Available online at www.sciencedirect.com

ScienceDirect

International Journal of Project Management 33 (2015) 1262 – 1273

International Journal of Project Management

www.elsevier.com/locate/ijproman

Diagnosing organizational risks in software projects: Stakeholder resistance

Simon L.R. Vrhovec *, Tomaž Hovelja, Damjan Vavpotič, Marjan Krisper

University of Ljubljana, Faculty of Computer and Information Science, Ljubljana, Slovenia

Received 8 August 2014; received in revised form 28 January 2015; accepted 16 March 2015
Available online 29 March 2015

Abstract

Critical success and failure factors of software projects were extensively studied. However, software project risk management has rarely researched organizational risks even though most problems occur when the social aspects are not addressed. By employing the resistance to change theory, our paper develops an organizational risk diagnosing (ORD) framework in order to show how can organizational risks be better understood and managed. Organizational risk factors may have non-trivial underlying root causes. A failure to diagnose them may result in ineffective risk responses that address the symptoms. A case study of a loan application software project has been conducted in one of the biggest banks in South-Eastern Europe. An analysis of the risk management process in the studied case allows a better understanding of organizational risk management.

Keywords: Software project; Stakeholder resistance; Risk management; Bank

1. Introduction

Software project failure rates remain alarmingly high despite surging investments in information systems and their importance for contemporary organizations (Altuwaijri and Khorsheed, 2011; Baccarini et al., 2004; Bannerman, 2008; El Emam and Koru, 2008; Hong and Kim, 2002). According to the CHAOS Manifesto 2013 (The Standish Group, 2013), only 39 percent of software projects were successful, i.e., completed on-time and on-budget, with all features and functions as initially specified. Another 43 percent of projects were challenged, i.e., completed and operational but over-budget, over the time estimate, and offer fewer features and functions than originally specified. The remaining 18 percent of software projects have failed, i.e., they were cancelled prior to completion or delivered and never used. When considering only large software projects, only 10 percent were successful, 52 percent were challenged and 38 percent have failed. This is at least worrying as large software projects failure may negatively affect the whole implementing enterprise (Bernroider et al., 2014; Hong and Kim, 2002; Lavbič et al., 2010).

Software projects are high risk activities due to the rapid pace of technological changes and the organizational changes they may impose (Aloini et al., 2007; Altuwaijri and Khorsheed, 2011; Bannerman, 2008; Cule et al., 2000; Hong and Kim, 2002; Kwahk and Kim, 2007; Li et al., 2011) therefore risk management is essential for project success (Baccarini et al., 2004; Tiwana and Keil, 2004; Wallace et al., 2004). In the recent years, much has become known about why software projects fail (de Bakker et al., 2010). Several risk factors have been identified and joined into checklists and classification frameworks (Bannerman, 2008). Also, stepwise tasks for managing risks, also known as process models, are widespread in theory and practice (Aloini et al., 2012; Bannerman, 2008).

However, software project risk management seems to be rather immature as risks are still not managed effectively (Aloini et al., 2007; Bannerman, 2008; de Bakker et al., 2010; Geraldi et al., 2011; Kappelman et al., 2006; Kutsch and Hall,

* Corresponding author at: Večna pot 113, 1000 Ljubljana, Slovenia. Tel.: +386 1 479 8297.
E-mail address: simvrh@gmail.com (S.L.R. Vrhovec).
URL: http://simvrh.wordpress.com/.

http://dx.doi.org/10.1016/j.ijproman.2015.03.007
0263-7863/00/

2005; Osipova and Eriksson, 2013). In addition to technical risks, software projects are subjected to organizational risks since they affect or are affected by the way of doing things in an organization (Benaroch et al., 2006; Sanderson, 2012; Sharma and Gupta, 2012). These risks should not be overlooked as most problems occur when the social aspects are not addressed (Atkinson et al., 2006; Laumer, 2011). People are one of the greatest sources of uncertainty in any project undertaking therefore organizational risks are difficult to manage and knowledge of risks alone is not enough to contribute to project success (de Bakker et al., 2010; Thamhain, 2013). Nonetheless, organizational risk factors have been rarely researched (Aloini et al., 2007). Risk management research has only recently shown more interest in stakeholder-related processes and put an emphasis on "soft skills" as a complement to the "hard skills" (de Carvalho and Rabechini Junior, 2015; Söderlund and Maylor, 2012). Research shows that different stakeholder perspectives need to be considered in order to build the bigger picture and manage risks effectively (Hartono et al., 2014).

Among the most prominent and well-researched organizational risks in software projects is resistance to change (Hong and Kim, 2002; Jiang et al., 2000; Kim, 2011; Kwahk and Kim, 2007; Laumer, 2011; Lundy and Morin, 2013; Meissonier and Houzé, 2010; Rivard and Lapointe, 2012; Žvanut et al., 2011). It has been thoroughly researched for over half of a century in managerial psychology and information systems research (Laumer, 2011; Oreg et al., 2011; Rivard and Lapointe, 2012). Resistance to change is a complex phenomenon and several sources of resistance which can be considered as risk factors have been identified in the literature. The success of resistance management depends on the ability to diagnose resistance, i.e., to distinguish symptoms from root causes (Fiedler, 2010; Laumer, 2011; Lundy and Morin, 2013; Rivard and Lapointe, 2012; Zander, 1950).

The paper builds on the premise that organizational risks may not be managed effectively if one only focuses on project-specific risks or uses existing risk management approaches, such as checklists and classification frameworks. Furthermore, improving the existing approaches by considering different stakeholder perspectives may not be enough if the stakeholders themselves do not distinguish the symptoms from the root causes. The purpose of this paper is to move beyond the limitations of existing risk management approaches and advance the diagnosing of root causes of organizational risks from multiple stakeholder perspectives. By employing the resistance to change theory we develop a new theoretical model – the organizational risk diagnosing (ORD) framework. The ORD framework attempts to identify in a novel way the non-trivial underlying root causes which organizational risk factors may have.

This paper is structured as follows. First, extant work on software project risk management and resistance to change is summarized. The concept of stakeholder resistance is introduced. Next, the resistance checklist aggregating various works on resistance to change is developed. Afterwards, the proposed ORD framework is presented along with an application to the resistance context. A case study of a software project introducing new loan application software is presented and analyzed. The contributions of our research to risk management and resistance research are then discussed including the limitations of the study and further research opportunities.

2. Theoretical background

2.1. Software project risk management

The ISO 31000 Standard defines *risk* as the "effect of uncertainty on objectives" (ISO, 2009). Even though they are commonly viewed from the narrow perspective of the negative side of the possible effect (Hartono et al., 2014), risks can have both a positive or a negative effect on a project when present (Benaroch et al., 2006; ISO, 2009; Wallace and Keil, 2004). In software projects, much work has been done on discovering risks, also known as *risk factors* (Benaroch et al., 2006). Other terms, such as "sources of risk", "critical success factors", "uncertainty factors ", "risk drivers" or "risk items", can also be found in literature (Aloini et al., 2007; Bannerman, 2008; Tiwana and Keil, 2004). In some risk management fields, such as construction, risk factors are differentiated from risks. In contrast to established software project risk management where risk factors are usually directly related to their effects (Aloini et al., 2007; ISO, 2009), construction project risk factors do not affect the project directly but do so through risks (Tah and Carr, 2001). This distinction helps in risk evaluation because the risk factors are more concrete abstractions of a risk and define situations that can be individually assessed with a limited amount of vague information or facts (Tah and Carr, 2001).

Three main approaches to software project risk management can be found in theory and practice (Bannerman, 2008; de Bakker et al., 2010): checklists, classification frameworks, and process models. *Checklists* are formed by joining risk factors that have been identified in past projects (de Bakker et al., 2010). Various checklists can be found in literature (Aloini et al., 2007; Schmidt et al., 2001). Risk factors in checklists are commonly a mixture of technical and organizational risks ordered by general risk probability in software projects (Aloini et al., 2007). Checklists are often comprised of too many potential risk factors to effectively identify and manage them (Bannerman, 2008; Cule et al., 2000). To deal with this issue, some risk factors can be grouped and managed together. Using the construction project risk management terminology, risk factors are grouped into risks using some *classification framework* according to different criteria, e.g., their perceived source (Baccarini et al., 2004; Bannerman, 2008; Cule et al., 2000; Keil et al., 1998; Liu and Wang, 2014; Wallace et al., 2004). Since checklists and classification frameworks are closely related, they both have the same drawback. Risk factors are generic and based on past research which raises the prospect that risk assessment may be biased or limited in scope (Bannerman, 2008).

The third risk management approach is *process models*. Process models specify risk management activities which follow a general risk management process: establishing the context, risk identification, risk analysis, risk evaluation, risk treatment,