The Journal of Systems and Software 109 (2015) 229–242

Contents lists available at ScienceDirect

The Journal of Systems and Software

journal homepage: www.elsevier.com/locate/jss

A general theory of software engineering: Balancing human, social and organizational capitals

Claes Wohlin [a,*], Darja Šmite [a], Nils Brede Moe [a,b]

[a] Blekinge Institute of Technology, SE-371 79 Karlskrona, Sweden
[b] SINTEF, NO-7465 Trondheim, Norway

ARTICLE INFO

Article history:
Received 4 December 2014
Revised 30 July 2015
Accepted 7 August 2015
Available online 13 August 2015

Keywords:
Software engineering theory
Intellectual capital
Empirical

ABSTRACT

There exists no generally accepted theory in software engineering, and at the same time a scientific discipline needs theories. Some laws, hypotheses and conjectures exist, but yet no generally accepted theory. Several researchers and initiatives emphasize the need for theory in the discipline. The objective of this paper is to formulate a theory of software engineering. The theory is generated from empirical observations of industry practice, including several case studies and many years of experience in working closely between academia and industry. The theory captures the balancing of three different intellectual capitals: human, social and organizational capitals, respectively. The theory is formulated using a method for building theories in software engineering. It results in a theory where the relationships between the three different intellectual capitals are explored and explained. The theory is illustrated based on an industrial case study, where it is shown how decisions made in industry practice are explainable with the formulated theory, and the consequences of the decisions are made explicit. Based on the positive results, it is concluded that the theory may have a good explanatory power, although more evaluations are needed.

1. Introduction

Software development is a very knowledge-intensive activity. It is an engineering endeavour involving a lot of design, and the production is relatively simple. To develop software many different people interact within an organization. Thus, software development is hugely dependent on people (DeMarco and Lister, 2013). However, people alone are insufficient. Software development is to a very large extent a team effort, and hence the interaction between people and the complementarity in expertise are prerequisites to be successful. Furthermore, the organization in which the people work provides the infrastructure and environment to be able to leverage on the individual skills and their combined value. The organizational aspects relate to processes, methods, techniques and tools being part of the work environment. These three aspects are captured in the concept of intellectual capital. The objective of the paper is to formulate a general theory of software engineering from empirical observations of how industry actively works with human, social and organizational capitals (components of intellectual capital) to help explaining and reasoning about combinations of intellectual capital components (ICCs) to be successful in software development.

Intellectual capital may be defined as: "the sum of all knowledge firms utilize for competitive advantage" (Nahapiet and Ghoshal, 1998; Youndt et al., 2004). The sum of all knowledge means that the concept of intellectual capital encompasses all assets available to a company. Different divisions of intellectual capital into components exist. Here it is chosen to use the division discussed by Youndt et al. (2004). Some alternative divisions are briefly introduced in Section 2.1. Youndt et al. (2004) divide the general concept of intellectual capital into three ICCs: human capital, social capital and organizational capital. They are depicted in Fig. 1 together with the main level where it primarily resides, i.e., individual, unit and organizational, respectively. The ICCs are described in Section 2.

Here, the concept of a unit is used to denote an entity utilizing the three components of intellectual capital: human, social and organizational capitals, respectively. The unit may be a team, a department or any other entity for which it is relevant to discuss the concept of ICCs. A unit includes people, who possess a certain level of human capital through their experiences and expertise. It also has a social capital both in terms of how it can leverage on the social interaction within the unit, and how it uses its external contacts to create value. The external contacts and networks may include customers, internal

* Corresponding author. Tel.: +46 455 385820.
E-mail addresses: claes.wohlin@bth.se (C. Wohlin), darja.smite@bth.se (D. Šmite), nils.b.moe@sintef.no (N.B. Moe).

http://dx.doi.org/10.1016/j.jss.2015.08.009

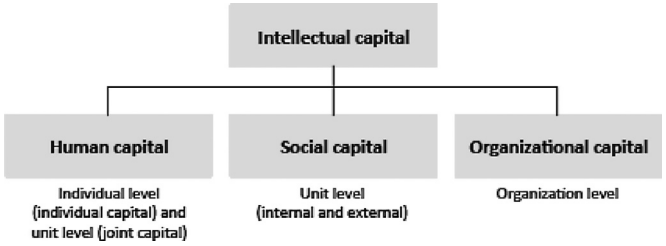

Fig. 1. Intellectual capital and its three components.

people in the organization, or external networks (including communities of practice, blogs and other external contacts and information). The unit exists in a context, which provides the organizational capital, for example, the support available to software engineers in terms of infrastructures. The latter includes all aspects of an organization that remain if removing all humans.

From the above reasoning, it becomes clear that the different components of intellectual capital are what make it possible to develop software. Based on this observation, this article contributes with formulating a theory of software development that captures the balancing of the ICCs that software organizations use in practice. Thus, the formulation of the theory is based on observations of practice and the insight that although organizations are different, they have a similar challenge. They need to balance the ICCs to be able to conduct their business in a cost-effective and competitive way. Balance refers to compensating loss in one ICC with improving either the same ICC or at least one of the other ICCs. The article presents the theory formulated and its constituents. Furthermore, it illustrates the theory in a real industrial case and also provides some examples taken from industrial collaboration.

The remainder of the article is structured as follows. Related work is presented in Section 2. Section 3 introduces the theory based on the steps recommended by Sjøberg et al. (2008). The theory is exemplified and illustrated by an empirical case in Section 4. In Section 5, a discussion is provided and the article is concluded in Section 6.

2. Related work

2.1. Intellectual capital and software engineering

In software engineering, there has been much discussion about how to manage knowledge, or foster "learning software organizations". In this context, Feldmann and Althoff have defined a "learning software organization" as an organization that is able to "create a culture that promotes continuous learning and fosters the exchange of experience" (Feldmann and Althoff, 2001). Dybå places more emphasis on action in his definition: "A software organization that promotes improved actions through better knowledge and understanding" (Dybå, 2001).

Because software development is knowledge-intensive work, intellectual capital is a particularly relevant perspective for software companies. Intellectual capital is called the main asset of software companies (Gongla and Rizzuto, 2001; Rus and Lindvall, 2002). It is seen as a construct with various levels (individual, network, and organizational) (Youndt et al., 2004). As mentioned above, Youndt et al. (2004) divide intellectual capital into three components: human, social and organizational capitals. This is not the only proposal for how to describe intellectual capital. Stewart (2001) describes the essential elements or assets that contribute to the development of intellectual capital as:

- Structural capital: Codified knowledge that can be transferred (e.g., patents, processes, databases, and networks).
- Human capital: The capability of individuals to provide solutions (e.g., skills and knowledge).
- Customer capital: The value of an organization's relationships with the people with whom it does business and share knowledge with (e.g., relationships with customers and suppliers).

The possession of each of these assets alone is not enough. Intellectual capital can only be generated by the interplay between them. Therefore, Willcocks et al. (2004) propose a framework, which also includes a fourth kind of ICC—social capital. Social capital helps to bring structural, human and customer capital together and encourages interplay among them.

Here it has been chosen to use the division of intellectual capital advocated by Youndt et al. (2004) for two main reasons. First, we agree with Youndt et al. that organizational capital is more fitting than the term structural capital because this is capital the organization actually owns (human capital can only be borrowed or rented). Second, both frameworks define social capital to consist of knowledge resources embedded within, available through, and derived from a network of relationships. We support Youndt et al.'s argument that such relationships are not limited to internal knowledge exchanges among employees, but also extend to linkages with customers, suppliers, alliance partners, and the like. We then see customer capital as part of social capital.

Creating intellectual capital is more complicated than simply hiring bright people. The importance of intellectual capital can be demonstrated by the ratio of intellectual capital to physical capital involved in the production of software. Symptomatically, the ratio of the software development industry is found to be seven times the ratio of other industries that are heavily reliant on physical capital, such as the steel industry (Bontis, 1997, 1998; Tobin, 1969). In a study on intellectual capital in Systematic Software Engineering Ltd, Mouritsen et al. (2001) found that the main motivation for understanding the different elements of intellectual capital was to make the company's knowledge resources and key competency areas visible and to monitor management's efforts to develop these. Also, management wanted to establish a new basis for deciding about the future of the company.

Youndt et al. (2004), through their review of intellectual capital, conceptualize intellectual capital through the three distinct components: human, social, and organizational. Human capital refers to individual employee's knowledge, skills, and abilities. In software engineering these are often associated with technical skills including design expertise, domain knowledge and product knowledge (Faraj and Sproull, 2000; Moe et al., 2014). Organizational capital represents institutionalized knowledge and codified experience stored in databases, routines, patents, manuals, infrastructures, and the like. Many traditional software companies that follow plan-driven approaches believe that a good process leads to a good product, and thus standardized and well-documented processes support developers, while interaction among software developers is usually minimized. Finally social capital consists of knowledge resources embedded within, available through, and derived from a network of relationships possessed by an individual or a social unit. Social capital is both the network and the assets that may be mobilized through that network (Bourdieu, 1986). It enables achievements that would be impossible without it or could only be achieved at an extra cost. Also, because social capital increases the efficiency of information diffusion, a company can have less redundancy in, e.g., skills or roles if the social capital is strong. An organization supports the creation of social capital when it brings its members together in order to undertake their primary task, to supervise activities, and to coordinate work, particularly in the context requiring mutual adjustment.

Different ICCs belong on different levels—individual, unit or organizational levels. While human and organizational capital components are rather straightforward, social capital is a more complex phenomenon. In the research on social capital, scholars have tended to adopt either an external viewpoint (the relations an actor