EVALUATING SOFTWARE ARCHITECTURE SOLUTIONS IN THE CLOUD ERA

Irit Hadar[1], Mor Brokman[1] and Ethan Hadar[2]
[1]*Department of Information System, University of Haifa, Haifa, Israel*
[2]*CA Technologies, Inc., Herzelia, Israel*

Keywords: Software Architecture, Cloud Solutions, Private Cloud, Hybrid Cloud, Public Cloud, Survey.

Abstract: The quality of software architecture solutions is difficult to evaluate, making selection between architecture alternatives cumbersome. The cloud offerings add alternatives, by increasing complexity and dynamic flexibility of underlying deployment considerations. This research aims to identify the architects' main quality concerns regarding the system in three different environments: physical (static) deployment, virtual (private) cloud and public cloud. In this position paper, we present our evaluation of the importance architects attribute to applications characteristics (Functionality, Availability, Usability) and infrastructure and platform characteristics (Efficiency, Serviceability, Portability) in each of the above deployment environments. A survey, in which 34 experienced software architects participated, reveals the differences between quality concerns of architects and end-users, and between the different environments and system layers. Architects consider Portability (dynamic nature of applications and systems) as more important in private and public clouds than in static environments, while in the other quality attributes no significant difference was observed. In addition, architects perceive that end-users do not distinguish between cloud and non-cloud environments, and significantly consider that applications characteristics are more important than infrastructure ones.

1 INTRODUCTION

Over the years, the complexity and scale of software systems have grown significantly in enterprise IT environments. Economic and other forms of cloud volatility require multiple solution approaches (Howard et al., 2011). Managing multiplicity requires versatile design; overcoming and managing software complexity and versatility calls for increased effort in software architecture. Many of the difficulties in architecture decision-making stem from the fact that architecture solutions are difficult to evaluate and predict, in particular, the future architecture quality of the designed system. Many existing evaluation approaches were found to be unsuitable for the evaluation of the quality attributes (e.g., performance) of the software system (Eskenazi et al., 2002). Some researches try to predict the performance of software architecture solutions without benchmarking them (e.g., Eskenazi et al., 2002; Cortellessa et al., 2007).

The cloud era has added yet more considerations and dilemmas, as well as opportunities, to the traditional work and decision-making of the architect. The tasks of evaluating a software architecture solution or selecting the best of several architecture alternatives become increasingly complex. As more possibilities for designing software system architecture are available, the architect needs to compare and examine different values of the same software solution offered, by different cloud vendors.

Various types of cloud computing exist and should be taken into consideration by the IT architect. Each type refers and is relevant to a different layer of the system: Application, Platform, or Infrastructure. One of the goals of this research was to identify the relative importance of different quality characteristics in each of the system layers. In addition, while focusing on software architects, we used their perspectives to understand end-users' perspectives, in addition to their own.

The motivation for using cloud services is influenced by the offered value and quality, such as scalability and availability characteristics, and accurate billing according to consumption. In order

Hadar I., Brokman M. and Hadar E..
EVALUATING SOFTWARE ARCHITECTURE SOLUTIONS IN THE CLOUD ERA.
DOI: 10.5220/0003912802750278
In *Proceedings of the 2nd International Conference on Cloud Computing and Services Science* (CLOSER-2012), pages 275-278
ISBN: 978-989-8565-05-1

to be able to choose between different cloud vendors offering similar services, on alternative environments (physical, virtual, or cloud), there is a need to understand the characteristics' importance, as architects perceive them.

This article seeks to capture the importance of architecture quality characteristics deployed on different cloud environments relative to regular static ones, from the perspective of both software architects and end-users.

The rest of the paper is organized as follows: Section 2 describes our research method and Section 3 presents the results obtained. We discuss the results and conclude in Section 4.

2 RESEARCH METHOD

In order to investigate architects' perceptions and attitudes toward different types of computing services, via either public cloud, virtual private cloud, or physical deployment, we conducted an exploratory study with experienced practitioners (software architects) at a large international IT firm. By identifying and understanding the software architects' perspective, we aimed to identify best practices for evaluating software architecture solutions with and without the cloud, and illuminate future research directions.

2.1 Settings

A total of 34 practitioners participated in the study, including 24 R&D architects, 4 field architects and 6 senior software developers, with an average experience of 7 years, all from the same large, international IT firm. Following a short pilot study, which included several observations of architecture meetings and follow-up interviews, we prepared a questionnaire for our survey. The survey was first evaluated and validated by two experts from the investigated firm and refined accordingly prior its distribution. The survey focused on the importance of each characteristic in non-cloud environments and in two types of cloud computing: private and public.

2.2 Data Gathering and Analysis

The questionnaires were electronically distributed and received via email. The data were statistically analyzed for each question using the SPSS software (Statistical Package for the Social Sciences; see http://www-01.ibm.com/software/analytics/spss/). We created Paired Samples Correlations and Paired Samples Tests (Paired t-tests; see http://www.ats.ucla.edu/stat/mult_pkg/whatstat/choosestat.html).

Missing values were treated as missing. Statistics for each analysis were based on the cases with no missing or out-of range data for any variable in the analysis.

The results were considered significant at the 0.05 level, under the assumptions that the paired differences are dependent and identically normally distributed (or have a number of practitioners which is higher than 30).

3 RESULTS

3.1 Non-functional Requirements

Rather than focusing on the commonly handled non-functional requirements (NFR) of Data Recovery, Resource Redundancy and Scalability, we aimed at identifying the next NFR software that architects believe should be handled by the cloud. According to the survey, 52.9% of the architects believe that the next NFR to become the cloud vendors' responsibility should be Security. Moreover, many of the architects who did not rate security as the first NFR to move to the cloud rated it as the second or third. In order to calculate and scale all characteristics mentioned in all ratings, first, second and third, we made a calculated average as follows. The characteristics mentioned as the first NFR got three points; the characteristics mentioned as the second NFR got two points; and the characteristics mentioned as the third NFR got one point. According to this calculation, the highest rated NFR was Security, with 67 points, followed by (with a big difference) Performance with 12 points, and Availability with 10 points.

3.2 The Different Layers in the Different Environments

A proposed or existing software system is usually evaluated using long lists of requirements, expressed as attributes, and measures for indicating how well they are satisfied by the system. The system must exhibit several system-wide properties, commonly referred to as "the ities", such as reliability, maintainability, usability, and portability.

Our first goal here was to find whether the perceived relative importance of each characteristic in each part of the system (Application / Platform /

Infrastructure) depends on the type of the system. Accordingly, we asked the architects to consider two types of potential application: MS-Project application and one for CRM-SAP system, with the percentage of importance (0-100%) of each characteristic in each part of the system (Application / Platform / Infrastructure). For analyzing the results, we created a paired t-test as follow: Given two paired sets: X_i for MS-Project and Y_i for CRM-SAP of $n = 34$ measured values.

From these matched pairs, we compared the two measurements by subtracting one from the other and basing test hypotheses upon the differences. Our null hypothesis H0 assumes that the mean of these differences is equal to 0, while the alternative hypothesis H1 claims that the mean of the differences is not equal to zero (the alternative hypothesis may be either one- or two-sided). Our paired t-test determined there is no significant difference between the two applications, thus confirming H0. This illustrates that the relative importance of each characteristic in each part of the system (Application / Platform / Infrastructure) does not depend on the type of the system. This conclusion strengthens the approach to developing a general framework for evaluating a software architecture solution.

Our next test aimed to identify what difference, if any, exists in the relative importance of each characteristic between the different layers of the system (Application / Platform / Infrastructure).

Our analysis revealed that the relative importance of both Functionality and Usability are significantly greater in the Application layer of the system than in the other layers. In all other NFRs, no significant difference between the layers of the system was identified.

3.3 The Different Perspectives

The software architect, who is responsible for designing the system, and the end-user, who is intended to use the system, each has specific views and considerations. We examined whether the concern level (0-10) for each characteristic in the different deployment environments differs when examining these two points of view, as the architects perceive them. Here, too, we used the sensitivity to application type of MS-Project and CRM-SAP, with regard to the results.

The architects filled four tables: for each application, a table for the architects' perspective and another for the end-users' perspective. Concern levels (0-10) for each characteristic in the three different deployment environments were defined: from 0 = Irrelevant or no concern to 10 = the highest level of concern. We created paired t-test for the two systems, and here, too, we found no significant difference (based on significance level of 0.05).

4 RESULTS ANALYSIS

Analysis of the data revealed several insights with regard to differences between the concern levels of architects and end-users. Figure 1 highlights higher level of concern of Portability (dynamic changes) in private and public cloud environments, than in static one. In general, architects exhibit higher concerns for all other characteristics as well in both private and public environments based applications than static ones, yet, not significantly different.

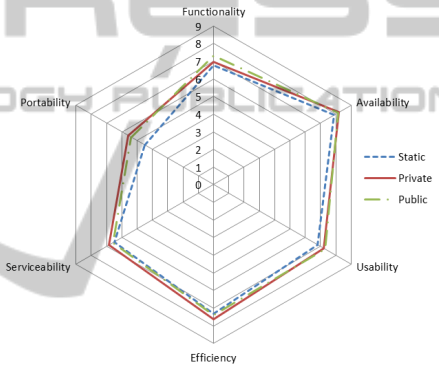

Figure 1: Preferences of architects in static, private and public deployment.

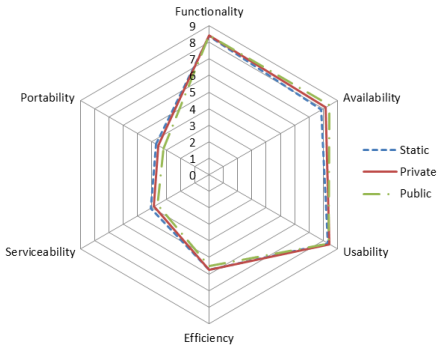

Figure 2: Preferences of end-users as perceived by architects in static, private and public deployment.

Architects perceive that end-users are insensitive to the application's environment; most of the concern levels are similar in each characteristic (see Figure 2). According to the architects, end-users consider that applications characteristics (Usability, Availability, Functionality) are more important than

the underlying infrastructure ones (Efficiency, Serviceability, Portability). This confirms the lack of end-users' sensitivity between the different platforms. A possible conclusion is that if end-users are not concerned with artifacts that are driven from the underlying infrastructure, they will not be concerned with the nature of the infrastructure (static/physical, private or public).

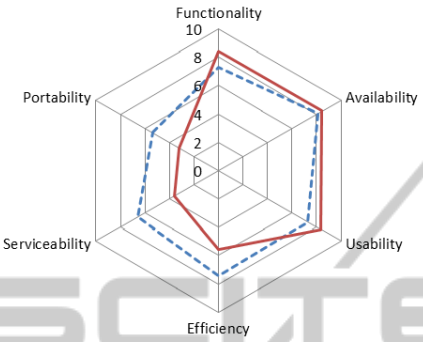

Figure 3: Preferences of public cloud deployment (architect line is dashed, end-users line is rigid).

Figure 3 shows there is no match between architects' levels of concerns, and their perception on end-users' concerns. This phenomenon is similar in all environments. Architects consider their responsibility to handle additional quality characteristics that are not usually visible by end-users.

5 DISCUSSION

The results presented in this paper are part of our ongoing research with the long-term objective of establishing a framework for the evaluation of architecture solutions in the cloud era.

In this paper we evaluated the importance architects attribute to applications characteristics (Functionality, Availability, Usability) and infrastructure and platform characteristics (Efficiency, Serviceability, Portability) in physical (static), private (virtual) and public clouds. Our survey results of 34 experienced software architects showed that Portability (dynamic nature of applications and systems) was considered more important in private and public clouds than in static environments. In addition, architects perceive that end-users do not distinguish between cloud and non-cloud environments, and consider applications characteristics to be more important that infrastructure ones.

Our future work will validate the results with end-users as well as include software architects from several different firms. An evaluation framework taking these results into consideration would be effective for evaluating an overall system solution in the cloud era.

REFERENCES

Cortellessa V., and Frittella. L., 2007. A framework for automated generation of architectural feedback from software performance analysis. In K. Wolter, editor, *Proc. of Fourth European Performance Engineering Workshop*, volume 4748 of Lecture Notes in Computer Science, pp. 171-185. Springer.

Eskenazi, E. M., Fioukov, A.V., Hammer, D.K., and Obbink, H., 2002. Performance prediction for software architectures. In: *Proceedings of PROGRESS 2002 workshop*, Netherlands.

Howard C., Nelson E., Rollings M., and Santos J., 2011. 2012 IT Professionals Planning Guide: Volatility, Multiplicity, Versatility, and Mobility. *Burton IT1 Research*, G00226064, 1 November 2011.