benefits-in-health-care.html

[4] A.Vanitha Katherine, K. Alagarsamy, “Conventional software testing Vs cloud testing”, International Journal Of Scientific & Engineering Research, Volume 3, Issue 3, March-2012 1 ISSN 2229-5518

[5] Cognizant reports, “Taking Testing to the Cloud”. March 2011.

[6] Jerry Gao, Xiaoying Bai, and Wei-Tek Tsai, “Cloud Testing- Issues, Challenges, Needs and Practice”, SEIJ, Vol. 1, No. 1, Sept 2011

[7] A.Vanitha Katherine and K.Alagarsamy, “A Fuzzy Mathematical Model For Performance Testing In Cloud Computing Using User Defined Parameters”, International journal of software engineering and applications”, July 2013.

[8]http://www.akamai.com/html/solutions/public_p rivate_cloud.html

[9] http://thecloudtutorial.com/cloudtypes.html

[10] Public or Private Cloud: The choice is yours – Aerohive Whitepaper, Aerhive networks, Inc, 2013

[11] The Agile Architecture Revolution: How Cloud Computing, REST-Based SOA, and Mobile Computing Are Changing Enterprise IT, Jason Bloomberg, Wiley CIO Series, 2013

[12] J. J. Marciniak, “Encyclopedia of software engineering”, Volume 2, New York, NY: Wiley, 1994, pp. 1327-1358

[13] E. F. Miller, “Introduction to Software Testing Technology,” Tutorial: Software Testing & Validation Techniques, Second Edition, IEEE Catalog No. EHO 180-0, pp. 4-16

[14]http://www.channelregister.co.uk/2011/03/10/ microsoft_on_the_cloud/

[15] M.R.C. van Dongen, “Computing the Frequency of Partial Orders”, springer link, Volume 3258, 2004, pp 772-776

[16] Merrifield, Richard E.; Simmons, Howard E. “Topological Methods in Chemistry” New York: John Wiley & Sons. p. 28. ISBN 0-471-83817-9. 1989.

[17] Miguel Reixa, Carlos costa, Manuela Aparicio, “Cloud services evaluation framework”, Proceedings of the workshop on open source and design of communication, Pages 61-69, ACM, 2012.

[18] George J. Klir and Bo Yuan, " Fuzzy sets and fuzzy logic: Theory and applications ", Prentice hall, Inc., 2000.

[19] J. A. Whittaker, "What is Software Testing? And Why Is It So Hard?" IEEE Software, January 2000, pp. 70-79

[20] Lotfi A. Zadeh, Is there a need for fuzzy logic?, Science direct Information Sciences 178 (2008) 2751–2779, Feb 2008

[21] Takagi-Sugeno Fuzzy Modeling for Process Control Kamyar Mehran Industrial Automation, Robotics and Articial Intelligence (EEE8005) School of Electrical, Electronic and Computer Engineering, Newcastle University, 2008 [22]http://www.scis.ulster.ac.uk/~siddique/CI/CIWeek2.pdf

[23]http://www.cloudways.com/blog/what-ispublic-cloud/

[24] Chris joyce, "Venn Diagrams", ARB material, Ministry of education, Newzealand, 2008.

[25] N. E. Mastorakis, "Modeling dynamical systems via the Takagi-Sugeno fuzzy model," Proceedings of the 4th WSEAS International Conference on Fuzzy sets and Fuzzy Systems, Udine, Italy, march 25-27, 2004. [26] George J. Klir, Bo yuan, "Fuzzy setrs and fuzzy logic", Prentice hall, 2000.