

Commonalities of Agile Methodologies

Position paper for the workshop “Commonalities of Agile Methodologies”, OOPLSA 2002

Klaus Marquardt
Käthe-Kollwitz-Weg 14, D-23558 Lübeck, Germany
Email: marquardt@acm.org

Copyright © by Klaus Marquardt

Agile development has become popular to an extent that most people want to do it, or at least try it – and all the other people want to sell their expertise and their tools. There is a lot of promise in agile development – but what makes it different to “non-agile” approaches?

Commonalities

- First things first. There is no replacement for real, functioning code, and none it sought. This also implies a no-excuses-policy, see no failure mode.
- No failure mode. In all agile methods, failure is not an option – some even exclude the project termination from their scope. All individual preparations for looking good or at least innocent in the (unlikely ☺) case of failure are banned. Alternatives are not considered by process (see first things first), but the participants are allowed to look for them if necessary (see empowerment).
- Option space. While failure of the project is not an option, all secondary things are kept open as long as possible. Any decision is avoided that unnecessarily closes an option dimension. Each decision is done in a way that is possibly even opens more options than before.
- Empowerment. The projects’ participants are allowed to do whatever helps them to fulfill their task, according to first things first.

These commonalities might explain the differences in reception of agile development and management ideas. Most developers love to be empowered and focus on the things they consider important. Some managers are quite reluctant because seems to be harder to control the project. The definition of tasks and deliverables strives to close options in order to be able to predict what will happen the next weeks, and it might be received as a loss of empowerment by individual developers. Only experienced, self confident managers know that a loose control by giving priorities and attitude is mostly much more valuable than a tight control through work packages and deliverables.

Why are the commonalities interesting?

In first order approximation, the commonalities are not interesting. Name is sound and smoke. It does not matter whether your process might be called somehow, as long as it supports the project participants, stakeholders, and their respective needs.

In second order approximation, it can be interesting to classify a process and allow for easy communication about it. Similar to design patterns, process patterns might be identifiable by their names and can quickly be checked whether they are appropriate for the project and organization.

The third order approximation might be interesting in the future. Development process assessments try to express the capability of the organization. Currently, the underlying engineering models focus on a waterfall like process model. Identifying the commonalities of agile methodologies could help to replace this model, and assess organizations against a different scale.