

Conventional Software Testing Vs. Cloud Testing

Mrs.A.Vanitha Katherine, Dr. K. Alagarsamy,

Abstract

Software Testing is one of the important activities in software development process. 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. Conventional testing requires expensive dedicated infrastructure and resources that are only used sporadically. The growing complexity of business applications, it is harder to build and maintain in-house testing facilities that mimic real-time environments. 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. However, it also explores challenges such as data security and a lack of standards. In this paper we present a comparative study on conventional software testing with cloud testing. We also highlight the challenges of these testing paradigms.

Index Terms: Cloud Computing, Cloud Testing, Testing a Cloud, WAPT, STLC, TaaS, SaaS,PaaS, IaaS

1 INTRODUCTION

Software testing is one of the significant activities in software development. It determines the correctness, completeness and quality of the software product. In general software testing is done internally using the infrastructure that exists (in-house testing) within the organization.

Over time, the software testing function has become a challenging activity for enterprises due to increasing technological complexities, software sourcing challenges, high costs to simulate and security. The growing complexity of business applications, it is harder to build and maintain in-house testing facilities that mimic real-time environments. Cloud computing is opening up new vistas of opportunity for testing. Compared to maintaining an in-house test environment, cloud-based testing offers a compelling combination of lower costs, pay-per-use and elimination of upfront capital expenditures [1]. The benefits, however, extend beyond cost. The non-cost factors includes utility like, on-demand flexibility, freedom from holding assets, enhanced collaboration, and greater levels of efficiency. Most importantly, it reduces time-to-market for key business applications. At the same time, cloud-based testing introduces a new set of challenges, such as data security and a lack of standards, especially in the public cloud model.

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. On-demand provisioning jumpstarts the process for forward-thinking companies, since testing resources required to meet time-to-market demands already exist in the cloud and can be provisioned instantaneously.

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.

Rest of the paper is organized as follows. Section 2 presents the conventional software testing process. Section 3 describes the cloud testing. Section 4 gives the Comparison.

2 CONVENTIONAL SOFTWARE TESTING

Software testing is not a single activity but a series of planned tasks that need to be executed along with the software development activities to ensure that a product is delivered without any errors. Conventional testing can be done in two ways (i). Manual testing (ii). Automated testing using tools. The below figure 1 shows the general software testing life cycle (STLC).

- Mrs.A.Vanitha Katherine is currently working as an Associate Professor in department of MCA, PSNA College of Engineering and Technolog,Tamil Nadu, India.9865302886 E-mail:avanitha@yahoo.com
- Dr.K.Alagarsamy is currently working as Associate Professor in computer centre of Madurai Kamaraj University, Tamil Nadu, India. Email:alagarsamymku@gmail.com

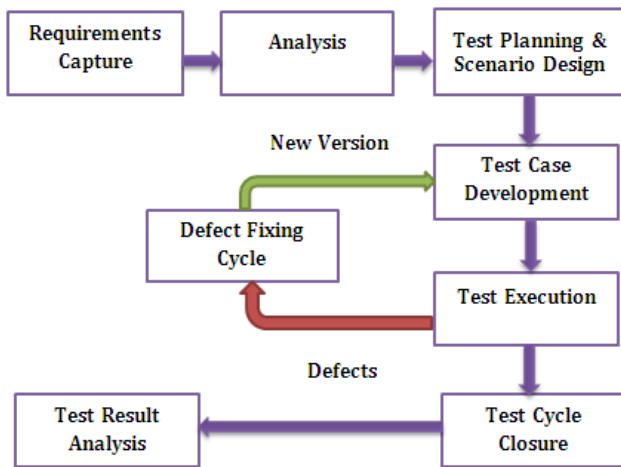


Figure 1: Software Testing Life Cycle (STLC)

Today's internet world the web applications are integral part of human life. The web based applications are powerful and have the ability to provide feature rich content to a wide audience spread across the globe [3]. These web applications are stored in remote server and accessed through the web browser. In order to produce the quality and secured web application, testing becomes the important activity in web application development life cycle. The next section, presents the conventional testing activities for the web application.

2.1 Web Application Testing (WAPT)

Web Application Testing (WAPT) is a process of checking the web applications to find the potential bugs before the code is moved into the live/ production environment. The following testing activities are generally carried out for WAPT [2] [4].

1. Functional Testing
2. Usability testing
3. Interface testing
4. Compatibility testing
5. Performance testing
6. Security testing

2.1.1 Functional Testing

Functional testing ensures that individual functions are working well. Test cases should ensure that boundary conditions are tested. Invalid inputs should prompt appropriate error messages. For web applications, functional testing can range from testing whether links are working to checking whether changes made by users in a web page are reflected in the database.

2.1.2 Usability testing

Usability testing plays a pivotal role with applications that are designed to make manual tasks easier. The applications should comply with accessibility standards. For web applications, usability test includes navigation test and content test.

2.1.3 Interface Testing

Interface testing ensures that individual components are connected properly. The output of one module should be fed to the intended module without any issues. Interface testing plays a vital role for web applications. The main interfaces are web server -application server interface and application server-database server interface. All the interactions between these servers are executed and tested properly.

2.1.4 Compatibility Testing

Compatibility of the web site is very important testing aspect which ensures that the web application displays correctly across different devices. This would include browser compatibility, operating system compatibility, mobile browsing and printing options.

2.1.5 Performance Testing

Performance testing on web applications measures the performance under various scenarios. Performance tests include stress testing, scalability testing and load testing.

2.1.6 Security Testing

Security testing attempts to explore the vulnerabilities of web applications. This would include test unauthorized access to secure pages should not be permitted, restricted files should not be downloadable without appropriate access, check sessions are automatically killed after prolonged user inactivity [5] etc.

2.2 Challenges

1. Testing is a periodic activity and requires new environments to be set up for each project [7].
2. It is harder to build and maintain in-house testing facilities that mimic real-time environments.
3. Huge cost investment for establishing testing environment.

3 CLOUD COMPUTING

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.

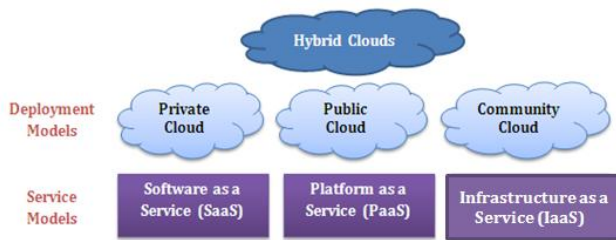


Figure 2: Cloud Computing

3.1 Testing under Cloud

Under cloud computing paradigm, testing can be classified as Testing a Cloud and Cloud Test.

3.2 Testing a cloud

Testing a Cloud [11][8] refers to the verification and validation of applications, environments and infrastructure that are available on demand. This ensures that applications, environments and infrastructure conform to the expectations of the cloud computing business model. Testing a Cloud includes availability testing, security testing, performance testing, interoperability testing, and disaster recovery testing, multi-tenancy testing.

3.3 Cloud Testing

Cloud Testing [11][6] is defined as "Testing as a Service". Testing as a service can be used to validation of various products owned by organizations or individuals. IT organizations that deal with testing products and services are making use of a cloud based licensing model for their end clients. For instance load testing is available as service; instead of installing a load testing product to test proprietary application, we can avail testing service on demand. The service includes functional as well as non-functional testing of various applications/products. The figure 3 shows the general cloud testing life cycle. The table 1 highlights the benefits of testing environment in cloud.

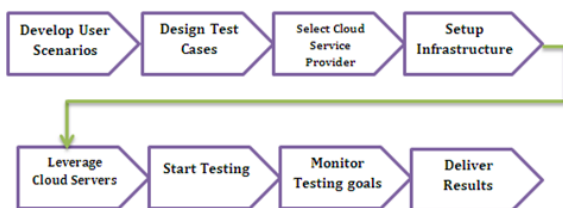


Figure 3: Cloud Testing Life Cycle

3.3.1 Functional Testing

Functional testing of both internet and non-internet applications can be performed using cloud testing. The process of verification against specifications or system requirements is carried out in the cloud instead of on-site software testing.

Support for Multiple browsers

Testing the applications across different browser versions (IE, Firefox, Safari, Google Chrome, Opera) i.e. multi browser comparisons.

Availability

Cloud offerings must be available at all the times. It is the responsibility of the cloud vendor to ensure that there are no abrupt downtimes. In addition the business of the client must not be adversely affected in case of any planned downtime [11].

Accessibility

Accessibility testing is intended to test the accessibility of the application across the geographies. Mostly 24/7 – application access may be required to cater to user groups across different geographies.

Data Security

Testing is to ensure that the potentially sensitive information which traverses the cloud is safe and secure.

Privacy

Testing is ensure that the privacy of the application users and associated information when used in Cloud.

3.3.2 Non Functional Testing

Non-functional testing is done to ensure that a web application meets the specified performance requirements. Unlike the conventional performance testing techniques, where scalability is limited to certain number of users within the network. In cloud, the applications scalability scope is much wider

Performance Testing

Measure response times and isolate issues related to specific steps or actions while system is subjected to increasing load from different locations and multi user operations.

Load Testing

Application stability is a major factor as the user count is expected to be in multiples of hundreds.

Stress Testing

Due to the cloud characteristics, it is imperative to identify issues as system is tested to breaking points maximum expected capacity or often beyond the expected usage.

Capacity Testing

Being hosted in a cloud environment it is prudent to determine maximum capacity for current or future hardware, bandwidth or other needs or to validate that installed hardware and network will support expected usage scenarios.

Other testing are failover testing, application security testing, GRC (Governance Risk Compliance) testing.

4 CLOUD TESTING vs. CONVENTIONAL SOFTWARE TESTING

Here, we compared the similarities and differences between conventional software testing with cloud testing. Table 2 shows the details in different aspects.

5 CHALLENGES

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.

6 CONCLUSIONS

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.

REFERENCES

- [1]. <http://www.cognizant.com/insights/perspectives/taking-testing-to-the-cloud>
- [2]. <http://www.softwaretestinghelp.com/web-application-testing/>
- [3]. <http://www.exforsys.com/tutorials/testing/challenges-in-testing-web-based-applications.html>
- [4]. <http://www.ibm.com/developerworks/web/library/webapptesting/>
- [5]. <http://www.guru99.com/web-application-testing.html>
- [6]. http://en.wikipedia.org/wiki/Cloud_testing
- [7]. Cognizant reports, "Taking Testing to the Cloud". March 2011.
- [8]. A.Vanitha Katherine, K. Alagarsamy, "Software Testing in Cloud Platform: A Survey", IJCA, Volume 46- No.6, May 2012
- [9]. <http://www.ibm.com/developerworks/cloud/library/cloudintro/>
- [10]. <http://www.nist.gov/itl/cloud/>
- [11]. Neha Mehrotra, "Cloud Testing Vs. testing a Cloud", 10th Annual International Software Testing Conference, Aug 2011.
- [12]. H.Kothandaraman, "Testing Applications n Cloud", Computer Science Corporation 2011.
- [13]. Jerry Gao, Xiaoying Bai, and Wei-Tek Tsai, "Cloud Testing- Issues, Challenges, Needs and Practice", SEIJ, Vol. 1, No. 1, Sept 2011

Table 1: Test Environment in Cloud [12]

Attributes of cloud solutions	Characteristics	Benefits
Advanced virtualization	Test resources (infra, tools and people) are pooled and virtualized	Providing efficient implementation of independent infrastructure
Rapid Provisioning	Test resources are provisioned on demand	Reducing test setup and execution time and eliminating errors
Service Catalog ordering	Test environment are readily available	Enabling visibility, control and automation
Elastic scaling	Test environment can be scaled up or down by large factor as the need emerges	Optimizes, infra and software license usage
Flexible pricing	Test resources are priced on supported topology and project phases	Offering pricing options tailored to user resource need
Metering and billing	Test resources used in reserved are charged back to LOBs	Prioritizing innovative projects
Re-platformability (Compatibility Testing)	Platform based configurations (for testing) can be created and managed	Comprehensive assurance on product / service
Maintenance of multiple test beds based for multiple release testing	Release based configurations (for testing) can be created and managed	Complete assurance on maintenance of product / service
Test factories setups	Emerging alternative models for supporting client testing needs	Collaborative platform enables collective productivity
Service virtualization	In a multi component architecture, availability of a dependent component(s) managed for testing	Effective completion of component level testing, despite the dependencies on Critical components
ALM (Application Lifecycle Management) / TLM (Test Lifecycle Management) support	In a multi component architecture, availability of a dependent component(s) managed	Effective completion of component level testing, despite the dependencies on Critical components

Table 2: Testing Environment Provisioning [12][13]

Testing	Cloud Based Software Testing	Conventional Web Application Testing
Primary Testing Objectives	<ul style="list-style-type: none"> - Assure the quality of functions and performance of SaaS, Clouds, and applications by leveraging a cloud environment - Assure the quality of cloud elasticity & scalability based a SLA 	<ul style="list-style-type: none"> - Assure the quality of system functions and performance based on the given specifications - Check usability, compatibility, interoperability.
Testing and Execution Time	<ul style="list-style-type: none"> - On-demand test execution by third-parties; - Online test execution in a public cloud; - Offline test execution in a private cloud 	<ul style="list-style-type: none"> - Offline test execution in a test lab. - Testing a product before its delivery
Testing Environment	<ul style="list-style-type: none"> - An open public test environment with diverse computing resources - A scalable private test environment in a test lab. 	<ul style="list-style-type: none"> - A pre-fixed and configured test environment in a test lab with purchased hardware and/or software.
Setting the testing environment and configuration	- In minutes	- In days
Test Simulation	<ul style="list-style-type: none"> - Virtual/online user access simulation - Virtual/online traffic data simulation 	<ul style="list-style-type: none"> - Simulated online user access - Simulated online traffic data
Snapshots of multiple machines together (for productive cloning)	- Limited only by storage space	- Not Applicable
COTs Package image / configuration images can be leveraged to setup landscapes	- Limited only by storage space	- Not Applicable
Configuration parameters for the network / lab	- Can be customizable and can develop script to automate this	- Mostly manual – prone to errors
Software tools consolidation for large clients	- SaaS model enable this option – customer pays only for the usage	- Cost is high – infra + maintenance
Cases where applicable, more than one tool can be used	- Economical alternative	- Economically not viable
Testing Costs	<ul style="list-style-type: none"> - Based on a pre-defined service-level-agreement (SLA) - TaaS and Cloud testing service costs (pay-as-you-test) - Engineering costs in SaaS/Cloud application vendors 	<ul style="list-style-type: none"> - Required hardware costs and software (license) costs - Engineering costs in a test process