Agile and Lean: Organizations, Products, and Development

Jeffrey Saltz
Syracuse University
jsaltz@syr.edu

Edward Anderson
University of Texas
Edward.Anderson@mccombs.utexas.edu

Alex Sutherland
Scrum Inc.
alex.sutherland@scruminc.com

Abstract

Over the past two decades, research in the area of agile and lean software development has mirrored the strong growth of the use of agile and lean methodologies. Agile and lean management practices (which we define broadly to include Scrum, XP, Lean Startup and other related approaches) roughly triple the success rate of software projects over traditional management approaches. Because software projects contribute so broadly to economic and social improvement, research on agile methods may produce significant productivity gains. However, much work remains to enable all the benefits of agile and lean concepts to be realized.

1. Introduction

An agile approach focuses on using a cycle of experimentation, inspection and adaptation to improve production. Agile is most often applied to software development, and we expect many papers in this mini-track to discuss software organizations and software engineering practices. However, we also welcome papers that describe other types of organizational "production", such as business intelligence, management initiatives, manufacturing, marketing, sales and finance.

A lean approach focuses rapid experiments by trying to continually reduce waste and minimizing work-in-progress. Lean has recently been popularized as a construct for start-up organizations ("Lean Startup" or "Lean Entrepreneurship"). Advocates claim a lean approach produces greater market satisfaction and customer engagement, earlier discovery of hidden market opportunities, higher revenues and more efficient use of development staff.

These approaches claim superiority in new product development over traditional approaches (such as "waterfall management") that fail to test development and market assumptions in long-range plans.

Agile and lean approaches challenge organizations large and small. People typically conflate small failures (learning) with large failures (organizational threats), assume that innovation means taking long-range untested risk, and establish and protect budgets with many baked-in production and market assumptions. These cultural realities interfere with agility and real innovation.

As a result, companies often invest enormous amounts of money in incomplete or abandoned agile transformations. What can organizations do to improve agile uptake? How do we know that the organization is improving? How can organizations diagnose problems without motivating gaming? What types of people are more likely to thrive in agile and lean organizations, and what roles should they take? What hiring practices result in better candidates? What training programs produce better results? What coaching structures work? How do we measure these activities?

The Agile/Lean mini-track explores these questions – to better understand agile and lean methods and their effects on quality, speed and communication. We solicited research papers and experience reports that explored agile development, lean product management and agile/lean organizations, and that we, as a community, help to ensure relevance and rigor [1].

2. Sessions

At this year's conference, we divide the papers into two loosely related themes. The first theme focuses on new and enhanced processes and frameworks. The second theme focuses on case studies to identify challenges and opportunities.

2.1. New or Enhanced Process Framework

This mini-track starts with "Everyone's Going to be an Architect: Design Principles for Architectural Thinking in Agile Organizations", where Horlach et. Al. propose six design principles to realize architectural thinking in agile organizations. The results are based on insights from interviews with sixteen employees and consultants with expertise on architecture management and organizational agility across several industries.

HICSS

This is followed by "In for a Penny, in for a Pound? A Lifecycle Model for Agile Teams", where Diegmann, where Dreesen, and Rosenkranz derive a lifecycle model of agile teams as well as threats to their success, based on interviews across a variety of industries and organizational contexts. Their model also includes pathways for teams to discard agile methods if these do not fit the team's needs.

In the fourth paper, "SKI: A New Agile Framework that supports DevOps, Continuous Delivery, and Lean Hypothesis Testing", Saltz & Sutherland explore the need for a new process framework that can effectively support DevOps and Continuous Delivery teams. There new framework, Structured Kanban Iteration (SKI), adheres to the lean Kanban philosophy, but augments Kanban by providing a structured capability-based iteration process (as opposed to Kanban-like no iterations or Scrum-like time-based sprints).

2.2. Case Studies

In "Towards Empirically Validated Remedies for Scrum Retrospective Headaches", Matthies and Dobrigkeit explore retrospective meetings, which are Scrum's instrument for process improvement and adaptation by presenting case studies of educational and industry teams, investigating the effects of eleven retrospective activities on five identified headaches.

In the next case study "Towards A Lean Innovative Approach to Rethinking Employees Turnover. Surviving with Less-Knowledge, but not Knowledge-less: A Case Study", Miller investigates what happens when the employees with critical knowledge leave. The paper seeks to identify the root impacts of the employee departure from the Lean ideal. Specifically, over a 3-year real-life case study, Miller explored and analyzed the implications of turnover in an industrial setting. The emphasis was to re-think the way organizations deal with turnover; The study suggests retaining organization knowledge, rather than retaining staff, through utilizing the lean methods to operate with less knowledge, but not knowledge-less!

In the third case study "The Impact of Modes, Styles, and Congruence of Control on Agile Teams: Insights from a Multiple Case Study" Dreesen, Diegmann, and Rosenkranz discuss the fact that agile software development (ASD) strongly relies on social interaction and teamwork. Their objective was to improve our understanding of how to enact control in agile teams and how these control mechanisms influence team autonomy and team performance. In this paper, they present their findings from four case studies conducted within two insurance companies and two software development firms. They found that it is not a question of 'what' controls should be exercised, but rather 'how' controls are implemented in practice.

This is followed by Hassani-Alaoui, Cameron, and Giannelia's paper, " 'We use Scrum, but …': Agile modifications and project success", where they explore how scrum changes in practice and how these changes impact various aspects of project success. Through interviews with representatives from 11 organizations who use scrum for software development, they found variability in the application of the guidelines, namely, that only a small number of guidelines are systematically followed, and that some guidelines are rarely followed consistently.

Finally, in the last case study, "On Solving the Business Requirements Engineering Problems of Information Systems Development Projects – Lessons from Three Projects", Dahlberg and Lagstedt conducted three case studies to investigate requirements engineering problems, and the reasons for them. Their focus was on how to synchronize business processes and information system development requirements in plan-driven (waterfall) and change-driven (agile) projects. The investigated cases indicate that the ontological and epistemological matching of information system and business process requirements engineering methods improves requirements quality.

3. References

[1] Tripp, J., Saltz, J., & Turk, D. (2018). Thoughts on Current and Future Research on Agile and Lean: Ensuring Relevance and Rigor, in *Hawaii International Conference on System Sciences (HICSS)*, 2018.