ELEKTROTEHNIŠKI VESTNIK 78(3): 123-127, 2011
ENGLISH EDITION

Software Engineering in the Cloud for Reducing the Application Time-to-market

Aljaž Zrnec

University of Ljubljana, Faculty of Computer and Information Science, Trzaska 25, 1000 Ljubljana, Slovenia
E-mail: aljaz.zrnec@fri.uni-lj.si

Abstract. In this paper we present the possibilities offered by cloud computing in development and testing of software. They allow us to shorten the time required for developing software and to shorten the time of software to become available on the market. We mainly focus on provisioning cloud resources on demand, accounting for the actual consumption of resources and adjusting the scale of the computing resources to contribute to more effective implementation of development and testing activities and ensure a higher level of quality of designed solutions.

Keywords: cloud computing, cloud, software, development process, cost

1 INTRODUCTION

Nowadays, in most business systems software is a key enabler of their business processes. Besides, the software availability and stability directly impact the company revenue and customer satisfaction. Software development is one of the most critical processes ensuring successful business in majority of business systems as it requires a lot of effort in ensuring adequate financial resources and software development resources (hardware and software).

Recently, software development is undergoing major changes. A growing number of independent software vendors transform themselves into service providers delivering their applications in the form of services hosted in the cloud [1, 2, 3]. They have become the so called Software-as-a-Service vendors (SaaS). Applications for financial management, taxes and investments represent a typical example of software that has been until recently available only for servers and desktops, but now it's available to thousands of simultaneous users in the form of secure web services in the cloud. Business success of cloud vendors depends on reducing the time needed for the development of the aforementioned services and promoting their entry into the market respectively (time to market). Each additional month or quarter in which the cloud services are accessible to users has a direct impact on increasing revenues, which affects the final financial statements. The SaaS delivery model itself aids business systems in accelerating their time to market, as the vehicle for application delivery is directly over the Internet, rather than the traditionally packaged distribution or downloadable modules.

Timely production of software has a major impact on business. In manufacturing companies, software development includes the design or simulation of a product to be sent to the market. In media and entertainment companies, creating a new brand or specific web site can accelerate the end user traffic, thereby driving advertising which directly impacts the revenue. In the financial-services industry, the need to provide new online capabilities for trading, portfolio and wealth management impacts the customer attraction and retention.

From the above we can clearly see that the speed at which software applications can be developed, tested and brought into production is one of the critical success factors for many companies. Therefore, any solution accelerating the application time-to-market has an immediate and measurable impact on return on investment (ROI).

The paper is organized as follows. In Section 2 we present problems occuring in the context of conventional software development in terms of preparing an appropriate development environment and issues related to performing software testing. In Section 3 we present advantages of using cloud computing for software development and factors accelerating the application time-to-market. In Section 4 we present ROI of using cloud computing for software development. The paper ends with a concluding discussion in Section 5 we draw conclusions and give proposals for our future work.

2 PROBLEMS IN SOFTWARE DEVELOPMENT

To configure hardware and software, software developers demand a highly dynamic environment. In practice, they all too often encounter resistance of system administrators [4] striving to establish more rigid environments and thereby providing greater stability in operation of the computer infrastructure in business systems. This is why for many years different

Received 28 December 2010
Accepted 18 February 2011

demands of developers and administrators have been conflicting and have as such considerably reduced productivity of software development.

2.1 Development environment

Application developers are regularly confronted with a request to establish special environments for developing, debugging and compiling appropriate software libraries and tools for making the software code. Typically, these environments are established for a limited period of time, usually several hours. Then they become obsolete and can be removed. Accessing appropriately configured development environments with an adequate processing power and storage space on demand within a computer center in the company, is crucial for creation of such development environments and for ensuring productivity of software development. To perform their tasks, the programmers should be able to quickly configure servers, storage and network connections. Unfortunately, in many development environments it is particularly the speed of relevant resource preparation that inhibits or even prevents execution of the programmer tasks as assurance of computer resources in the business system is in the domain of the IT department (administrators) or, resources allocated to the development are limited to a certain level. These conditions are forcing developers to carry out development activities in the traditional - waterfall manner thus preventing parallel execution of activities and slowing down the development process. The inability of developers to rapidly provision, de-provision or copy (clone) properly configured development environments are therefore serious inhibitors to productivity and hence time-to-market.

2.2 Testing

The software development process also includes testing activities. To be successfully completed, these activities require more resources than there are available for development within computer facilities in the business system. The test environment is very dynamic by its nature. It is created to run a series of test cases and to review and analyze the results, but then becomes dormant until the next iteration. The described procedure of testing shows the need for more agile IT capability, where test environments can be provisioned on demand. It would be ideally if the role responsible for testing itself created a proper test environment.

2.3 Quality assurance

2.3.1 Performance testing

The Quality Assurance (QA) process has its own requirements that affect execution of tests and timely preparation of a sufficient amount of computing resources. On top of running functional test cases and regression sequences, the QA process typically bears the additional burden of performance and scalability testing making the QA environment unique contrasted to the developer's iterative code/test cycle.

To implement performance testing, the QA environment needs to provision computer-center resources that perform at production levels. This includes:

- appropriate number of servers,
- appropriate number of processors in servers,
- adequate amount of the RAM memory,
- fast communication links to provide an adequate quality of service (QoS), and
- appropriate storage systems with the types of throughputs and latencies that will be seen in production.

The objective of establishing the QA environment for performance testing is to optionally reflect the actual situation of the production environment. This is often very difficult to attain within a computer center in a business system, because the setting up an environment to reflect the actual situation of production, even on a limited scale, is a highly complex task. Even if a business system has adequate computer resources to create such an environment, its creation solely for the purpose of performance testing is very questionable in terms of cost and time. Therefore, the activity of performance testing defined within the QA process is often postponed to the production phase.

2.3.2 Scalability testing

With the advent of software services, scalability testing became one of the most important activities within the testing process. With delivery of applications in a hosted or SaaS model, the number of concurrent users accessing the service can very quickly rise to a very high number, which can lead to performance degradation of the vendor's network connections, servers and storage. Software that performs perfectly well for ten users can encounter bottlenecks or even bugs in its code when exposed to 100 or more concurrent users. The need for QA process to have adequate computer-center resources at its disposal on demand can be viewed as a major accelerator for time-to-market provided that scalability and performance of the new software have been thoroughly tested.

3 SOFTWARE DEVELOPMENT IN THE CLOUD

Today, IT environments in which a development process takes place should be more agile to meet the highly dynamic and resource-intensive needs of the application development environment. A new generation of cloud services can address these

requirements for more agile IT by providing the following important capabilities:

- on-demand provisioning of resources in less than an hour through a self-service access,
- multiple-service levels (Quality of Service - QoS) to address the wide range of requirements in the application lifecycle,
- rapid scale-up and scale-down of resources with usage-based billing,
- support for virtual development environments and multi-tier application architectures, and
- migration of the existing virtual-server images and workloads into the cloud.

Developers can leverage these capabilities on their own, with no intervention of IT administrators in the business system, which dramatically speeds time-to-market for new applications.

3.1 On demand resource provisioning

As mentioned in Section 2, ensuring appropriate resources for developing applications within the computer center in a business system can be a very complex and expensive task. In doing so, administrators must work together, to make appropriate development environments. This involves collaboration with IT administrators, who prepare proper development environments on request of developers. Preparation of development environments usually takes a lot of time and effort in the mutual communication between developers and administrators, as the latter are reluctant to interfere with the resources of the computer center which they are directly responsible for.

By using the cloud services, it is possible to establish appropriate computing resources (development environment) in the cloud. Computing resources are prepared on demand and are used in different software development activities of the software process. Using these resources can accelerate the development cycle by creating multiple development environments. Multiple development environments enable several software process activities to be carried out simultaneously. For example, using several development environments can be established to run multiple tests concurrently with the development activity. The idea of preparing the necessary resources in the cloud is presented in Fig. 1:

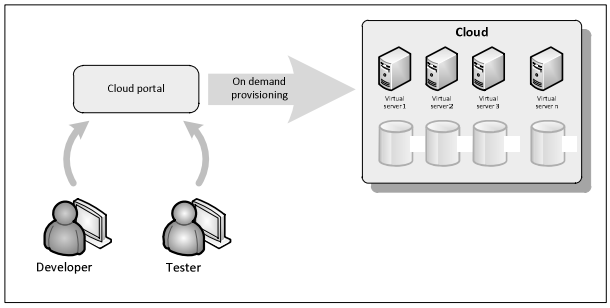

Figure 1. On-demand resource provisioning in the cloud

Developers who use the cloud services can prepare development environments tailored to particular needs of the development-process activities. This makes it possible to release some of the key resources of the computer center that could have posed a bottleneck in the context of the development-process activities. In this way, developers can increase productivity. Moreover, administrators no longer need to fear that changing the configuration of the resources in a computer center of a business system may jeopardize operation of the business information system.

3.2 Using different levels of QoS in the cloud

The use of the cloud services is linked to multiple Quality of Service levels (QoS - Quality of Service). The higher the QoS level required, the better its performance and the higher the price of its application.

Developers can use different levels of QoS, so that for each developmental stage in the application life cycle they can optimize the relationship between service performance and its price. For example, the ability for developers to create an environment in a cloud that is optimized for the needs of development and functional testing would entail lower levels of performance guarantees, but at the return of a lower hourly cost. As the application lifecycle moves to QA (see Section 2.3), the development environment needs to more closely reflect the production environment. This means that a higher level of performance, security and availability needs to be assigned to our development environment for performance and scalability testing. In exchange, the hourly cost of such environment gets higher.

The QA process will also benefit from on-demand up and down scaling of cloud resources, as this finally solves the problem of testing performance and scalability of applications at a large scale, but without indefinitely reserving and paying for resources when they are unused. With usage-based billing, the cloud will bring closure to the need to pay for unused resources and their prior booking in traditional computer centers, enabling the customers to pay for resources used, as opposed to resources deployed.

3.3 The use of virtual environments and support for multi-tier architectures

Services in the cloud support the establishment of different virtual environments. The most classic use of the cloud includes the establishment of an environment consisting of several virtual servers. These virtual servers are used to perform development, functional testing and quality testing (see Section 2.3). Today, a selected set of cloud services is also beginning to support modern, multi-tier application architectures [5]. Typically, these are 3-tier architectures consisting of a presentation tier, logic or application tier and data tier. In this way development of traditional web applications is directly supported by providing a proper number of virtual servers used for development in each aforementioned tier.

When the development in a virtual cloud environment is finished, the images of virtual servers can be easily transferred to the production environment. The production environment can be established in the cloud, so the images only have to be moved to another virtual environment intended for the production purpose. Another possibility is to install the images of virtual servers to local servers in a computer center in a business system. The advantage of the two cases is the possibility to avoid problems related to configuring a new application for transfer from the development to the production environment, which again affects the speed of the application time-to-market.

Services in the cloud also allow to transfer virtual server images in the opposite direction, namely from the computer-center servers or a private cloud to the cloud of a particular service provider. In this case, the initial application-development activities are performed using the own computing infrastructure. When it comes to performance and scalability testing at a large number of users, the development environment is moved to the cloud, because computer centers in business systems don't usually have adequate resources to perform such tests.

4 SOFTWARE DEVELOPMENT IN THE CLOUD: RETURN ON INVESTMENT

Many authors have proposed several different models [6] to calculate ROI of using cloud computing. The proposed models take into account several economic and technological indicators which make the ROI calculation quite complex.

A simplified model for the ROI calculation of a cloud usage to develop software (and thereby reducing the time needed for application to enter into the market) takes into account the different economic and technological advantages of using services in the cloud. These services impact the cost of the development process. Their technological advantages are as follows:

- recently, the progress of cloud computing contributed to emergence of services enabling to set up a complete virtual development environment in the cloud,
- developers can set up a development environment on demand very quickly,
- resources used within a virtual development environment in the cloud can dynamically adapt to the needs of a development process,
- virtual environments support the use of multi-tier architectures directly, and
- possibility of migrating virtual development environments between the cloud and own computing infrastructure or between different clouds.

Economic advantages are as follows:

- costs for the cloud-service usage are charged only for the resources and the time actually used,
- the use of the cloud services doesn't imply any advance costs, and
- the use of a cloud services doesn't require any long term commitment to a particular cloud vendor.

The technological advantages of using the cloud enable us to implement a parallel execution of development activities and assure quality of the developed applications, thus speeding up the entire development process at lower cost. The economic advantages of using services in the cloud considerably reduce the cost of the development process. Speeding up the execution and reducing the costs of the development process increase the speed of the application time-to-market and its fast ROI. Fig. 2 shows the relation between the development activities transferred to the cloud, duration of application development using these activities and ROI of a software development in the cloud.

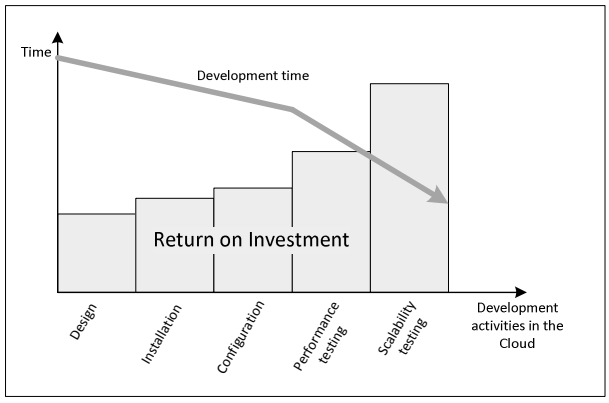

Figure 2. Return on Investment

We can see that ROI increases with the number of development activities carried out in the cloud. It is particularly affected by performance and scalability testing under the high load. These two activities can be carried out quickly and at a low cost if executed in the

cloud. By accelerating the development process, they make the total time for application development to decrease and the investment to become profitable more quickly.

5 CONCLUSION

Using cloud services in development of software in terms of speed of development and profitability of investment is meaningful. Of course, this doesn't mean that a business system which already has a computer center with adequate resources has to immediately move all of its software development activities into the cloud. First, a business system should examine its capabilities and identify which activities cannot be supported by its own IT infrastructure. Then, on the basis of identified activities, it should decide whether a particular activity will be carried out in the cloud and what will be the impact of using the cloud services on the overall pace of the development process. It is necessary to consider several criteria for implementation of development activities in the cloud, such as the cost of services, security and performance [7]. However, we certainly find it reasonable to carry out performance and scalability testing under the heavy load in the cloud, since business systems usually lack the resources enabling them to carry out these activities to the extent required.

Our further work will therefore be focused on adapting the software development process by using the cloud services. We will define a model of a procedure based on several criteria enabling us to answer the question whether a particular development activity should be carried out locally in a business system or by using the cloud. Our model will be based on the theory of process adaptation to the project-specific needs [8, 9], that defines several kinds of decision rules for process adaptation.

REFERENCES

[1] B. HAYES, Cloud Computing, *Communications of the ACM*, Vol. 51, No. 7, pp. 9-11, 2008

[2] A. REED, S. G. BENNET, Silver Clouds, Dark Linings: A Concise Guide to Cloud Computing, *Prentice hall*, ISBN-13: 978-0-131-38869-7, 2010

[3] N. STINCHCOMBE, Cloud computing in the spotlight, Vol. 6, No. 6, pp. 30-33, 2009

[4] S. VRHOVEC, R. RUPNIK, A model for resistance management in IT projects and programs, *Electrotechnical review*, Article in reviews

[5] G. J. POPEK, R. P. GOLDBERG, Formal requirements for virtualizable third generation architectures, *Communications of ACM*, Vol. 17, No. 7, pp. 412–421, 1974

[6] M. SKILTON, Building Return on Investment from Cloud Computing, The Open Group, http://www.opengroup.org/cloud/whitepapers/ccroi/index.htm, 2010

[7] D. SVANTESSON, R. CLARKE, Privacy and consumer risks in cloud computing, *Computer Law & Security Review*, Vol. 26, No. 4, pp. 391-397, 2010

[8] A. ZRNEC, D. VAVPOTIČ, An approach to create a project-specific software development process. *Electrotechnical review*, Vol. 75, No. 1/2, pp. 44-49, 2008

[9] A. ZRNEC, M. BAJEC, M. KRISPER, An approach to adapting the IS software process. *Uporabna informatika*, Vol. 14, No. 1, pp. 75-80, 2006

Aljaz Zrnec received his B.Sc., M.Sc. and Ph.D. degrees from the Faculty of Computer and Information Science, University of Ljubljana in 1999, 2002 and 2006 respectively. He works in the Laboratory for data technologies as a lecturer and assistant in the field of databases. His research interest is in database management systems and cloud computing. He is the author or coauthor of numerous professional and scientific papers.