2015 27th International Symposium on Computer Architecture and High Performance Computing

Cloud Services Evaluation through QoE: a methodological approach

Frederico G. I. da Costa
Institute of Electronics and Telecommunications - PUCRS
Porto Alegre, Brazil

M. Cristina F. de Castro
Institute of Electronics and Telecommunications - PUCRS
Porto Alegre, Brazil

Candice Muller
Institute of Electronics and Telecommunications - PUCRS
Porto Alegre, Brazil

Fernando C. C. de Castro
Institute of Electronics and Telecommunications - PUCRS
Porto Alegre, Brazil

***Abstract* — Cloud computing has been touted as a revolutionary concept in computing in the Information Age, since it enhances the quality of communication and it is highly cost-effective. Cloud computing market has attracted the interest of several providers and corporations, creating an environment in which the user´s Quality of Experience (QoE) becomes a competitive advantage. Cloud services are often available as Web applications, since Web browsers may provide a more user-friendly interface. Thus, Web Application Response plays a critical role in the perceptions of cloud service users. This article proposes a methodology to evaluate the user Quality of Experience of cloud services, with focus on web applications, using the MOS score in a user-centered approach. This methodology estimates the QoE from the end-to-end response time and adjusts the estimated score according to the evaluation context, thought maximum session time. Estimation of QoE is a differentiating factor in choosing cloud service providers and defining the form of implementing cloud applications (e.g., through programming language, page type or application server). The proposed methodology has been applied to three cloud service servers, located in Brazil, Europe and USA and several case studies in business contexts have been evaluated, comparing different clients and server applications in a monitored environment. The results point to the crucial role that the evaluation period plays in the comparison of solutions.**

Keywords— QoE; cloud services; cloud computing; web application; quality of experience; MOS.

I. Introduction

The growth of the global market for Information Technology (IT), together with the growing dependence of the modern society for services of this kind, has generated an increased demand for cost-effective solutions. Cloud Computing characterizes an approach that uses shared resources and seeks to deliver the maximum number of benefits at the lowest possible cost.

In this new scenario, the user-perceived quality of experience is considered a crucial factor in service migration to the cloud environments.

Cloud services are often available as web applications and web services. Web services are used as an interface between the cloud service solutions and the users [1] while web browsers provide a friendly interface to interact with users in a flexible environment [2].

The demand for cloud services has been met by an increasing amount of Cloud Service Providers (CSPs). This scenario has consolidated an environment of competition and pressure on prices. While IT environments and applications were under the management of organizations and often directly connected to their headquarters through a local network service, there was no competition (and not even a performance standards) to compare applications. The migration of services to the cloud has set up the beginning of a competitive market in which the users' perception on provided services has been gaining attention.

This competitive scenario has brought a parallel interest in the quality of the users' experience [3]. Cloud service providers need to understand what the users perceive on the services being delivered [2] because, insofar as personal and business applications migrate to the cloud, the perceived quality of service becomes an important differential among the providers [4]. In this new scenario, the QoE concerning the cloud service experienced by the users is considered a crucial factor referring to the migration to the cloud, once different providers and solutions can be compared.

This article proposes a methodology for evaluating the Quality of Experience (QoE) of cloud service users, focusing on web applications, through the MOS score. Estimates of QoE is a key differentiating factor in the choice of cloud service providers, and it also helps to shape the implementation of cloud applications (e.g. through programming language, page type or application server).

The proposed methodology has been implemented, considering three cloud service servers located in Brazil, Europe and the USA, and several case studies have been evaluated in commercial situations by comparing the different client and server applications in monitored settings. The results have shown the importance of session time to estimate the QoE (compared to network parameters such as latency and download rate) and explained objectively by a score the extent of impact of geographical location of the service provider, programming language and page type on the quality related to the cloud service users' experience.

This paper is designed as follows: in section II, the quality of the experience is shown in cloud setting, and the QoE mapping (MOS score) is shown through the network and application criteria. Section III presents the proposed methodology for estimating the QoE. Section IV discusses case studies. Section V presents the conclusion.

1550-6533/15 $31.00 © 2015 IEEE
DOI 10.1109/SBAC-PAD.2015.30

II. QoE AND CLOUD COMPUTING

The growth of cloud-based services, such as DropBox, Youtube, Google Mail, among others, poses new challenges for both users and providers. These challenges need to be addressed to ensure the adoption of this new paradigm, represented by virtualization and cloud services [4]. Competition from providers is based mainly on service cost reduction, and it is pressuring down the providers profit margins in this type of solution. Thus, the providers need to consider another differentiating factor rather than pricing alone. Therefore, there is the evaluation of the quality of services provided under the users' perception. Quality of Experience is the performance of the whole system according to user's perspective, and it is an important tool in understanding the judgment of value [5, 6].

However, assessing users' perception is not a simple task. Some studies have shown that there is a logarithmic relationship between the waiting time and the scale of the users' satisfaction, i.e., user satisfaction is not linearly related to the waiting time in simple interactions services, [7, 8, 9, 10].

Wang [3] has proposed the construction of an improved prediction model of QoS, under the infrastructure point of view, through the efficient allocation of the distributed data center clusters, but he did not assess the role of applications (both user and server) in this model .

Jarschel [11] went ahead and was in charge of studies about virtualization of game consoles (video games) through a WAN simulator, and also assessed the impact of packet losses on the users' perceived quality.

Yao [2] has set up a platform for the evaluation of performance and QoS, taking into account the application layer (web applications), but the evaluation did not include the users' perception. Moreover, the author evaluated only a real server, without testing its approach in commercial situations (virtual machines) in services and in operating scenarios of cloud providers.

Casas [12, 13] has conducted a detailed assessment of users' QoE, based on the Personal Cloud Storage and File Synchronization applications (CSFS) such as the Dropbox [12] and Remote Virtual Desktop [13]. However, the author did not evaluate web applications in commercial servers.

The aim of this paper is to address this gap, by suggesting a methodology for evaluating the QoE in cloud services using performance parameters. This methodology represents the network, the server and the client application. This methodology provides a score to compare providers, programming language and page type. With this methodology, providers and users will have an objective measure to compare different solutions.

QoE is related to the users' perception and expectations to an application and network performance, typically expressed by QoS parameters [14]. Several studies have discussed wich would be the most sensitive network parameters, and how these would affect the user QoE [10, 14, 15]. QoE can be expressed as the Mean Opinion Score (MOS), which is a numerical indication of the perceived quality, being widely chosen as a result of subjective tests for modeling objective quality [16]. Mean Opinion Score is a subjective scale of evaluation, involving five possible values, according to the ITU-T P.800 recommendation [17].

Several QoE evaluation studies have been carried out using different approaches, suggesting two main hypotheses about the relationship between QoS and QoE.

The first hypothesis is that IQX, which indicates the relationship between QoS and QoE parameters, is exponential. Usually this hypothesis is adopted when the QoE is modeled with network parameters (e.g. packet loss) [14].

The second hypothesis is based on the Weber-Fechner's law of psychophysics, which sees the relationship between the performance parameter and QoE as logarithmic. Usually this hypothesis is adopted when QoE is modeled using the page or session loading time [8].

III. CLOUD SERVICES AND QoE RATING METHODOLOGY

Fig. 1 shows the suggested methodology to estimate QoE of cloud service

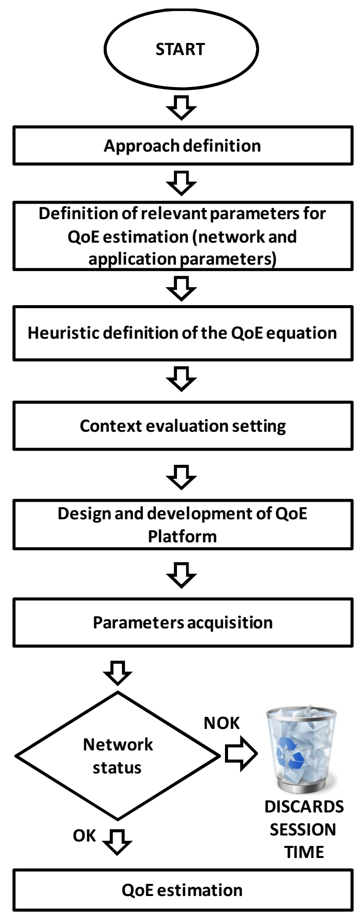

Fig. 1. Methodology to estimate cloud services QoE

A. *Approach Definition*

Cloud services are often provisioned and made available as web applications and accessed through web browsing;

therefore, the suggested methodology is designed to evaluate cloud services from web browsing and applications.

Studies carried out in the laboratory have shown that centric metric QoS parameter has a direct correlation with the QoE [18, 19], suggesting that QoS parameters are enough to estimate web QoE [15]. Thus, the approach we adopted is based on network and application parameters, which makes the methodology applicable to different scenarios.

B. Definition of the relevant parameters (network and application) to QoE

Hosfeld [20] has pointed the technical factors impacting web QoE: bandwidth, the page loading time, packet loss and browser type. These factors are related to network parameters and application parameters. When applied to a cloud setting, session time stands out among other parameters as the most suitable for capturing network factors and application factors (client and server). Session time refers to an end-to-end response time, as shown in Figure 2. It is capable of quantifying the response time referring to client, server and network, which, from the users' point of view, represents the time elapsed to access hosted applications to cloud providers.

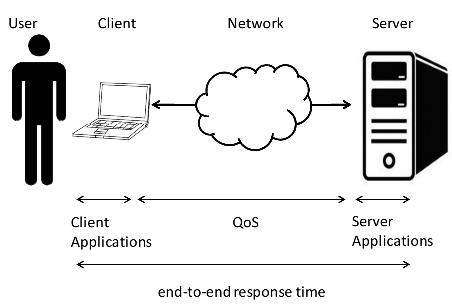

Fig. 2. End-to-end response time

Thus, the suggested methodology uses the session time parameter for estimating the QoE. The session time is obtained by the end-to-end response time for loading a web page and indirectly counting network QoS parameters (such as latency, bandwidth and packet loss), response time to clients' applications (applications available in browsers) and response time to servers hosted in the cloud (such as web servers).

C. Heuristic definition of the QoE equation

QoE equation needs to be generalizable enough to suit different application scenarios. The application scenario, which sets the context for the user's expectations, should be considered in order to match an estimated users' context perception. Studies have indicated that modeling the perceived quality is dominated by the maximum session time [19]. Thus, the QoE equation, that enables the adjustment of the maximum time session according to the application scenario, is required.

ITU-T G.1030 [19] has put forward an equation for simple time event logarithmic form. The parameters considered in addressing this equation have made it possible to adapt to the context of the user's expectations. This equation was designed to estimate MOS score, adjusting simple time events perceived by the users as instantaneous to **5** (maximum score), and score **1** (worst case) to the maximum session time observed in the evaluation context. The MOS, which quantifies the QoE, is defined by:

$$\mathrm{MOS} = \frac{4}{\ln\left(\frac{0.003\mathrm{Max}+0.12}{\mathrm{Max}}\right)} * (\ln(\mathrm{t}) - \ln(0.003\mathrm{Max} + 0.12)) + 5 \quad (1)$$

Where: t is the session time (s) and Max is the maximum session time (s) of evaluation context.

Eq. (1), as proposed by the ITU-T P.1030 recommendation, takes into account end-to-end performance, allowing context adjustment to the users' expectations by setting the maximum session time.

The t session time (end-to-end response time) represents the network and application parameter used to map the QoE, including:

• The client's application response time,

• The time for the content availability by the remote server and

• The network response time.

This equation is not able to capture all the possible aspects of Human-Computer Interaction, though it is able to estimate the user experience considering network issues, client application and its backend servers.

In the methodology we propose, Max , the maximum session time, is set in order to capture the evaluation context in the worst case (highest response time), and it is associated with lower MOS score (score 1). The adjustment of the maximum session time, Max, in accordance with the evaluation context, is crucial since modeling the perceived quality of web browsing is dominated by this parameter. For example, the perceived quality of a 5-second-session would be much higher if the maximum session time were 50 seconds than in it would be in the case this maximum session time were 4 seconds [19].

D. Evaluation context

The evaluation context is the comparative setting on which the quality of experience evaluation is estimated. In the suggested methodology, for each assessment context, changes are expected in the maximum session time, in order to adjust the methodology to the users' expectations.

The evaluation context is defined from an initial scenario where a web application is hosted on a cloud server in order to be accessed through the internet by a browser. From this initial setting, context variables are defined, i.e., variables that promote changes in the scenario. Thus, the context variables define the comparative context for estimating QoE.

The methodology proposed in this paper aims to provide subsidies for users and cloud service providers to choose the way of implementation and delivery of its services to users. Thus, the following items are defined as context variables:

• The language used in the implementation of the client's application;

• The type of website page deployment on the server;

• The geographical location of the cloud service provider.

E. Design and development of QoE Platform

The use of browsers has been growing increasingly, not only in interactive applications, such as online games and video streaming, but also in wide provisions for cloud services and access. This extensive use of browsers is motivated by their availability in the various devices with internet access, such as tablets, Smartphones and computers. Developers seek to improve the quality of their applications hosted in the cloud, whereas users try to estimate the quality of cloud services being offered. Both are interested in estimating the quality of the user experience, and need simple tools and methods to provide an estimated perceived performance.

In this context, the platform proposed by this methodology was designed to be accessed through a web browser, which makes it possible to estimate the QoE in a simple and highly accessible manner. The platform user accesses the application available in the browser, which requests a page to the cloud server. The server receives the request and provides the requested page to the client. The time duration from the page request until it is fully available on the browser (end-to-end response time) is recorded and stored by the platform.

F. Parameters acquisition

The QoE platform performs data collection for assessing both the quality of experience, and the state of the network during the collection period. Session time is collected to estimate QoE and the following parameters are used to assess network status:

• download rate available between the server and the client;

• RTT latency between the client and the server in the cloud;

• Loss of packets between the client and the server in the cloud.

Assessment of network status is based on the collected network parameters. This step is important to identify any abnormalities that may mask the results of QoE, negatively affecting the assessment.

G. Network status validation

Session time data need to be validated before being used to estimate the QoE of cloud service. Network abnormalities can distort the QoE obtained from data collected during the observation period.

Data validation is conducted by determining acceptable performance thresholds of network performance parameters (download rate, RTT latency and packet loss). If the thresholds are met, the network status will be considered as normal and the session time is stored to be used in the estimation of QoE. If the thresholds are not met, it is assumed that the network has an abnormality, in which case, the session time will be discarded.

H. Estimates of QoE

The collected and stored session time done by the evaluation platform are used for estimating the QoE through Eq. (1) , which relates the MOS score to the session time, in accordance with the evaluation context.

The estimation of the quality of the users' experience through the MOS score is key both to decision making and to compare providers and implementation modes, as it offers the sensitivity of the changes in users' perception adjusted to the context of the evaluated possibilities.

IV. RESULTS

The QoE estimation methodology proposed in this work has been implemented, considering the following context variables:

(1) The language used in the Client's Browser:

- JavaScript,
- Flash Action Script

(2) The type of web page implemented in the cloud server:

- Static
- Dynamic in PhP,
- Dynamic in Python,

(3) The geographical location of the cloud service provider:

- Server located in the United States, in the region of Kansas (USA),
- Server located in the Netherlands, in the region of Amsterdam (Europe),
- Server located in Brazil, in the region of São Paulo (Brazil).

The case studies are grouped in four evaluation contexts, and each context considers the variables presented in Table 1.

TABLE I. EVALUATION CONTEXT

EVALUATION CONTEXT NUMBER	CONTEXT VARIABLE		
	Client Application	***Server page type***	***Cloud server location***
1	JavaScript Flash	Static PhP Python	USA
2	JavaScript	Static PhP Python	Brazil USA EUROPE
3	Flash	Static PhP Python	Brazil USA EUROPE
4	JavaScript Flash	Static PhP Python	Brazil USA EUROPE

Each context consists of a set of cases which are determined from the combination of context variables. Table 2 shows the evaluated cases.

TABLE II. CASES EVALUATED

CASE N#	CONTEXT VARIABLE			EVALUATION CONTEXT N#
	Client Application	***Server page type***	***Cloud server location***	
1	JavaScript	Static	Brazil	2, 4
2	JavaScript	PhP	Brazil	2, 4
3	JavaScript	Python	Brazil	2, 4
4	Flash ActionScript	Static	Brazil	3, 4
5	Flash ActionScript	PhP	Brazil	3, 4
6	Flash ActionScript	Python	Brazil	3, 4
7	JavaScript	Static	USA	1, 2, 4
8	JavaScript	PhP	USA	1, 2, 4
9	JavaScript	Python	USA	1, 2, 4
10	Flash ActionScript	Static	USA	1, 3, 4
11	Flash ActionScript	PhP	USA	1, 3, 4
12	Flash ActionScript	Python	USA	1, 3, 4
13	JavaScript	Static	EUROPE	2, 4
14	JavaScript	PhP	EUROPE	2, 4
15	JavaScript	Python	EUROPE	2, 4
16	Flash ActionScript	Static	EUROPE	3, 4
17	Flash ActionScript	PhP	EUROPE	3, 4
18	Flash ActionScript	Python	EUROPE	3, 4

For each country we used a different provider and All providers were hired with the same vCPUs, memory, disk, and a bandwidth provision of 10 Mbps. In client side (located in Brazil) the internet access is provided via an ADSL of 10 Mbps. Case studies were conducted with Google Chrome on client side. On the server side we used Nginx. Both sides were configured to not make any cache.

As defined in the methodology, before estimating QoE, the validation of the collected session time data is performed by evaluating the previously defined network parameters limits. For data collection in this paper, the following limits for the network parameters have been defined:

• A maximum packet loss of 2% during the evaluation period,

• A maximum rtt latency of 500 ms during the evaluation period

• A minimal download rate of 100 Kbps

The evaluation period of each case presented was a week, with a one-hour interval between collections.

As expected, the MOS score obtained did not have a single value throughout the observation period due to time variations session. Session time is influenced by network and application issues. Fig.3 and Fig.4 show the estimated MOS for Case 4 in Context 4 versus latency and download rate respectively. The area highlighted in red in Fig.3 and Fig.4 shows the importance of observing session time to estimate the QoE of cloud service. Variations of MOS observed in these regions could only be identified by a parameter, such as session time, since latency and download rate (parameters commonly used to estimate MOS) remained constant. The session time variations observed in the highlighted region may result from several factors, associated with either client application or server (e.g. quality and type of the provider's disc, resource sharing on provider, type and hardware status of the provider or client, stability, occupation of shared resources, etc.).

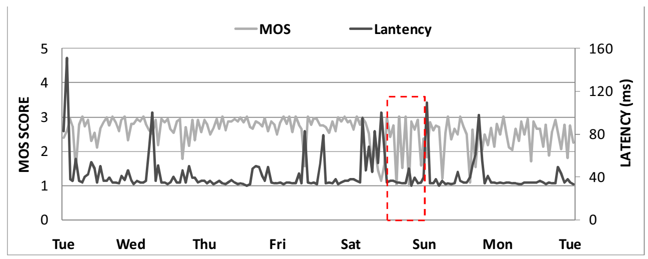

Fig. 3. MOS case 4 versus Latency

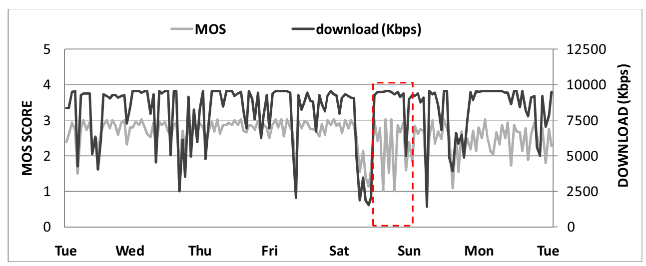

Fig. 4. MOS case 4 versus Download

From the data stored and validated, the average session time was calculated for each case and then the MOS is calculated. Fig. 5 shows the MOS score obtained by the average session time of cases evaluated in the Context 1 (cases 7 to 12) which aims to compare the different types of pages on a single server located in the USA for the applications of JavaScript and Flash Actionscript. The maximum average session time in this context was 4.4419 s.

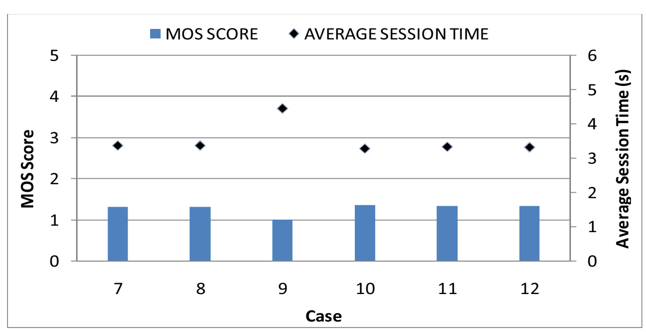

Fig. 5. Context 1 MOS score versus average session time.

Estimated average MOS scores for cases related to Context 1 were below 1.5, which is equivalent to an assessment between Bad and Poor. This score is consistent for a session time between 3 and 4 seconds, compared to the perception of instantaneous time.

However, it is important to note that the choice of a particular Cloud service solution should not be based solely on average MOS score, since the MOS varies significantly according to the network, client applications and provider performance, as illustrated by the behavior that occurred in all cases and in all contexts and was illustrated in Figures 3 and 4. Thus, the observation of MOS score behavior is critical throughout the evaluation period, and especially on the user interest period.

Fig.6 shows the average MOS in each context variable, which is calculated by the arithmetic mean of cases of same context variable. In Fig.6 may be observed that in customers' applications, Flash ActionScript (cases 10, 11 and 12 - Average MOS 1.3403) has a better performance when compared to JavaScript (cases 7, 8, and 9 - Average MOS 1.2133). When compared the page type on the server, static pages (cases 7 and 10 - Average MOS 1.3357) had the best performance, followed by dynamic pages PhP (cases 8 and 11 - Average MOS 1.3261) and dynamic pages in Python (cases 9 and 12 - Average MOS 1.1685).

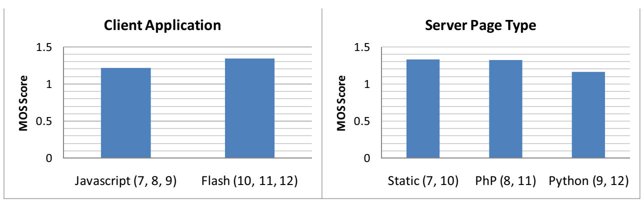

Fig. 6. Context 1 MOS versus Context Variables

Context 1 has compared the client and server applications on a single provider, where highlights the variation of the MOS score over the evaluation period and the best QoE case with client application implemented in Flash, which accesses static pages hosted the cloud service provider in USA.

Fig. 7 shows the MOS score obtained by the average session time of the cases evaluated in Context 2, comparing the different page types on servers located in BRAZIL, USA and EUROPE for JavaScript application. The maximum average session time in this context was 5.1020 s.

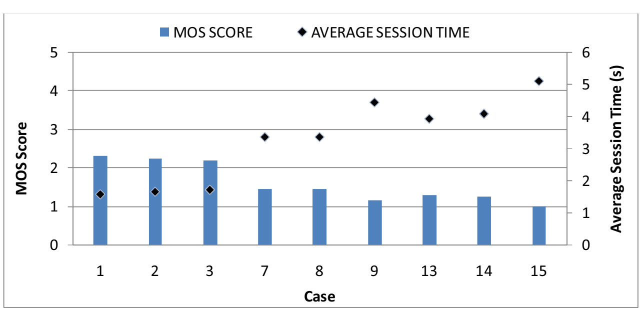

Fig. 7. Context 2 MOS score versus average session time

Context 2 presented scores with significant variations among the cases evaluated. This behavior is explained by the variation of server geographical location in relation to access and highlights the impact of this context variable in the QoE.

Fig.8 shows the average MOS in each context variable and demonstrates that, in relation to the geographical location of the CSP, the provider located in Brazil (cases 1 to 3 - Average MOS 2.2514) had a better performance when compared to the one located in the USA (cases 7, 8 and 9 - Average MOS 1.3587) followed by the provider located in Europe (cases 13, 14 and 15 - Average MOS 1.1780). When the page type is compared on the server, static pages (cases 1, 7 and 13 - Average MOS 1.6840) had the best performance, followed by the dynamic pages PhP (cases 2, 8 and 14 - Average MOS 1.6514) and the dynamic pages in Python (cases 3, 9 and 15 - Average MOS 1.4528).

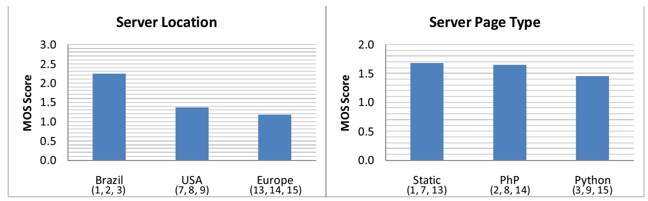

Fig. 8. Context 2 MOS score versus Context Variables

Context 2 has showed servers in different geographical locations, different type of page on the server accessed through the client application JavaScript. In this Context, Case 1 that implements static page in server located in Brazil is highlighted, with an average MOS 2.3011.

Fig. 9 shows the MOS score obtained by the average session time of the cases evaluated in Context 3 which tries to compare the different page types on servers located in BRAZIL, the USA and EUROPE for the Flash application. The maximum average session time in this context was 3.8753s. The variation of MOS score in Fig. 9 shows the crucial role that the geographical location of the service provider plays on the users' quality of experience.

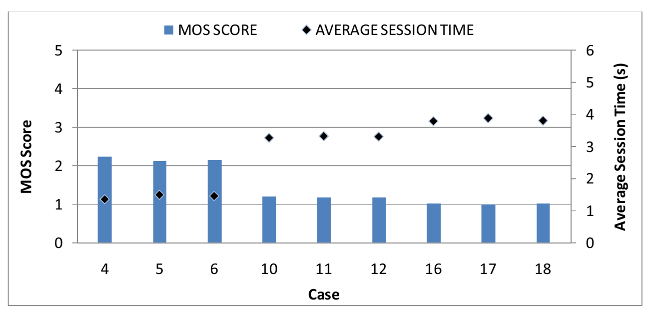

Fig. 9. Context 3 MOS score versus average session time

Fig.10 shows the average MOS for each context variable, and demonstrates that, in relation to the geographical location of CSP, the provider located in Brazil (cases 4 to 6 - Average MOS 2.1740) had a better performance when compared to the one located in the USA (cases 10, 11 and 12 - Average MOS 1.1914) followed by the provider located in Europe (cases 16, 17 and 18 - Average MOS 1.0172). When compared to the type of page on the server, the results were very similar, static pages (cases 4, 10 and 16 - Average MOS 1.4902), dynamic pages Python (cases 6, 12 and 18 - Average MOS 1.4560) and dynamic pages in Php (cases 5, 11 and 17 - Average MOS 1.4365).

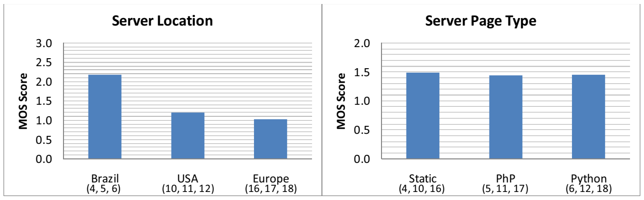

Fig. 10. Context 3 MOS score versus Context Variables

Fig. 11 shows the MOS score obtained by the average session time of cases evaluated in Context 4 that tries to compare the different page types on servers located in BRAZIL, the USA and EUROPE for JavaScript and Flash ActionScript applications. The maximum session time in this context was 5,1020s. This context showed great variation in the MOS score mainly associated with the geographical location and to the client application.

Fig. 11 shows the performance of MOS score compared to the session time for each assessed case.

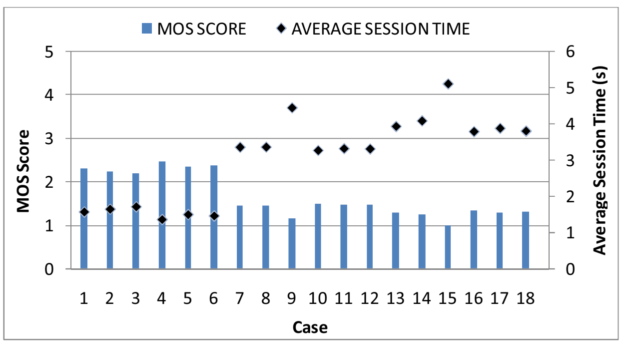

Fig. 11. Context 4 MOS score versus average session time

Fig.12 displays the average MOS for each context variable, and shows that, in relation to the geographical location of CSP, the provider located in Brazil (cases 1 to 6 - Average MOS 2.3242) had a better performance when compared to the located the USA (cases 7 to 12 - Average MOS 1.4201), followed by the provider located in Europe (cases 13 to 18 - average MOS 1.2486). When comparing page type on the server, static pages (cases 1, 4, 7, 10, 13 and 16 - Average MOS 1.7219) had the best performance, followed by dynamic pages PhP (cases 2, 5, 8, 11, 14 and 17- Average MOS 1.6806) and by dynamic pages in Python (cases 3, 6, 9, 12, 15 and 18 - Average MOS 1.5904).

The client application, Flash ActionScript, (cases 4,5,6,10,11,12,16,17,18 - Average MOS 1.7325) has shown a better performance followed by the JavaScript clients' application (cases 1,2,3,7,8,9,13,14,15- average MOS 1.5961), which confirmed the findings of Yao [2] who identified a faster response time for flash applications, when compared to JavaScript.

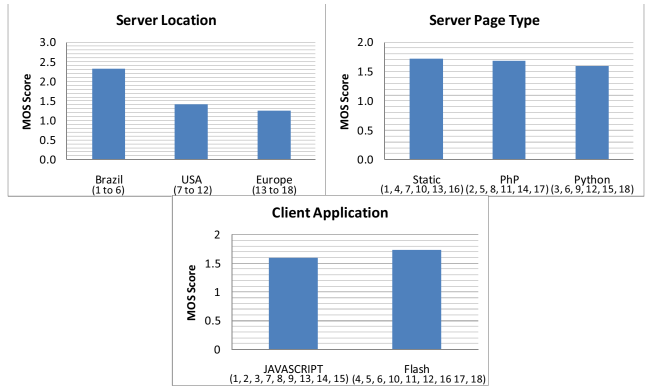

Fig. 12. Context 4 MOS score versus Context Variables

From data presented in Context 4 we conclude that the geographic location of the CSP, the page type on the provider and the client application impacted the MOS through evaluated cases. This context allowed evaluate all the context variables indicated in this study in a single comparative scenario, where we highlight the average MOS results Case 4 (Brazil / Flash / Static - Average MOS 2.4583).

V. CONCLUSIONS

This paper proposed a methodology for assessing the quality of experience in cloud services, based on an approach focused on web applications. Cloud computing is a current topic of great importance to ICT (Information and Communication Technologies); indeed, it is presented as an alternative capable of revolutionizing the structure of corporations, businesses and even the design of new products. As this promising market "conquers" organizations and users, there is a strong competition and services are differentiated by its quality. In this competitive scenario, the level of the users' satisfaction about the cloud service emerges as a decisive factor for the choice of different solutions provided by different CSP (Cloud Services Providers). In this context, the quality of the users' experience (QoE) has become a differentiating factor among providers.

The methodology proposed in this paper is able to capture network and implementation aspects through the use of end-to-end response time, in order to estimate the quality of experience perceived by the users. At the same time, the methodology is able to adjust the MOS score according to the evaluation context, taking into account the variables presented in this setting.

Results had showed great variability in MOS score during observed period (one week), both within the evaluation context, as in the evaluated cases. Usual network parameters are unable to capture all variations of QoE, as shown in the discussion of Case 4. In this context, the evaluation period plays a critical role in the comparison of solutions. In this way, the monitoring of QoE in cloud services must be permanent since there are nondeterministic factors which might affect the shared infrastructure (eg momentary spikes caused by other clients for atypical times of use, elasticity limitations provider caused by mismanagement of resources by the provider, etc.).

From the results observed the importance of estimating the QoE through a methodology capable of capturing performance variations, not only the network, but also related to the client application performance and the performance of CSPs.

The case studies presented in this paper have demonstrated that the geographic location of the Cloud Services Providers seems to be really relevant; among the evaluated context variables, the geographical location of the server has shown to be the greatest impact on the estimated QoE. The results point out that the average difference between the best case in MOS (Context 4 located in Brazil - MOS = 2.3242) and the worst case (CSP located in Europe - MOS = 1.2486) was 1.0756.

Referring to the application, Flash ActionScript has shown better results if compared to JavaScript. Furthermore, the use of static pages on the server showed a better response when compared to the dynamic pages.

ACKNOWLEDGMENT

This work was partially founded by Hewlett-Packard Brasil Ltda. using incentives of Brazilian Informatics Law (Law nº8.2.48 of 1991).

REFERENCES

[1] F. R. Sousa, L. O. Moreira, and J. C. Machado, "Computação em nuvem: Conceitos, tecnologias, aplicações e desafios," *II Escola Regional de Computação Ceará, Maranhão e Piauí (ERCEMAPI).* pp. 150-175, 2009.

[2] Y. Yao and H. Xiaojun, "Evaluating the performance of cloud services in a browser-based network measurement platform," in *Networks (ICON), 2013 19th IEEE International Conference on*, 2013, pp. 1-6.

[3] Y. A. Wang, H. Cheng, L. Jin, and K. W. Ross, "Estimating the performance of hypothetical cloud service deployments: A measurement-based approach," in *INFOCOM, 2011 Proceedings IEEE*, 2011, pp. 2372-2380.

[4] T. Hobfeld, R. Schatz, M. Varela, and C. Timmerer, "Challenges of QoE management for cloud applications," *Communications Magazine, IEEE,* vol. 50, pp. 28-36, 2012.

[5] A. Khan, L. Sun, E. Jammeh, and E. Ifeachor, "Quality of experience-driven adaptation scheme for video applications over wireless networks," *Communications, IET,* vol. 4, pp. 1337-1347, 2010.

[6] D. F. T. W. WT-126., "Video services quality of experience (QoE) requirements and mechanisms," ed, 2007.

[7] S. Egger, P. Reichl, T. Hosfeld, and R. Schatz, "Time is bandwidth? Narrowing the gap between subjective time perception and Quality of Experience," in *Communications (ICC), 2012 IEEE International Conference on*, 2012, pp. 1325-1330.

[8] P. Reichl, S. Egger, R. Schatz, and A. D'Alconzo, "The Logarithmic Nature of QoE and the Role of the Weber-Fechner Law in QoE Assessment," in *Communications (ICC), 2010 IEEE International Conference on*, 2010, pp. 1-5.

[9] K. Hyun Jong and C. Seong Gon, "A study on a QoS/QoE correlation model for QoE evaluation on IPTV service," in *Advanced Communication Technology (ICACT), 2010 The 12th International Conference on*, 2010, pp. 1377-1382.

[10] L.-y. Liu, W.-a. Zhou, and J.-d. Song, "The Research of Quality of Experience Evaluation Method in Pervasive Computing Environment," in *Pervasive Computing and Applications, 2006 1st International Symposium on*, 2006, pp. 178-182.

[11] M. Jarschel, D. Schlosser, S. Scheuring, and T. Hossfeld, "An Evaluation of QoE in Cloud Gaming Based on Subjective Tests," in *Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2011 Fifth International Conference on*, 2011, pp. 330-335.

[12] P. Casas, H. R. Fischer, S. Suette, and R. Schatz, "A first look at quality of experience in Personal Cloud Storage services," in *Communications Workshops (ICC), 2013 IEEE International Conference on*, 2013, pp. 733-737.

[13] P. Casas, M. Seufert, S. Egger, and R. Schatz, "Quality of experience in remote virtual desktop services," in *Integrated Network Management (IM 2013), 2013 IFIP/IEEE International Symposium on*, 2013, pp. 1352-1357.

[14] M. Fiedler, T. Hossfeld, and P. Tran-Gia, "A generic quantitative relationship between quality of experience and quality of service," *Network, IEEE,* vol. 24, pp. 36-41, 2010.

[15] S. Egger, T. Hossfeld, R. Schatz, and M. Fiedler, "Waiting times in quality of experience for web based services," in *Quality of Multimedia Experience (QoMEX), 2012 Fourth International Workshop on*, 2012, pp. 86-96.

[16] X. Jie, X. Liyuan, A. Perkis, and J. Yuming, "On the Properties of Mean Opinion Scores for Quality of Experience Management," in *Multimedia (ISM), 2011 IEEE International Symposium on*, 2011, pp. 500-505.

[17] ITU-T, "P.800: Methods for subjective determination of transmission quality," ed, 1996.

[18] E. Ibarrola, F. Liberal, I. Taboada, and R. Ortega, "Web QoE Evaluation in Multi-agent Networks: Validation of ITU-T G.1030," in *Autonomic and Autonomous Systems, 2009. ICAS '09. Fifth International Conference on*, 2009, pp. 289-294.

[19] ITU-T, "Rec. G.1030: Estimating End-to-End Performance in IP Networks for Data Applications," ed, 2014.

[20] T. Hosfeld, S. Biedermann, R. Schatz, A. Platzer, S. Egger, and M. Fiedler, "The memory effect and its implications on Web QoE modeling," in *Teletraffic Congress (ITC), 2011 23rd International*, 2011, pp. 103-110.