Evaluation of Knowledge Work Productivity – Case Distributed Software Development

Pekka Kamaja[1], Katriina Löytty[2], Mikko Ruohonen[2], Timo Ingalsuo[2]
[1] Haaga-Helia University of Applied Sciences, Helsinki, Finland
[2] CIRCMI, School of Information Sciences, University of Tampere, Tampere, Finland
pekka.kamaja@haaga-helia.fi
katriina.loytty@uta.fi
mikko.j.ruohonen@uta.fi
timo.ingalsuo@uta.fi

Abstract: The increased attractiveness of knowledge work offerings in cost competitive countries has pushed companies to seek for even more intensive forms of outsourcing. Striving for higher competitiveness and productivity, companies are looking beyond their current ways of outsourcing to create partnerships related to their core operations like research, development and innovation (RDI).

Today and in future the success of distribution of knowledge intensive efforts in globally operated businesses is dictated by game changing capabilities such as quality of services, dynamic competence building of individuals and teams, as well as management of decentralized work. Yet, these are just examples of a larger spectrum of factors manifesting the dynamism of distributed software development.

The DD-SCALE research program (2014-2016) has been targeted at investigating especially the aforementioned matters. The lack of feasible evaluation practises for explaining productivity of knowledge organizations in a comprehensive way has been an important driver. Managing and measuring productivity in the global settings of knowledge work has been in the focus of the program.

Thus far, a baseline concept of the DD-SCALE evaluation framework, containing some 90 aggregated indicators, has been constructed. These Capability Indicators are related to individual, team and organizational capabilities, aiming to explain productivity in software engineering organizations. The development work was supported by context related theories concerning distributed knowledge work and Intellectual Capital (IC), which has been a key in capturing the less visible elements of productivity. Intellectual capital lenses were utilized in analysing interviews and workshops conducted with managerial level software RDI professionals in two globally operating case-companies.

The contribution of this study is finding new dimensions for complementing the rather short-term input-output approach in assessing productivity. Also further elaboration about factors underpinning productivity in building a comprehensive evaluation framework is exercised. Eventually, the emphasis of this article is to bring forth the practical uses for the DD-SCALE evaluation framework.

Initial instantiations of the framework include assessing 1) the influence a change in an organization's operation model has on software developers' perceptions, and 2) organizational impacts of technology transfer activities. As further research, the plan is to evaluate the comprehensiveness of the framework and advance its development into an actual tool to support managerial decision making.

Keywords: knowledge work, global software engineering, productivity, evaluation framework, intellectual capital, organizational capability

1. Introduction

Knowledge work in software industry in its very core is about requirements analysis, software architecture design, setting targets and scheduling, software development, testing, deployment and delivery. In desire for higher productivity, the management methodologies of software development projects in the 90's paved way for the advent of delivery approach that builds software incrementally, known as Agile approach (Schwaber and Beedle, 2002, p.16). Since the beginning of the 2000's and the aim to increase efficiency of distributed software engineering, approaches such as DevOps, Continuous Integration (Vijayasarathy and Butler, 2016) and the Scaled Agile have extended the views of traditional Agile. Contributions of migrating to Agile methodologies can be significant. For instance, Hewlett Packard has reported outstanding reduction in development costs (Young, 2012, p. 25; Gruver and Mouser, 2015, p.17).

Eventually, these methodologies are not magic wands to bring quick wins, but frameworks that enable knowledge work based operations. As such, they call for a clear focus on how to utilize them. In desire for higher productivity, shorter time-to-market, higher quality and appropriateness to customer are ultimately the key requirements to gain productivity enhancements (Kamaja et al, submitted to EJKM). Moreover, these can be seen as drivers that have pushed software companies to seek opportunities in balancing midst the onsite, onshore, nearshore and offshore sourcing settings (Oshri, Kotlarsky and Willcocks, 2007), optimizing their resource network (Cantor, MacIsaac and Mannan, 2016), increasing the transparency of global resource distribution (Herbsleb, Kastner and Bogart, 2016) and re-structuring the forms of collaboration with clients. Yet, these are just a few examples of how companies have seized opportunities to improve their overall productivity. Besides changing the company boundaries, like in networking, new technologies in software engineering have offered opportunities for enhancements, like computerizing knowledge work based software testing activities (Frey and Osborne, 2014).

Evaluation of knowledge work productivity is troublesome if not impossible (Johnson, Mawson and Plum, 2014). The drilling down in an organization from the unit level to teams and individuals is shifting from general to particular. In most cases, the evaluation of productivity is exercised by means of key performance indicators (KPIs) or other evaluation metrics that are anchored to visible objects at unit level that are structural and organizational knowledge by nature. Eventually the enhancements of operations are carried out by people where the teams are seen as the most vital unit of knowledge work (Ibid.). However, bringing up the people dimension, human capital of individuals, easily blurs the visibility to the root causes underpinning productivity. Finding the wisdom of managing knowledge work productivity necessitates a comprehensive view on knowledge intensive organizations, encompassing organizational, human and customer relationship capitals, as suggested in intellectual capital literature (see e.g. Brooking 1996).

The key rationale behind the DD-SCALE RDI program (2014-2016) is managing distribution of software development work. This research was conducted together with two case companies that operate in software RDI intensive industries. Both of the companies have several sites distributed globally. The first, called Company A, is an international software and service company that provides IT-solutions for the telecommunications industry employing ca. 700 people. The second, Company B, is a large telecom network software and technology company with 20 000 employees.

The first of the objectives by company representatives was to shed more light on the impact of distribution on productivity. Thus a literature study and field work of the causes underpinning productivity was conducted first, which extended the knowledge and understanding of the complexities in explaining productivity. Results have been reported in former publications by the DD-SCALE research group. (e.g. Kamaja, Ruohonen and Ingalsuo, 2015; Löytty and Ingalsuo, 2015; Kamaja et al, submitted to EJKM)

Secondly, the case companies' decision makers were bothered especially by how to judge total productivity, which is predominantly dependent on the productivity of software engineering teams and developers. To meet these knowledge requirements, a baseline concept for a comprehensive evaluation framework was built. The purpose of the framework was to provide an assessment tool addressing the capabilities and the related, more focused, fragments of success factors in global software development work (Kamaja, Ruohonen and Ingalsuo, 2015).

Ultimately, an explanation to the question, how to judge productivity and, moreover, measure it, is complex and challenging and no universal yardsticks can be provided, especially when knowledge work is concerned (Johnson, Mawson and Plum, 2014). Nonetheless, DD-SCALE succeeded in stating the logic of factors impacting on productivity in distributed software engineering and outlining a comprehensive evaluation framework for distributed software engineering (Kamaja et al, submitted to EJKM). Questions such as, how can a framework, its dimensions and indicators be effectively implemented in practice and, more generally, what could be the practical uses of the framework are discussed in the Analysis section.

2. Theoretical background

This section discusses the theoretical background for the concepts applied in building the DD-SCALE evaluation framework. The foundation of the framework is casted on the basis of intellectual capital. Deciding upon the object of evaluation is, indeed, a dilemma of choosing between productivity and performance as the overall focus of evaluation. Lastly, the question of appropriate forms of evaluation metrics is discussed.

The dominant view in the framework is the dynamic intellectual capital approach (Ståhle and Grönroos, 2000) also termed Intellectual Capital with Dynamic Capabilities (Wu, Lin and Hsu, 2007). Intellectual capital as such is defined in terms of the well-known triplet holding human, structural and relational capital (Brooking, 1996; Roos and Roos, 1997). Human capital is the human capabilities, such as knowledge, skills, motivation, learning, social abilities. Structural capital is about organizational structures, structured knowledge and practices and processes. Relational capital consists of purposeful business relations, company external networks and brand (Huang, Luther and Tayles, 2007).

Intellectual capital is seen the most important source for sustainable competitive advantages in companies (Roos and Roos, 1997, p.8). Yet, the dynamism in intellectual capital is akin to the resource based view, which is furthermore interested in capabilities of building competitive advantage (Dierickx and Cool, 1989; Roos and Roos, 1997). In turn, "capabilities are all skills, processes, tools, and systems that an organization uses as a whole to drive meaningful business results" (McKinsey, 2014).

Making sense of the evaluation approaches, another discipline close to the intellectual capital frameworks is that of knowledge work performance, which is bound with explaining the individual, team and organizational knowledge work and working environment related assets (Bosch-Sijtsema, Ruohomäki and Vartiainen, 2009). Most of the appraisal tools within knowledge work are related to human and structural capital but also signs of relational capital can be found, like the consideration of customer perspective (Palvalin et al, 2014). Compared with the intellectual capital solutions, the knowledge work evaluation tools are more restricted in terms of explaining intangible assets.

Which approach to use then, productivity or performance? In its traditional form productivity, defined as quantifying productivity of work in terms of cost, effort and duration, is about inputs expended to produce the output (Tangen, 2005). This definition is apt for manufacturing industries where the boundary of observing the input and output is clear. In turn, in knowledge intensive organizations the contributions from inputs are dispersed downstream, thus perceived not only by the supplier organization but its customers and also customers' customers (Han, Chang and Hahn, 2011). That is why more extensive explanatory concepts of productivity are needed for knowledge intensive organizations such as the definition of productivity in terms of five main dimensions: cost, flexibility, speed, dependability or quality (Slack, Chambers and Johnston, 2001).

Making the choice between productivity and performance, it seems that productivity in explaining the effectiveness of knowledge intensive organizations becomes feasible at the level of teams and individuals where the boundaries and timespan of inputs and outputs are well-defined. In turn, performance is an even broader term that covers both overall economic and operational aspects (Tangen, 2005, p. 20) where the outputs are seen taking effect both in short and long term and is apt for entire organizations.

Next is the choice of metrics, which is challenging. For example, the productivity yardsticks that are linked with measuring velocity or quality of operations, offer a highly general view on evaluating operations. Moreover, they are used as key performance indicators which makes recognizing the difference between productivity and performance blurred.

When especially KPIs are favoured, typically not more than ten, they are the high level metrics in reference to the development, performance or position of the business of the company that are anchored to strategy (PricewaterhouseCoopers, 2007). One of the pitfalls of relying on KPIs is that they should be used to aid understanding and learning, not as targets. Sometimes managers are not knowledgeable enough of the operations, and the lack of understanding the rationale of a particular KPI can lead to biased information. When a KPI, such as innovativeness, is grounded on the count of initiatives and ideas (Parzefall, Seeck and Leppänen, 2008) that leaves out the innovative activity outside the strategic targets (Young and Kaiser, 2013) that is the emergent strategy at company grass roots (Mintzberg, 1994, p. 24).

A viable solution to avoid the lack of visibility to grass roots in an organization is to apply various performance metrics linked to levels of organization (Parmenter, 2007, p.2) or have dedicated systems such as performance management and capability planning systems that are integrated within one single system portfolio (Taticchi, Balachandran and Tonelli, 2012, p. 48).

3. Methods

Overall, the study followed the Design Science Research paradigm, where the aim is to design an artefact that answers to specified business needs or problems by incorporating the view of “Develop/Build” – “Justify/Evaluate” -cycle of the Information Systems Research Framework (Hevner et al, 2004, p. 80). Business needs are drawn from people, organizations and technology of the research “Environment”, such as a company or industry. Applicable knowledge, that is foundations consisting of theories, methods and methodologies, are drawn from the existing “Knowledge Base”. Within the design cycle, the built theories and artefacts are evaluated by means such as case and field studies, and other appropriate methods. The assessment of the evaluation feeds back into the design as refinements. As a result, the aim is to produce new applications to the Environment, and add new knowledge to the Knowledge Base. (Ibid.) This study covers the design of the baseline concept of the evaluation framework.

The research unfolded as a “search process” (Hevner et al, 2004, p. 82) together with the participating companies, where effective solutions were sought for answering the research problem and business needs (Hevner et al, 2004): Throughout the study, meetings and workshops were held among the case companies and research partners firstly, to define, analyse and formulate the initial research problem and later to reflect on the progress and interim results. After the initial goal setting, literature in the intellectual capital, performance management, distributed software development and knowledge work disciplines were investigated to gain understanding of the current state of research and focal concepts within the problem domain. This work provided a refined and sharper version of the research problem and enabled the researchers to devise interview plans and conduct workshops covering relevant themes.

The data was collected through interviews and workshops with the DD-SCALE research partners and international ICT professionals of the case companies. The 16 resulting transcriptions were analysed by qualitative, research data based content analysis with the aim to find and categorize relevant elements (Schreier, 2014) that would relate to productivity and software engineering work of high performance. The analysis approach was iterative and inductive – increasing the level of abstraction from individual elements towards higher level categorizations (Thomas, 2006) – with practices adopted from grounded theory methods (Corbin and Strauss, 1990) as the nascent results were refined during iteration rounds for better applicability to the problem domain. This analysis produced a preliminary model, or concept, for the desired framework for evaluating the elements of productivity in distributed software work. As the model formed, it was assessed and discussed in collaboration with the case companies with the aim to ensure its relevance to the organizations’ RDI activities.

Therefore, the study also adopts elements of action research (Baskerville and Wood-Harper, 1996). Essential to action research is that researchers and practitioners collaborate in the research efforts (Ibid.). Moreover, research subjects have a possibility to influence on the results (Ibid.), which is cut out for increasing their relevance in organizations (Baskerville and Myers, 2004). Traditionally, action research has two stages, the “diagnostic” stage for problem definition with research participants, and the “therapeutic” stage of introducing changes and evaluating their effects (Blum, 1995 cited in Baskerville and Wood-Harper, 1996, p. 237). For the support of scientific rigor, further steps have been defined, which can be applied depending on the study (Baskerville and Wood-Harper, 1996; Susman and Evered, 1978 cited in Ibid): “(1) diagnosing, (2) action planning,

(3) action taking, (4) evaluating and (5) specifying learning". Confluences between this study and these action research steps can be identified especially in the phases 1-2 and 4-5. The action taking -phase, 3, mostly remains as further work, while the evaluation and specifying learning -phases focused upon the applicability and relevance of the designed framework.

4. Analysis, results and discussion

The analysis of the transcribed data progressed iteratively from raw coding to a refined set of categorized elements. During the analysis the results-in-progress were processed among the researcher team with reflections to theory and other researchers' feedback. Atlas.ti -software was utilized in coding and categorizing the data. Increased understanding of the research object of productivity in distributed software work was gained by alternately viewing the data as a whole and as individual, broken up elements (Klein and Myers, 1999). The analysis resulted in a set of indicator candidates which were identified to be related to high performance and productivity in teams. These candidates were further refined by removing redundant items and by categorizing them in meaningful, representative entities of the phenomena (Corbin and Strauss, 1990; Nickerson, Varshney and Muntermann, 2013). (Kamaja et al, submitted to EJKM) Figure 1 illustrates the analysis process and its results at a high level.

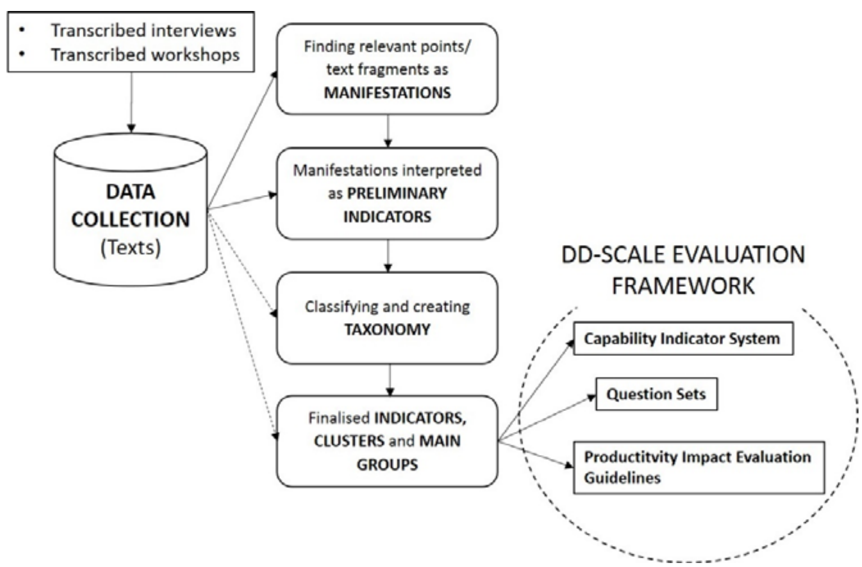

Figure 1: The analysis process and results

4.1 The framework

The construction of the DD-SCALE evaluation framework was grounded on the concepts discussed in the theory section. Human capital of individual employees is the engine that keeps the company moving towards the targets, but it requires to be underpinned by structural and relational capital as enablers. Put together they form meaningful bundles that are the capabilities, as vehicles, making the company capable of moving towards the targets.

The choice between the productivity and performance management ended up using them both. The concept of productivity becomes more feasible when the financial perspective is less pivotal. However, evaluation of the entire organization is in favour of performance approach. At the same time, the unit of evaluation in the DD-SCALE framework is capability, which means that the cycle of value stream engendered by a particular capability is relatively short. Accordingly, productivity was chosen as the object of measurement.

The overall breakdown structuring of the framework is seen in Figure 2 in two ways. First, the extension at the bottom of the figure highlights the levels of fragmentation down from the level of main groups to the clusters. Each cluster represents one capability indicator holding one or more root indicators that are the fragments of productivity of software (SW) development operations. (Kamaja et al, submitted to EJKM) As such, the sixteen main groups are divided into 88 clusters, and furthermore, to 320 indicators.

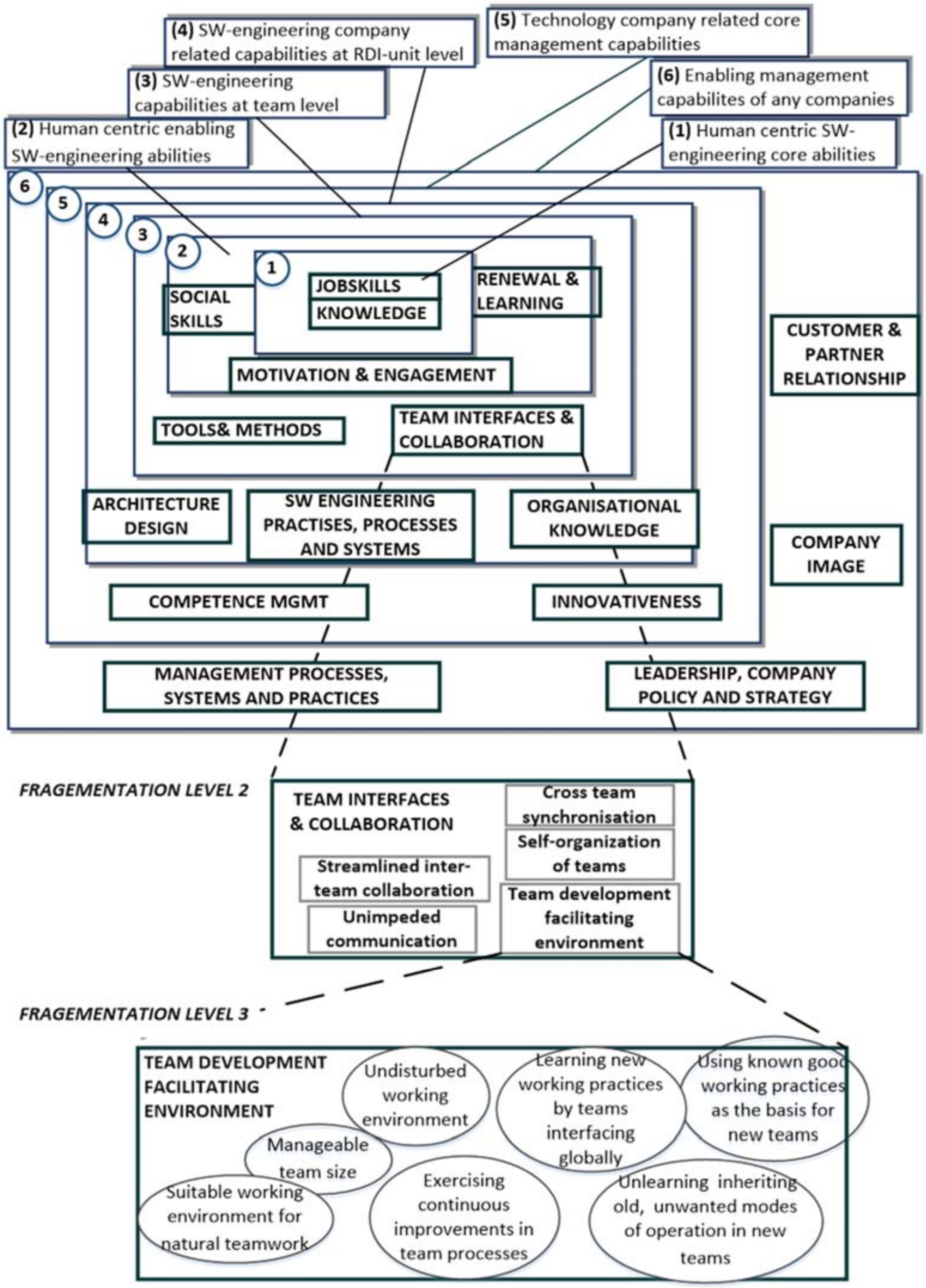

Figure 2: Capability indicators breakdown structuring with the six immediacy levels of productivity of intellectual capital (Kamaja et al, submitted to EJKM)

Second, the top part of the figure, the six nested boxes with numbered descriptions, depicts the main categories of productivity factors arranged on six levels. These levels are arranged according to their immediacy to productivity in software engineering: The closer a particular capability is to the topmost level – level 1, the more direct impact on productivity it is seen to have. The capabilities of Job skills and Knowledge are the seen as "the front line entities" that influence the total productivity of distributed software work. The capabilities on the level 6, such as those of Leadership, company policy and strategy, are seen as among the least direct ones. (Kamaja et al, submitted to EJKM)

In more detail, the level 1 is made of human related core capabilities, which are focal to successful, high performance software work and thus to productivity. The level 2 covers social skills, renewal and learning, and motivation and engagement which are understood as enablers for the level 1 capabilities. The level 3 focuses on team related capabilities, team interfaces and collaboration as well as tools and methods which facilitate successful collaboration across geographical and organizational boundaries. The software engineering specific management practices, architecture of product technology and structured organizational knowledge, such as artefacts and documents, on the level 4 contribute to managing the distribution and dispersion of resources. The level 5 management capabilities important to technology companies include innovativeness and competence management as central to sustainable and competitive operation. Finally, the level 6 contains the general management and leadership abilities as well as care of customer relationships and company image as the more general, fundamental capabilities of companies. Eventually, the capabilities on each level are strongly interconnected two-way. (Kamaja et al, submitted to EJKM)

4.2 Implications for practise

At the outset of the DD-SCALE program, the far-reaching expectation by the partner companies was to get a productivity evaluation methodology that ultimately would give financial estimates in assessing the yield versus sunk costs from the labour and other resources of software engineering. This expectation was complemented with that of an impact analysis tool, which would help to estimate the impact on productivity of technology transfer projects carried across sites. The target here was an investment analysis approach where the factors, intellectual by nature, would become recognized with tangibles.

Responding to these challenges was started by creating the aforementioned comprehensive explanatory model of productivity, which was the baseline concept. Along the research program other practical uses, to accompany the two expected tools, were identified. Couple of them were finalized and trialled. Indeed, the benefit of the DD-SCALE program was not targeted towards a few outcomes. It seems that practical applications deriving from the baseline concept were spinning off as the project progressed. This is especially thanks to the insightful ideas by company representatives on how to apply the research findings in practice for management purposes.

The first practical application, the first use, along the path of the joint research project was a comprehensive mapping of factors conceptualizing and aiming to explain productivity in distributed software work. The mapping, a relatively comprehensive and large concept, aimed to enrich the management's knowledge of the grass root causes that sometimes are obscure. The aim was not only to extend the managers' knowledge base, but also to utilize the mapping as a guideline in helping to ensure the overall quality of operations. For guideline purposes, the mapping was extended by more specific information about the essence of the 320 root indicators as the bottom line entities of explaining productivity. This was indeed the second use of the baseline concept.

The third service was focused on monitoring change in software design operations, more largely, change in the operation model. The preferred case by the Company A was testing the DD-SCALE framework for monitoring the restructuring of operation model according to the Scaled Agile Framework that is an overall management approach meant especially to strengthen the integrity across the levels of management and team operations. For this purpose, a question set for the evaluation of anticipated pain spots in change was planned. Then, a highly focused survey tool holding just twelve test questions was geared up on top of the entire range of 320 root indicators. The trial was successful and the Company A representatives were looking for new uses for the framework.

The future directions for the uses of DD-SCALE framework could be divided into the three perspectives. The first, is to strive for developing a solution for enhancing the project communication between management and teams in software development work. Due to heuristics in software development, an achilles heel is visibility on progress design work within and across the teams. Thus, the work can be stuck at several causes such as the inadequate information about the customer needs, less viable software architecture, poor requirement analysis of the case and so on. It would be likely that grounded on the findings on the levels of capability and root indicators, structured communication could be defined or the means of communication according to the forums of Agile approach could be complemented.

The second line in promoting the usability of the DD-SCALE framework, is to develop an analytical productivity assessment tool which necessitates a careful consideration of the cycles of inputs and outputs and their relatedness. For example, improvements of factors increasing the velocity of design work are seen quite instantly as increased yield. Inputs on the quality factors of software operations require a longer period for materializing the outputs, whereas inputs on customer appropriateness of software feature development can be seen in sustaining the customer relationship as well taking effect in the long run. Although an accurate prediction model explaining the effects between single inputs and outputs is impossible, it would be possible to develop a productivity impact tool by using a right amount of approximation.

The third goal is striving for a comprehensive capability framework that can be applied for evaluation of company preparedness to meet planned objectives or, in general, for performance review purposes. This would enable companies to diagnose their capability gaps in a systematic, objective way. Moreover, companies can better establish a foundation for the effective design of learning programs that link learning results to the business and include meaningful, quantitative targets.

References

Baskerville R. and Myers M. (2004) "Special Issue on Action Research in Information Systems: Making IS Research Relevant to Practice – Foreword", *MIS Quarterly,* Vol 28, No. 3, pp 329-335.

Baskerville, R.L. and Wood-Harper, A.T. (1996) "A Critical Perspective on Action Research as a Method for Information Systems Research", *Journal of Information Technology*, Vol 11, pp 235-246.

Bosch-Sijtsema, P.M., Ruohomäki, V. and Vartiainen, M. (2009) "Knowledge Work Productivity in Distributed Teams", *Journal of Knowledge Management*, Vol 13, No. 6, pp 533-546.

Brooking, A. (1996) *Intellectual Capital: Core Assets for the Third Millennium Enterprise*, International Thomson Business Press, London.

Cantor, M., MacIsaac, B. and Mannan, R. (2016) "Steering Software Development Workflow: Lessons from the Internet." *IEEE Software*, Vol 33, No. 5, pp 96-102.

Corbin, J. and Strauss, A. (1990) "Grounded theory research: procedures, canons and evaluative criteria", *Qualitative Sociology*, Vol 13, No. 1, pp 3-21.

Dierickx, I. and Cool, K., (1989) "Asset Stock Accumulation and Sustainability of Competitive Advantage", *Management Science*, Vol 35, No. 12, pp 1504-1511.

Frey, C.B., and Osborne, M.A. (2014) "The Future of Employment: How Susceptible Are Jobs to Computerisation?" A conference paper in University of Lund, Sweden. Oxford Martin Programme on Technology and Employment. [online] Available at: http://www.oxfordmartin.ox.ac.uk/downloads/academic/future-of-employment.pdf [Accessed 19 October 2016].

Gruver, G., and Mouser, T. (2015) *Leading transformation: Applying Agile and DevOps Principles at Scale,* IT Revolution, Portland.

Han, K., Chang, Y. B. and Hahn, J. (2011) "Information Technology Spillover and Productivity: The Role of Information Technology Intensity and Competition", *Journal of Management Information Systems*, Vol 28, No. 1, pp 115-145.

Herbsleb, J., Kastner, C. and Bogart, C. (2016) "Intelligently Transparent Software Ecosystems", *IEEE Software*, Vol 33, No. 1, pp. 89-96.

Hevner, A.R., March, S.T., Park, J. and Ram, S. (2004) "Design Science in Information Systems Research", *MIS Quarterly*, Vol 28, No. 1, pp 75-105.

Huang, C. C., Luther, R. and Tayles, M. (2007) "An evidence-based taxonomy of intellectual capital", *Journal of Intellectual Capital*, Vol 8, No. 3, pp 386-408.

Johnson, J., Mawson, A., and Plum, K. (2014) "Effectively Measuring Knowledge Worker Productivity", White Paper. Allsteel Inc., Iowa. [online] Available at: http://www.allsteeloffice.com/SynergyDocuments/WPA_Insight_WorkerProductivity.pdf [Accessed 16 October 2016].

Kamaja, P., Ruohonen, M. and Ingalsuo, T. (2015) "Challenges in the Intellectual Capital Evaluation for Dynamic Distributed Software Development Teams – DD-SCALE Program in Progress", In Cegarra Navarro, J. G. (ed.) (2015) *Proceedings of the 7th European Conference on Intellectual Capital* (ECIC2015), Cartagena, Spain, 9-10 April, 2015, pp 173-180.

Kamaja, P., Ruohonen, M., Löytty, K. and Ingalsuo, T. (submitted to EJKM) "Intellectual Capital Based Evaluation Framework for Dynamic Distributed Software Development", *The Electronic Journal of Knowledge Management*. [online] Available at: http://www.ejkm.com/main.html [To be published].

Klein, H.K. and Myers, M.D. (1999) "A set of principles for conducting and evaluating interpretive field studies in information systems", *MIS Quarterly*, Vol 23, No. 1, pp 67-93.

Löytty K., and Ingalsuo T. (2015) "Factors Impacting Successful Collaboration and Productivity of Distributed Software Development Teams", In Öörni, A., Iivari, N., Kuutti, K., Oinas-Kukkonen, H. and Rajanen, M. (eds.) (2015) *Proceedings of the 38th Information Systems Research Seminar in Scandinavia* (IRIS 38), Oulu, Finland, August 9-12, 2015.

McKinsey. (2014) "Building capabilities for performance", Discussion paper. [online] Available at: http://www.mckinsey.com/business-functions/organization/our-insights/building-capabilities-for-performance [Accessed 8 November 2016].

Mintzberg, H. (1994) *The Rise and Fall of Strategic Planning*, Prentice Hall.

Nickerson, R.C., Varshney, U. and Muntermann, J. (2013) "A method of taxonomy development and its application in information systems", *European Journal of Information Systems*, Vol 22, No. 3, pp 336-359.

Oshri, I., Kotlarsky, J. and Willcocks, L. (2007) "Managing Dispersed Expertise in IT Offshore Outsourcing: Lessons from Tata Consultancy Services", *MIS Quarterly Executive*, Vol 6, No. 2, pp 53-65.

Palvalin, M., Vuolle, M., Jääskeläinen, A., Laihonen, H. and Lönnqvist, A. (2014) "SmartWoW – Constructing a tool for knowledge work performance analysis", *International Journal of Productivity and Performance Management*, Vol 64, No. 4, pp 479-498.

Parmenter, D. (2007) *Key performance indicators: developing, implementing, and using winning KPIs*, John Wiley & Sons.

Parzefall, M., Seeck, H. and Leppänen, A. (2008) "Employee innovativeness in organizations: A review". [online] Available at: http://lta.hse.fi/2008/2/lta_2008_02_a2.pdf [Accessed on 10 November 2016].

PricewaterhouseCoopers LLP. (2007) "Guide to Key Performance Indicators - Communicating the Measures That Matter". [online] Available at: www.corporatereporting.com [Accessed 25 October 2014].

Roos, G. and Roos, J. (1997) "Measuring your company's Intellectual Performance", *Long Range Planning*, Vol 30, No. 3, pp 413-426.

Schreier, M. (2014) "Qualitative Content Analysis", in Flick, U. (ed.) (2014) *The SAGE Handbook of Qualitative Data Analysis*. SAGE Publications Ltd.

Schwaber, K. and Beedle, M. (2002) *Agile Software Development with Scrum*, Prentice Hall.

Slack, N., Chambers, S. and Johnston, R. (2001) *Operations Management*, 3rd ed., Pearson Education Limited, Harlow.

Ståhle, P. and Grönroos, M. (2000) *Dynamic Intellectual Capital*, WSOY.

Tangen, S. (2005) "Demystifying productivity and performance", *International Journal of Productivity and Performance Management*, Vol 54, No. 1, pp 34-46.

Taticchi, P., Balachandran, K. and Tonelli, F. (2012) "Performance Measurement and Management Systems: State of the Art, Guidelines for Design and Challenges", *Measuring Business Excellence*, Vol 16, No. 2 pp 41-54.

Thomas, D.R., "A General Inductive Approach for Analyzing Qualitative Evaluation Data", *American Journal of Evaluation,* Vol 27, No. 2, pp 237-246.

Vijayasarathy, L.R. and Butler, C. W. (2016) "Choice of Software Development Methodologies: Do Organizational, Project, and Team Characteristics Matter?" *IEEE Software,* Vol 33, No. 5, pp 86-94.

Wu, S-H., Lin L., and Hsu, M-Y. (2007) "Intellectual Capital, Dynamic Capabilities and Innovative Performance of Organizations", *International Journal of Technology Management*, Vol 39, No. 3-4, pp 279-296.

Young, F. and Kaiser, K. (2013) "Value Based Management", INSEAD. [online] Available at: http://executive.education.insead.edu/value-based-management.php [Accessed 10 November 2016].

Young, M. (2012) "A Practical Approach to Large-Scale Agile Development. A Hewlett Packard case study: The transformation of LaserJet FutureSmart Firmware". [online] Available at: http://www.agileleadershipnetwork.org/wp-content/uploads [Accessed 4 November 2016].