The social interaction of developers and IT operations staff in software development projects

Jon Iden [a],[*], Bendik Bygstad [a],[b]

[a] *NHH Norwegian School of Economics, Bergen, Helleveien 30, 5045 Bergen, Norway*
[b] *University of Oslo, Gaustadalléen 23, 0373 Oslo, Norway*

Abstract

This paper investigates how developers and IT operations staff interact in software development projects. We analyse data from 42 IT professionals from 18 Norwegian firms through the lens of social interaction and project management theory. Our analysis suggests that their social interactions are hampered by a variety of factors. The study contributes to the research by providing an analysis of the elements of social interaction and how they contribute to better outcomes. For practice, we offer an assessment instrument for improving the social interaction in software development projects.
© 2017 Elsevier Ltd. APM and IPMA. All rights reserved.

Keywords: Project management; Software development projects; Social interaction; Software developers; IT operations staff

1. Introduction

The development of software can be perceived as a social process involving several participants (Ghobadi and Mathiassen, 2016; Kautz et al., 2007; McLeod and Doolin, 2012), including developers, user and business representatives, vendors, staff from IT operations and external consultants. These participants must integrate their domain knowledge to achieve project success (Ghobadi and Mathiassen, 2016; Tesch et al., 2009). Clearly, the nature and quality of the interactions between participants influence the performance and outcome of the project (McLeod and MacDonell, 2011; Procaccino et al., 2006). For example, combination of knowledge is required to fully exploit the potential of project teams (Mueller, 2015) and has been found to have a significant impact on successful project outcomes (Park and Lee, 2014; Pee et al., 2010).

This article explores the interactions between two distinct participants—developers and IT operations staff—in software development projects. Developers are concerned with the activities needed to plan, design, and build the software that will ultimately be used by some part of the organisation to address a business requirement. Such activities include software acquisition through custom development, purchase, provisioning and any combination thereof. The main objective of IT operations staff is to ensure that the software, once built and deployed, can meet its defined service level. Software should be designed and built for operability, reliability, performance and manageability and tested for compliance. In Fig. 1, we use the traditional waterfall model (Sommerville, 2015) as an example of the activities conducted throughout the lifetime of a project. We acknowledge, however, that each project is different and that various process models, or combinations of them, are used in practice. In the figure, we have edited the original waterfall model, including a deployment and handover phase between the development and operational activities. We have also explicitly indicated the roles, and their possible overlap, in relation to the timeline. The indicated overlap is based on the following perspectives: a) software developers, after deployment, are regularly involved in operational activities for a limited period, for example, to resolve critical errors, and b) IT operations staff are needed in software development, for example, to help to identify the non-functional requirements and prepare for them prior to deployment.

---

* Corresponding author.
  E-mail addresses: jon.iden@nhh.no (J. Iden), bendikby@ifi.uio.no (B. Bygstad).
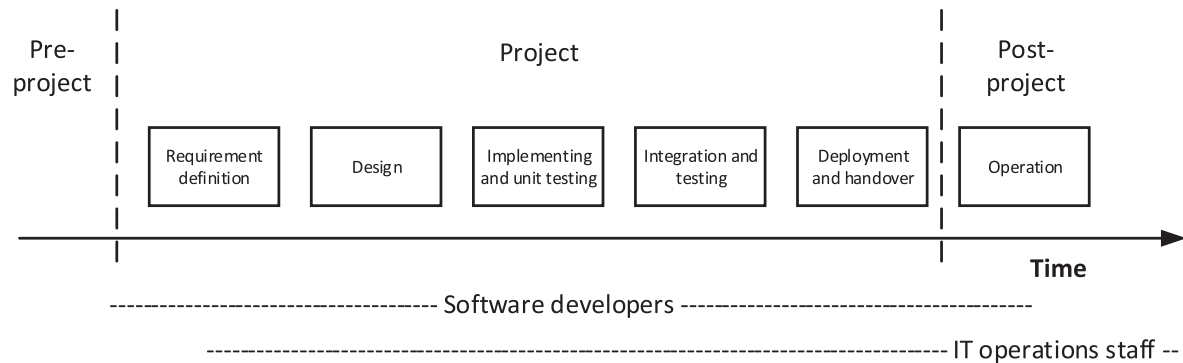
Fig. 1. The software development project.

From a research perspective, interactions in software development have mainly been investigated with regard to the interactions between developers and users (Ghobadi and Mathiassen, 2016; Majanoja et al., 2017; Park and Lee, 2014; Shim et al., 2010; Tesch et al., 2009). With some minor exceptions (Iden et al., 2011), the interaction of developers and personnel in IT operations has largely been neglected. While well-functioning interactions are mandatory, text books on software development (e.g. Sommerville, 2015) offer little guidance on this issue, and research has identified a number of serious problems that hamper developers and IT operation staff from collaborating effectively in projects (Iden et al., 2011). This, combined with several discussions with practitioners, has convinced us that understanding and improving the way in which developers and IT operations staff interact in software development projects should be a high priority.

Our aim is to contribute to the growing awareness regarding this topic, and we are therefore responding to the call for more qualitative research on software development from a social science perspective (Avenier and Thomas, 2015; Avison and Malaurent, 2013; Kautz et al., 2007; McLeod and MacDonell, 2011). We analysed data from three roles to understand the variety of behaviours and experiences. Specifically, we target the roles of 'software developer', 'IT operation staff' and 'system owner', the latter representing the user and the business side and having the overall responsibility for the procurement and development of an information system.

Our research objectives are as follows: (i) to explore how the three roles experience and observe the interactions of developers and IT operations staff in software development projects, and (ii) to integrate the findings into an assessment instrument that incorporates this knowledge, thus enabling project organisations to analyse and improve upon it. To address these objectives, we investigate the following research question: *What problems exist in the interaction of developers and IT operation staff in software development projects?*

2. Theoretical background and analytical framework

In this section, we outline the theoretical background and analytical framework for our study.

2.1. Software development and IT operations

The focus of this study is on the development of new software. Thus, we exclude on-going maintenance from our study, although we acknowledge that the boundaries between development, maintenance and operation might be ambiguous. Software development is about constructing new applications that support the needs of the business, and it is commonly organised through projects with fixed goals, budgets and time limits, normally under the responsibility of the software development function (Sommerville, 2015). Several methodological approaches to software development exist (Beck and Andres, 2004; Boehm, 1988; Jacobson et al., 1999; Royce, 1970; Schwaber, 2004). Conversely, IT operations are focused on running and controlling the various software in production, maintaining the IT infrastructure and handling incidents and user requests (Iden and Eikebrokk, 2013). IT operations are normally based in the computer centre and include such functions as service desks, operation systems and storage management, database and transaction management, networks and telecommunication and control and monitoring of online and batch production and networks (Cannon, 2011). Today, the work of IT operations is organised according to the principles of IT service management and is based on process-oriented frameworks, such as the Information Technology Infrastructure Library (ITIL). Typically, software development and IT operations are both functions within information systems (IS) organisations, but outsourcing and offshoring have introduced a variety of organisational arrangements. These characteristics are summarised in Table 1.

There are at least four concerns related to the interactions of developers and IT operations staff, as illustrated in Table 2. First, the professionals seem to be divided into two cultures, as developers and IT operations staff, due to different specialisations, often reside in different units or business organisations. The professionals may also be physically and geographically separated, creating challenges in communication and knowledge sharing (Mueller, 2015; Turkulainen et al., 2013). This organisational divide may hamper effective interactions. Second, although software development is an activity performed under the control of the software development function, the responsibility for operating the software is transferred to IT operations at