Chapter 7
Requirements Engineering for Cloud Application Development

Holger Schrödl
University of Magdeburg, Germany

Stefan Wind
University of Augsburg, Germany

ABSTRACT

In industrial practice, cloud computing is becoming increasingly established as an option for formulating cost-efficient and needs-oriented information systems. Despite the increasing acceptance of cloud computing within the industry, many fundamental questions remain unanswered, or are answered only partially. Besides issues relating to the best architectures, legal issues, and pricing models, suppliers of cloud-based solutions are faced with the issue of appropriate requirements engineering. This means eliciting optimal understanding of the customer’s requirements and implementing this into appropriate requirements of the solution to be realised. This chapter examines selected, established requirements engineering methods in order to study the extent to which they can be applied to the specific requirements of cloud-based solutions. Furthermore, it develops a comparison framework containing the features of cloud computing. This comparison framework is applied to four established process models for requirements engineering. Recommendations for a requirements engineering process adapted to cloud computing are derived.

INTRODUCTION

While cloud computing has already found its way into practice, deficits in the scientific basis (Leimeister, Böhm, Riedl, & Krcmar) remain. One such shortfall is requirements engineering for cloud computing - as a separate unit with its own different domains. Before elaborating on the basics of requirements engineering, we take a short insight what encompasses cloud computing and cloud-based solutions.

For the term “cloud computing”, a unique definition is not yet established (Armbrust et al., 2010). But the essence of every definition is the intention to provide IT services over the internet from a provider without taking care of the essential
resources which are needed to provide this particular service. To distinguish this approach to other service concepts like ASP (Application Service Providing) we draw attention to the dimension of agility. While the ASP concepts focusses on mid- and long-term relationships between provider and user, cloud computing supports on-demand relationships with more agility than a traditional application service providing (R. Buyya, Ch S. Yeo, & S. Venugopal). Therefore, cloud computing is a highly modular service providing concepts. In a classical definition of cloud computing, we consider three levels of services: Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS). Cloud-based solutions are software applications, which are combined and/or enhanced with modules from these three levels of service providing, integrated to one single solution (Grossman, 2009).

While some preliminary research initiatives have been carried out under the sub-domain of Software as a Service (SaaS) (c.f. Berkovich, Esch, Leimeister, & Krcmar, 2009, Berkovich, Esch, Leimeister, & Krcmar, 2010), none have yet been carried out for cloud computing in general. Because of its specific characteristics and the different requirements fields, it is necessary to make a distinction between these and established requirements. Forrester Research Consultants has investigated eleven different cloud computing vendor offers with respect to fields of application, costs and commercial benefits, and has drawn a sobering conclusion: many offers do not meet - or only partially meet - customers’ requirements (Staten, 2009). The success of cloud computing, therefore, depends a vast understanding on how well customers’ and other stakeholders’ requirements and wishes are met. The basis for developing successful offers is a requirements engineering system adapted to cloud computing.

The success of development processes and projects essentially depends on whether the results meet the requirements of stakeholders (such as the customer, executive management, legislators etc.). A crucial element here is the implementation of an appropriate and professional requirements engineering tool (Lindemann, 2009; Pohl, 2008; Rupp, 2009). Errors concerning the requirements are one of the main reasons why development projects fail (Aurum & Wohlin, 2010). Evidence of this is provided on a regular basis by the CHAOS study (See Figure 1) carried out by the American consultancy firm, the Standish Group. In a recent study, carried out in 2009 almost 48% of the problems or shortcomings in software development could be traced back to poor requirements engineering (Standish Group, 2010).

Moreover, studies carried out in a wide range of domains (product development, software engineering etc.) show, that errors made while determining requirements include a significant influence on the development process (Hall, Beecham, & Rainer, 2002). And work and costs involved in eliminating the errors increase disproportionately to the time at which they appear (Berkovich, Leimeister, & Krcmar, 2010). The reason for this is the early point in time within the process at which the requirements are defined. It means than any errors occurring at that early stage will affect all the future phases (such as design, implementation etc.) (Pohl, 2008). In his error pyramid, Lefringwell works on the basis that fixing an error at the implementation stage is up to 100 times more difficult, and at the maintenance stage, up to 1000 times more difficult than at the start of development stage (Dörnemann & Meyer, 2003).

Cloud computing is an issue in which, in general, enterprise IT managers are showing a vast deal of interest. According to the latest survey carried out by Sterling Commerce GmbH, a software supplier in Duesseldorf, 87% of all senior IT managers in Germany are planning to move to cloud-based information systems in the B2B sector (Sterling Commerce, 2010). The main driver of such considerations is the study on cost pressures: most companies want to reduce costs by