

# Reviewing Nist Cloud Computing Definition

**Danko Naydenov**

*Eurorisk Systems Ltd.*

*31, General Kiselov Str., 9002 Varna, Bulgaria*

*E-mail: sky at eurorisksystems dot com*

**Abstract:** *The main goal of this paper is to analyze the National Institute of Standards and Technology (NIST) [4] definition of Cloud Computing. This include analysis of main definition also analysis of Essential Characteristics, Service Models and Deployment Models. At the same time a comparison of these definitions with real services offered on the market is made. As a consequence, comes extension and clarification of certain definitions.*

**Key words:** *Cloud computing definition, SaaS, PaaS, IaaS, EaaS*

## 1. INTRODUCTION

At present there still is no single and generally accepted definition of Cloud Computing. This is large difficulty for theoretical research in this area. For this reason, scientists often stop at one particular definition and use it in their researches.

## 2. OBJECTIVES AND TASKS

One of the most commonly used and most popular definition is that of NIST [5]. Defined goal is to analyze the correctness and relevance of this definition. To achieve this goal there are formulated several tasks specifically related to this definition:

- to verify the correctness of Essential Characteristics;
- to verify the correctness of Service Models;
- to verify the correctness of Deployment Models;

## 3. ANALYSING ESSENTIAL CHARACTERISTICS

Essential Characteristics are too specific and certain cloud vendor can focus on one or several of them and at the same time the vendor cannot meet others of these

characteristics. For instance, if the service is cloud based web email, the customers cannot use “On-demand self-service” or “Rapid elasticity” but at the same time customer can monitor his current usage of the resources by using “Measured service” and provider apply the “Resource pooling”.

The problem with characteristics definition is that it is not specified if all five Essential Characteristics need to be met. As it was demonstrated that there are services which are cloud based and do not meet all Essential Characteristics. On the other hand, there are serviced that meet some of Essential Characteristics, but they are not cloud service.

The conclusion which can be made is that it is possible to identify if some service is cloud base if this service meets some of Essential Characteristics, but it is not sufficiently.

#### **4. THE SERVICE MODELS**

The summarize NIST definitions of Service Models are [5]:

- SaaS - this is model where the customer receives final service, which can be used either with tin clients like web browsers or by using application programing interface (API);
- PaaS - this model offers to the customer environment which can be used to deploy applications. This environment has its own specificity and characteristics which can be significantly different from well-known on premises;
- IaaS - this model offers to the customers basic computing resources as processor, memory, storage, network and others;

According to these definitions, IaaS model represents only virtual machine. Operating system is also program and taking into account this, virtual machine with running on it operating system should be classified as PaaS. A simple research will show that almost all IaaS providers offer not only virtual machine but also an operating system for it [1]. This has its explanation, namely because these operating systems are optimized for usage in offering service from vendor.

On the other hand, these are PaaS providers that offer specific environment for deploying applications, along with certain database servers where applications can store data, and many other services, but in the end of configuring process of cloud service, a dialog for choosing an IaaS provider is appear and this IaaS provider will be used for real computing processes [3].

After summarizing all possible cases, it is clear that difference between IaaS and PaaS is very fuzzy. It is possible a given service to be promoted as PaaS but in fact to be IaaS and vice versa. All this leads to the conclusion that it is possible to be defined a new additional layer disposed between the two services.

Let assume that the name of this new layer is Extended Infrastructure as a Service (EIaaS).

Let assume that the NIST definition of IaaS is correct.

This means that everything that is different, as a service from virtual machine should be assigned to the new service. This will define the difference between IaaS and EIaaS.

There is a need to define the difference between EIaaS and PaaS.

Let assume that there is an environment which has been used to deploy application and this application is running on particular virtual machine. In this case with increasing the number of users the current computing resource is sharing among all users. This leads to increasing the response time of any user compared to response time in case of single user (Fig 1a). This is command model for many providers that promote themselves as PaaS [3].

In contrast to describe model there are PaaS providers offering such a service which guarantee that each user is supplied with dedicated computing unit [1]. This computing unit is not a virtual machine, but it is a virtual processor and memory. In this case the response time is not affected by the number of simultaneous working users (Fig. 1b).

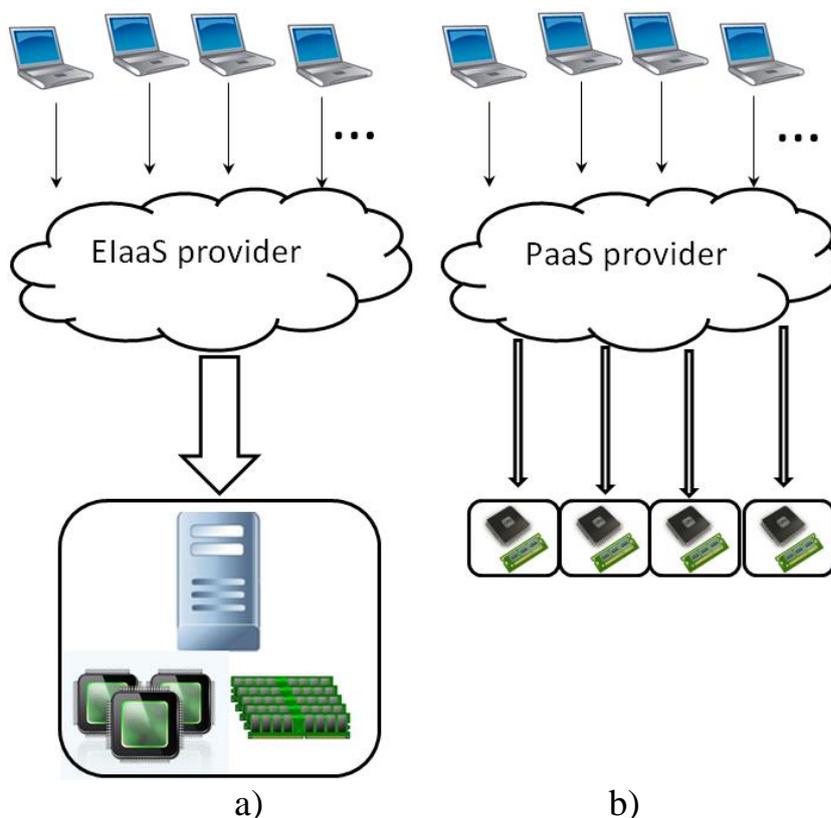


Fig. 1. EIaaS and PaaS providers model

As a result, the following formal definition can be given:

- if there is functional dependence between the number of users and performance of each one of them (there is single particular virtual

machine which resources are shared among all users) then this is EIaaS (1a);

- if there is not functional dependence between the number of users and performance of each one of them (for each user it is allocated independent computing unit) then this is PaaS (1b);

$$\text{a) } \exists UP = f(UN) \quad \text{b) } \nexists UP = f(UN) \quad (1)$$

where

UP is user performance

UN is user number

## 5. THE DEPLOYMENT MODELS

According NIST Private cloud model is when infrastructure is exclusively used by single organization. Community cloud model is when infrastructure is used by many organizations that have shared concerns. Basically, these different organizations are interconnected and very often they have the same owner. An example of such a situation occurs quite frequently in car industry where different brands, which are separate organizations, belong to one big holding.

From this point of view the difference between Private and Community clouds is concluded only in number of users and today's Private clouds are yesterday's Community clouds and today's Community clouds are tomorrow's Private clouds. In other words, Private and Community clouds by their nature are the same thing.

Public cloud model in terms of the number of users could be attributed to previous models just in the present model the number of users is still greater. But there are some major differences, and this is the fact that Public clouds are accessible for everyone. Also, Public clouds are owned not by companies but by big corporation or even government organizations. This is a very important characteristic because these types of clouds are planned and organized to hold vast amount of infrastructure.

Special attention should be given to the amount of infrastructure in the Private or Community clouds. In these cloud models when it is necessary additional computing resources, there is a very controversial situation. On one hand, to ensure this resource it is necessary to buy corresponding hardware, to install it, to locate it and all this can lead to additional difficulties such as lack of space, lack of network resources (available ports on network device) also there can be difficulties to configure new hardware to work with existing infrastructure. On the other hand, according the basic definition of cloud computing this computing resource has to be rapidly provisioned. From this perspective, and taking into account one of Essential Characteristics of cloud environments (Rapid elasticity which gives to the consumer, the capabilities available for provisioning often appear to be unlimited and can be appropriated in any quantity at any time [5]), the question arises whether Private and Community models are at all cloud models after they do not meet the main definition and some of

Essential Characteristics. Public model, as it was noted, is owned by large companies or government organizations, and because of that the amount of existing infrastructure is very big and this allows unlimited rapid scaling. Therefore, the Public model is the only model that meets the definition of Cloud computing.

Considering that Private and Community models are not cloud models and only Public cloud meet the definition then the hybrid model takes a slightly different meaning. The Hybrid models should include mandatory in themselves Public model in combination with some of the other models. This will combine local computing resources with cloud resources to solve particular task.

## 6. CONCLUSION

Because of the wide variety of cloud services and due to constant proposing new ones it is very difficult to make concrete definition of something so dynamic with respect to its characteristics. The current paper attempts to specify some of the already made definitions depending on the current provision of real services in this area, but this run the risk with development of cloud computing to become obsolete definitions which was proposed. From this perspective, it is possible an approach in which instead of making precise definition to attempt to make more abstract definition that eliminates specific representations.

## REFERENCES

- [1] Charles Severance (May 7, 2009). Using Google App Engine. *Publisher: O'Reilly Media.*
- [2] Jurg van Vliet, Flavia Paganelli (March 14, 2011). Programming Amazon EC2. *Publisher: O'Reilly Media.*
- [3] Matthew Nohr (Author), Isaac Rabinovitch (November 19, 2013). Implementing AppFog. *Publisher: Packt Publishing.*
- [4] <http://www.nist.gov>
- [5] <http://www.nist.gov/itl/csd/cloud-102511.cfm>