

# Bring Into Play the Software Metrics in SAAS - A Cloud Computing Prospective

Pallavi Narang<sup>1</sup>, Dr. I. S. Hudiara<sup>2</sup>, Dr. Sawtantar S. Khurmi<sup>3</sup>

Research Scholar, Department of CSE, Desh Bhagat University, Mandi Gobindgarh, Punjab, India<sup>1</sup>

Director Research, Chitkara University, Rajpura, Punjab, India<sup>2</sup>

Prof. Head, Department of CS, Desh Bhagat University, Mandi Gobindgarh, Punjab, India<sup>3</sup>

**ABSTRACT:** This paper deals with a kind of distributed computing over the internet known as cloud computing, which can run a program or application on more than one computer at the same time. It is divided further into three categories namely: (i) Infrastructure as a service (IaaS), (ii) Platform as a service (PaaS), and (iii) Software as a service (SaaS). Software as a service is a kind of service which provides much remuneration to the service consumers. What brings Cloud computing in the field of software is a complex issue. Software metrics as a subject area is over 30 years old, but it has barely penetrated into mainstream software Engineering. Software measurement is very important and delicate issue, especially when dealing applications into the practice. Software metrics provide a basis for planning and predicting software development. Quality in fact aids higher productivity, which has brought software metrics to the limelight.

**KEYWORDS:** Software Engineering, SAAS, Cloud Computing, Software Metrics.

## I. INTRODUCTION

Cloud computing supports on demand capability, scalability and delivery of services on pay-as-you-go basis. [8] According to [5], cloud computing is a shared pool of computing resources for on-demand networks. With minimal management efforts the resources like storage, services, servers and networks can be easily released and provisioned. [6] According to Wikipedia [7] cloud computing includes large group of computers those support Internet based computing, shares information, software, hardware and resources on demand. According to them cloud computing is: "A larger computing paradigm that is scalable and is driven by economies of scale, in which a pool of abstracted virtualized, storage, managed computing power, dynamically-scalable, services and platforms are delivered on demand to customers over the Internet in distributed environment". Also cloud computing can be used to support cooperative and remote e-learning [9].

Cloud computing is an online service model by which hardware and software services are delivered to customers depending upon their requirements. It involves delivering of the hosted services accessed through internet by the users. It has three services namely: Software as a service (SaaS), Platform as a service (PaaS), and Infrastructure as a service (IaaS). The services are sold on demand either by a minute or hour and the customer can access services as their wish for a particular given time. The services are managed by the service provider or companies like Amazon, Google, and IBM etc.[1]

Cloud can be public, private, hybrid and Community. In a public cloud, all the resources are owned by cloud provider and they sell the resources to public on demand. End users can rent required resources and pay as per usage. In a private cloud, cloud is rented and owned by an organization. The organization uses cloud resources for its private use only. These types of special clouds are personally built by an enterprise for serving their critical business processing needs. In a Hybrid cloud, it is the collaboration of two or more above mentioned cloud infrastructures (private, community, or public). The sole purpose of hybrid cloud is to provide extra services and resources to end users to serve their high demands. In a Community cloud, it is another type of Private cloud. But here cloud resources are shared among the members of a closed community having same resource requirements and interest. This type of community cloud may be operated by collaborate efforts of all or by a third party alone.

It is essential to introduce definition of software metrics. Software metrics provides measurement of the software

# International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2014

product and the process of software production. In this paper, the software product should be seen as an abstract object that begins from an initial statement of requirement to a finished software product, including source and object code and the several forms of documentation exhibited during the various stages of its development. [2]

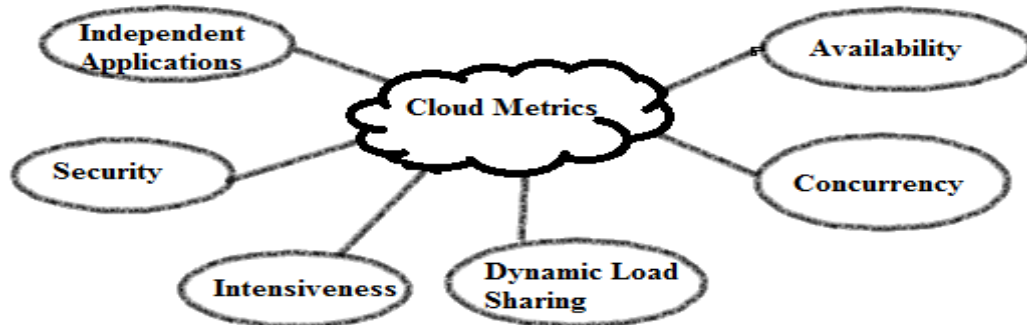


Fig.1 Measurement metrics of cloud [3]

Paper is organized as follows. Section II describes proposed study in the field of evaluating metrics for the cloud. A small overview of cloud computing is given in Section III. Section IV presents classification of software metrics along with the case study in Section V. Finally, Section VI presents conclusion.

## II. RELATED WORK

Empirical evidence exist showing that there exist a relationship between (many of) these metrics and software quality. However, with the growing complexity and size of Object Oriented software systems, the ability to reason about such a major issue using loads of metrics would be more appropriate in practice.

We proposed the study of a new metric, called Quality Assurance Indicator (Qi), capturing in an integrated different attributes of Object Oriented software systems such as coupling (*interactions between classes*), coercion and complexity. These Three C's form the backbone of Quality Assurance Indicator proposed. The Quality Assurance Indicator of a class is based on intrinsic characteristics of the class, as well as on the Quality Assurance Indicator of its collaborating classes. The metric has, however, no ambition to capture the overall quality of OO software systems. However, the objective is not to evaluate a design by giving absolute values, but more relative values that may be used for identifying critical classes on which more QA effort is needed to ensure software quality.

## III. OVERVIEW OF CLOUD COMPUTING

Cloud Computing has gained lot of recognition and has become a catchphrase in corporate world, with information technology as its backbone with full support. Lot of work is currently going on, in various dimensions of Cloud Computing still there are some research challenges prevailing in this area. Not only Cloud Computing has given an edge to enhance research in its own field but has a lot of prospects in other fields of research in collaboration. Research Issues may be categorized as Technological Issues and Business Issues and which are discussed one by one below [11]:

A: Technological Issues- This domain focuses on the cloud computing technology. Researches in this area are focused on inherent components and mechanisms driving it. Technological issues involve:

- 1) Scalability in Cloud Computing
- 2) Resource Allocation
- 3) Load Balancing
- 4) Data Security and locality

# International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2014

5) Ensuring Quality of Service by migration of Virtual Machines

B: Business Issues- This domain concerns the business models and implications of cloud computing technology. Researchers in this domain treat cloud computing as a black-box technology which can generate business value to both service providers and consumers. It mainly involves:

- 1) Return on Investment (ROI) Cost/Benefit
- 2) Pricing/Billing
- 3) Trustworthiness in the cloud

## IV. CLASSIFICATION OF SOFTWARE METRICS

There are three types of software metrics: process metrics, project metrics and product metrics. [4]

### 1) Process Metrics:

Process metrics highlights the process of software development. It mainly aims at process duration, cost incurred and type of methodology used. Process metrics can be used to augment software development and maintenance. Examples include the efficacy of defect removal during development, the patterning of testing defect arrival, and the response time of the fix process.

### 2) Project Metrics:

Project metrics are used to monitor project situation and status. Project metrics preclude the problems or potential risks by calibrating the project and help to optimize the software development plan. Project metrics describe the project characteristics and execution. Examples include the number of software developers, the staffing pattern over the life cycle of the software, cost, schedule, and productivity. [10]

### 3) Product Metrics:

Product metrics describe the attributes of the software product at any phase of its development. Product metrics may measure the size of the program, complexity of the software design, performance, portability, maintainability, and product scale. Product metrics are used to presume and invent the quality of the product. Product metrics are used to measure the medium or the final product. We can find more efficient ways of improving software project, product and process management. [2]

## V. THE BOEING 777: CASE STUDY

The cloud computing model especially the public cloud is unsuited to many business applications and is likely to remain so for many years due to fundamental limitations in architecture and design. Enterprises that move their IT to the cloud are likely to encounter challenges such as security, interoperability, and limits on their ability to tailor their ERP to their business processes [12]. In today's intensely competitive environment, traditional application systems such as ERP, lacks the autonomy and flexibility required by dynamic market. Given that, more and more companies are offering their software by Software as a Service (SaaS) application platform [13].

*The Boeing 777 project: A Case Study* – Boeing with its 777 airplane project was a giant leap forward in the direction of Software quality and is compelling case and point in the importance in reinforcing strong software quality management. With almost 2.5 million lines of code written for the new jetliner's state of the art avionics and other on board software, it was super critical to ensure best software quality practices and implementation. Complications like an extensive network of third party suppliers who would supply crucial components for the 777 made it a large challenge to ensure that deadlines are met without a compromise on software quality as a whole. [14]

*Key Takeaways* – Boeing realized early enough of the importance of enforcing a uniform set of metrics. Also vital learning from Boeing's experience is that done properly, enforcing software quality in a project ensures that program risk points can be identified early which would allow a reasonable time to apply corrective measures without delaying a

# International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2014

project indefinitely. Additional key points are the implementation of metrics allowed each project point to be having a check and balance so that the project flows smoothly without any major roadblocks. A good consequence of the metrics implementation was the streamlining and the regularity of communications between Boeing and its vendors, which was touted as being of equal importance to the metrics as well. Clear goals, milestones and constant monitoring of the key metrics around software design coding and testing made sure the 777 project was a success. [2]

Looking at rising demand for the implementation and successful case studies of software quality, it is safe to conclude that in the future time, software metric's importance will increase multifold due to change in technology which lead industry leaders to like embrace newer and more stringent approaches to monitoring, improving as well as delivering better software quality in products as well as processes. A number of metrics are proposed and exercised for measuring the quality of a system or software before implementation. Future research directions include improvement in existing metrics based on the nature and magnitude of the problem statement as per the growth of era towards cloud computing. There is a scope for various tools to support software project development reducing time, effort and cost of the project while increasing quality in consistent manner.

## VI. CONCLUSION

Even though there are number of software metrics so far, the main focus of all these metrics on how to improve the process of developing, maintaining and managing software. From the above discussion, a strong point is raised that metrics should be used in the Cloud Computing to reduce the number of errors in software being delivered using SAAS. Metrics have capability to identify potential problems that may lead to check out errors in the system and decreases over all development and pay-as-you go cost. The complexity of software will directly affect the maintainability, and reliability of the software. There is no adequate international standard for any of the extensively used software metrics till date. Absence of firm theoretic background and the assertion of methods, software metrics are still young in comparison of other software theories.

## REFERENCES

- [1] Banerjee Sarbojit and Jain Shivam, "A survey on software as a service (SAAS) using quality model in cloud computing", *International Journal Of Engineering And Computer Science*, Vol. 3, Issue 1, Jan 2014, pp 3598-3602.
- [2] Rawat Mrinal, Mittal Arpita, Dubey Sanjay, "Survey on impact of software metrics on software quality", *International Journal Of Advanced Computer Science and Applications*, Vol. 3, Issue 1, 2012 .
- [3] Singh Gurudev et.al, "CM – management facets for cloud performance", *International Journal Of Computer Applications*, Vol. 23, No. 3, June 2011.
- [4] Tu Honglei, Sun Wei, Zhang Yanan, "The Research on Software Metrics and Software Complexity Metrics", *International Forum on Computer Science-Technology and Applications*, 2009.
- [5] Donlin Chen, Mingming Ma, and Qiuyun Lv, "A Federation Model for Education under Hybrid Cloud Computing", Elsevier, 2nd International Conference on Future Computers in Education, Vols.23-24, PP 340-343, 2012
- [6] Aida Ghazizadeh, "Cloud computing benefits and architecture in e-learning", Seventh IEEE International Conference on Wireless, Mobile and Ubiquitous Technology in Education, pp. 199-201, 2012.
- [7] [https://en.wikipedia.org/wiki/Cloud\\_computing](https://en.wikipedia.org/wiki/Cloud_computing)
- [8] I. Foster, Y. Zhao, I. Raicu, and S. Lu, "Cloud Computing and Grid Computing 360-Degree Compared", IEEE Grid Computing Environments Workshop, pp. 1-10, 2008.
- [9] LIN Yu- hua, "Practice and Innovation of an experimental teaching mode in universities under environment of cloud computing", Research and exploration in laboratory, vol. 30, pp. 271-274, 2011.
- [10] <http://www.pearsonhighered.com/samplechapter/0201729156.pdf>.
- [11] Singh A, Malhotra M, "Agent Based Framework for Scalability in Cloud Computing", IJCSET, Vol. 3, Issue 4, 2012, pp. 41 - 45.
- [12] Hofmann, P. SAP Labs., Palo Alto, CA, USA Woods, D., "Cloud Computing: The Limits of Public Clouds for Business Applications", IEEE Conference, Dec. 2011.
- [13] Xiaoyan Jiang Yong Zhang; Shijun Liu, "A Well-designed SaaS Application Platform Based on Model-driven Approach", IEEE Conference, Nov. 2010.
- [14] Lytz, R., "Software Metrics for the Boeing 777: A Case Study", Software Quality Journal, Issue 4, 1995, pp 1-13.