Aalto University
School of Science
Degree Programme in Information Networks

Eevert Saukkokoski

# What exactly is the use of dailies:

## Evolution of practice and process in a software development team

Master's Thesis Research Plan
Espoo, 2015

**DRAFT! — November 16, 2015 — DRAFT!**

Supervisor: Professor Riitta Smeds
Advisor: Otso Hannula M.Sc. (Tech.)

# Contents

# Chapter 1

# Introduction

## 1.1 Background and motivation

Agile software development practices have become mainstream (West et al., 2010). The Scrum methodology (Schwaber, 1995) belongs among the most popular (West et al., 2010), being heralded as virtually a de-facto industry standard (Marchenko and Abrahamsson, 2008).

Necessity of face-to-face interaction is emphasized in agile literature as critical for transfer of ideas and achieving innovative results. (Highsmith and Cockburn, 2002) In the agile manifesto, this was considered important enough for it to take first place on a list of agile values: "individuals and interactions over processes and tools". (Beck et al., 2001)

The "daily scrum", a meeting of developers deriving its name from the Scrum methodology, as the most prominent component of the leading individual methodology (West et al., 2010), can be claimed to have become a symbol of sorts for agility itself.

Agile process models have been further characterized with simplicity and ease of adaptation as key (Abrahamsson et al., 2002). For instance, time constraints of a daily may be relaxed or the general agenda modified if it is found that they do not aid in reaching desired goals (Marchenko and Abrahamsson, 2008). Agile is represented as the antithesis of dogma, being fit for critical inspection and malleable to use-case specific needs. Yet its

nature seems to have changed with introduction to the mainstream (West et al., 2010), with Marchenko and Abrahamsson (2008) citing problems like "too many meetings" and disciplined effort required to "keep it simple" as present challenges.

- Kanban endorses "starting where you are" and evolving your process from there
- "Evolution": incremental improvement through an iterative process of "seeing what sticks"

## 1.2 Research problem

> *How do dailies enable process evolution in a software development team?*

Agile processes and practitioners' understanding thereof evolve with use. I am interested in discovering whether the "heart" of agile, namely the daily meetup, has a relation to enabling this evolution. Can the daily itself be seen as a tool for gaining knowledge not only on proceedings, but the processes and practices applied?

## 1.3 Timeline

**November 2015** Kickoff, research plan drafted, collection of empirical data started.

**December 2015** Empirical data collection complete. Beginning of empirical data analysis.

**January 2016:** Theoretical framework assembled.

**February 2016:** Empirical data analysis complete.

**March 2016:** Interpretation of empirical data through theory.

**April 2016:** Intensive writing period. Completion of theoretical synthesis. Empirical conclusions.

**May 2016:** Thesis completed.

# Chapter 2

# Theoretical framework

## 2.1 Theoretical approach

In my thesis I will review literature from software development process, organizational, and sociological research to create a theoretical synthesis.

Depictions of agile methodology adoption, practice and evolution:

- Rise of agile into the mainstream and the abandonment of agile orthodoxy (West et al., 2010)
- Agile practice and culture (Sharp and Robinson, 2004; Robinson and Sharp, 2005; Robinson et al., 2007)
- Challenges in agile adoption (Nerur et al., 2005; Marchenko and Abrahamsson, 2008)

(Problem: Most "agile" literature seems to take the perspective of Scrum, XP et al, where I'd be more interested in a Kanban & Lean perspective.)

Understanding the agile organization as an evolving sociotechnical system:

- Roots of agile principles (Vidgen and Wang, 2006; Nerur et al., 2010)
- Systems of problems and interactive planning (Ackoff, 1997)

Characterization of scrum / kanban boards or virtual ones:

- The role of story cards (Sharp et al., 2006)

- Visualizations as objects of continuously unfolding epistemology (Ewenstein and Whyte, 2009)
- Boundaries as enablers of and barriers to innovation (Carlile, 2002)

How might new information be created by manipulating aforementioned artefacts:

- New knowledge creation by intersubjectively accepted novel distinctions (Tsoukas, 2009)

## 2.2 Theoretical research questions

- How does process evolution take place in knowledge work?
- What are the prerequisites for process evolution in knowledge work organizations?
- How can the practice of a daily support process evolution in a software development team?

# Chapter 3

# Empirical study

## 3.1 Empirical approach

- Ethnographically-inspired studies of agile practices have been undertaken before (Robinson et al., 2007; Marchenko and Abrahamsson, 2008) and similar approaches would seem valid in this context
- Microethnography (Streeck and Mehus, 2005) sounds like a suitable empirical lens considering that we are interested in seeing if and how speech and manipulation of cognitive artefacts might yield knowledge on the problem of processes and work organization

## 3.2 Empirical research questions

- How is the daily used as a practice for furthering goals by individual team members?
- How does manipulation of process artefacts within a daily affect practices outside it?
- How is knowledge created within the proceedings of a daily?

## 3.3 Research setting

- Helsinki-based software startup with a headcount of 20 people
- In focus is the software development team with 10 people
- The team's daily meeting around a collection of virtual "kanban boards" can be taken as an opportunity to learn about how the boards are used and the consequences of their use

# Chapter 4

# Conclusions

## 4.1 Practical implications

- Dailies are a huge cost: they are synchronous and take up a significant slice of developers' time
- Because no tangible "work" gets done, the only justifiable use of dailies is the potential for knowledge transfer and creation
- Exposition on the mechanisms of knowledge creation within dailies will allow organizations to structure their use of the time taken for more of the intangible benefits desired

# Bibliography

P. Abrahamsson, O. Salo, J. Ronkainen, and J. Warsta. *Agile software development methods. Review and analysis.* VTT Technical Research Centre of Finland, 2002.

Russell L Ackoff. Systems, messes and interactive planning. *The Societal Engagement of Social Science*, 3(1997):417–438, 1997.

K. Beck, M. Beedle, A. Bennekum, A. Cockburn, W. Cunningham, M. Fowler, J. Grenning, J. Highsmith, A. Hunt, R. Jeffries, B. Kern, J Marick, R. C. Martin, S. Mellor, K. Schwaber, J. Sutherland, and D. Thomas. Manifesto for agile software development. 2001. URL `http://agilemanifesto.org`. Noudettu 2010.10.23.

Paul R Carlile. A pragmatic view of knowledge and boundaries: Boundary objects in new product development. *Organization science*, 13(4):442–455, 2002.

Boris Ewenstein and Jennifer Whyte. Knowledge practices in design: the role of visual representations as 'epistemic objects'. *Organization Studies*, 30 (1):07–30, 2009.

J. Highsmith and A. Cockburn. Agile software development: The business of innovation. *IEEE Computer*, 34:9(9):s. 120–127, 2002. ISSN 0018-9162.

A. Marchenko and P. Abrahamsson. Scrum in a multiproject environment: An ethnographically-inspired case study on the adoption challenges. In

*Agile, 2008. AGILE '08. Conference*, pages 15–26, Aug 2008. doi: 10.1109/Agile.2008.77.

Sridhar Nerur, RadhaKanta Mahapatra, and George Mangalaraj. Challenges of migrating to agile methodologies. *Communications of the ACM*, 48(5): 72–78, 2005.

Sridhar Nerur, Alan Cannon, VenuGopal Balijepally, and Philip Bond. Towards an understanding of the conceptual underpinnings of agile development methodologies. In *Agile Software Development*, pages 15–29. Springer, 2010.

Hugh Robinson and Helen Sharp. Organisational culture and xp: three case studies. In *Agile Conference, 2005*, pages 49–58. IEEE, 2005.

Hugh Robinson, Judith Segal, and Helen Sharp. Ethnographically-informed empirical studies of software practice. *Information and Software Technology*, 49(6):540 – 551, 2007. ISSN 0950-5849. doi: http://dx.doi.org/10.1016/j.infsof.2007.02.007. URL `http://www.sciencedirect.com/science/article/pii/S0950584907000110`. Qualitative Software Engineering Research.

K. Schwaber. Scrum development process. In *OOPSLA Business Object Design and Implementation Workshop*, pages 10–19, 1995.

Helen Sharp and Hugh Robinson. An ethnographic study of xp practice. *Empirical Software Engineering*, 9(4):353–375, 2004.

Helen Sharp, Hugh Robinson, Judith Segal, and Dominic Furniss. The role of story cards and the wall in xp teams: a distributed cognition perspective. In *Agile Conference, 2006*, pages 11–pp. IEEE, 2006.

Jürgen Streeck and Siri Mehus. Microethnography: The study of practices. *Handbook of language and social interaction*, pages 381–404, 2005.

Haridimos Tsoukas. A dialogical approach to the creation of new knowledge in organizations. *Organization Science*, 20(6):941–957, 2009.

Richard Vidgen and Xiaofeng Wang. Organizing for agility: A complex adaptive systems perspective on agile software development process. 2006.

Dave West, Tom Grant, M Gerush, and D D'silva. Agile development: Mainstream adoption has changed agility. *Forrester Research*, 2:41, 2010.