Systematic Review Protocol

Agile Practices in Distributed Software Development

Raoul Vallon, Bernardo José da Silva Estácio, Rafael Prikladnicki, and Thomas Grechenig

This Web extra complements the Voice of Evidence article “ADAPT: A Framework for Agile Distributed Software Development.” It describes in detail how we performed the systematic literature review that we used—as part of coauthor Raoul Vallon’s PhD thesis—to systematically analyze work related to our ADAPT (Agile Distributed Adaptable Process Toolkit) framework for applying agile practices to distributed software development (DSD). It covers studies on agile DSD practices from 1999 to 2014.

Study Design

We built on a literature review and mapping studies of agile DSD practices that Samireh Jalali and Claes Wohlin\(^1\) conducted between 1999 and 2009. We also consulted guidelines that Kai Petersen and his colleagues\(^2\), and Barbara Kitchenham and Stuart Charters\(^3\) developed for designing systematic mapping studies. In addition, we incorporated Fabio da Silva and his colleagues’ experiences in study design.\(^5\)

We limited our search to peer-reviewed studies in English with the search terms: (agile OR scrum OR “extreme programming” OR “pair programming” OR “lean development” OR “lean software development”) AND (“global software development” OR “global software engineering” OR “distributed software development” OR “distributed software engineering” OR “distributed software development” OR gse OR gsd OR “dispersed team” OR “spread team” OR “virtual team” OR offshore OR outsource). We searched the ACM Digital Library (http://dl.acm.org), the AIS Electronic Library (http://aisel.aisnet.org), EI Compendex (www.engineeringvillage.com), IEEE Xplore (http://ieeexplore.ieee.org/Xplore/home.jsp), Inspec (www.theiet.org/resources/inspec/index.cfm), and Scopus (www.scopus.com).

We also manually searched the proceedings of the most relevant conferences from 2010 to 2014: the Agile Alliance Conference (AGILE), the International Conference on Agile Software Development (XP), and the International Conference on Global Software Engineering (ICGSE).

Review Inclusion Process

Our initial search yielded 288 potentially relevant studies. We then excluded 48 because they were duplicate studies, or didn’t present work or results in progress papers, experience reports, books, theses, or workshop papers. We read the remaining studies’ abstracts and eliminated corner cases, leaving 152 papers to consider. Finally, after a full-text analysis, we selected 95 studies to investigate.

Data Extraction and Synthesis

We extended Jalali and Wohlin’s classification of distribution types, which was inspired by Darja Šmite and her colleagues’ research,\(^6\) as follows:

- **location**—offshore (different countries), onshore (same country), or unclear;
- **legal entity**—outsourcing (different companies), insourcing (same company), or unclear;
- **geographic distance**—far (at least two hours’ flying time), near (less than two hours’ flying time), or unclear; and
- **temporal distance**—large (time zone difference of more than four hours), small (four hours or less), or unclear.
If a study represented more than one distribution type—such as a multisite environment with two sites involved in onshoring and one in offshoring—we assigned it the type that represented a more complex software-development environment because that would make the entire project more complicated. In our example, offshoring would be the more complex environment because it could require dealing with time-zone, cultural, and other differences.

Šmite and her colleagues used different wording for describing geographic and temporal distance (time zone difference), depending on whether the case entails offshoring (for which they used the terms “far, near” and “large, small” respectively) or onshoring (for which they used the terms “close, distant” and “similar, different” respectively). For our systematic review, we used only their offshoring terms, even for describing onshoring cases, because this made it easier to categorize all of the studies.

We classified distributed teams as isolated, integrated, or unclear. In an isolated distributed team, members in different locations work remotely on independent tasks. In an integrated distributed team, members in different locations work on the same tasks. These terms correspond to what Jalali and Wohlin called distributed and virtual teams respectively.

Also conforming to Jalali and Wohlin’s research, we classified the following:

- project size—small (up to 20 people), medium (21 to 50 people), large (more than 50 people), or unclear; and
- project duration—long (more than seven months), medium: (one to seven months), short (less than one month), or unclear.

We based our knowledge areas on the SWEBOK Guide. If a study featured multiple cases, we extracted agile practices for each case.

References

Raoul Vallon is a postdoctoral researcher at TU Wien, where he leads the AMMA (Amazing Makers) work group on empirical software engineering. He developed the ADAPT (Agile Distributed Adaptable Process Toolkit) framework as part of his PhD research. Contact him at raoul.vallon@inso.tuwien.ac.at.

Bernardo José da Silva Estácio is a PhD student in Pontifícia Universidade Católica do Rio Grande do Sul’s Computer Science School. Contact him at bernardo.estacio@acad.pucrs.br.

Rafael Prikladnicki is an associate professor in the Computer Science School and director of the Science and Technology Park at Pontifícia Universidade Católica do Rio Grande do Sul, where he also leads the
MuNDDoS research group on global software development. Contact him at rafaelp@pucrs.br.

**Thomas Grechenig** is a professor of industrial software engineering and head of the Industrial Software Group at TU Wien. Contact him at thomas.grechenig@inso.tuwien.ac.at.