

Adopting Agile Development in the Large

2018-02-01

Many large organizations have traditionally used *waterfall* software development models. Because of the flaws and inefficiencies of the waterfall model large organization are looking to adopt *agile* development models. The adoption of agile development model requires transition that can be challenging and take considerable amount of time and effort. Part of the challenge is scaling the agile model, because it was developed for smaller organizations with team size of roomful of people. Agile practices such as customer involvement, informal communication over documentation, continuous integration and frequent system builds often require changes withing the organization and the codebase. This essay discusses about the benefits, drawbacks and the adoption problems of the transition into agile development and about agile development in large organizations.

Agile Development in Large Organizations

Large organization and systems contain components that violate the principles of agile development. They also require a number of adaptations to work. Sommerville (2010) lists a number these challenges and adaptations:

- 1) Large systems consists of multiple parts, often developed in separate from each other by separate teams that maybe in other places. This also makes it practically impossible to have a view of the whole system.
- 2) Large systems interact with other existing systems, which may not be compatible with agile development practices.

- 3) System configuration takes larger portion of development time in large integrated systems. This is not compatible with agile development practices.
- 4) Development of large systems is often constrained by external rules and regulations which limit the development.
- 5) Due to long produrement and development time, it is challenging to maintain coherent teams with knowledge of the system over the whole development period.
- 6) It is impossible to involve the whole set of stakeholders into the development process since large systems have lots of stakeholders.

In order for large organization to adapt agile methods they have to maintain the fundamentals for agile development such as *flexible planning*, *frequent system releases*, *continuous integration*, *test drive development* and *good team communications*. Critical adaptations are

- 1) More up-front design and system documentation, software architecture has to be designed, documentation about the critical aspects of the system.
- 2) Cross-team communication mechanisms, for instance through phone or video.
- 3) Continuous integration may be practically impossible, but organization should maintain frequent system builds and releases of the system.

Agile Development at Ericsson

Lassenius and Engblom discusses about transformation from waterfall model to using scrum at Ericsson developing a mobile core network node (Lassenius

2013). To indicate the scale of the transformation, the development of the network node had been on going since 1990s, 50 million lines of code had been written and 400 people in 40 teams and 3 different locations were developing it.

Engblom described that the main problems in waterfall model were

- Too big scope resulted in lead times of two years of development time were no longer viable for 21st century software.
- Responsiveness and flexibility were not good enough and changes would take too long to implement.
- Quality of the software would degrade since size of the tests kept growing.
- Motivation inside teams was suffering since there was no room for developers to grow and they had limited view of the whole picture.

The reasons for choosing agile described were to avoid oversized specification and the use of direct communication over documents. Engblom also describes how they scaled the agile practices inside Ericsson

- Initial testing with one team revealed problems with time boxing, also test and build times were found to be too long. This was a problem for implementing continuous integration.
- Cross-functional teams were implemented and component guardians who were available for teams to consult for specific knowledge about specific parts of the system.
- Scrum masters were chosen from former line and project managers and later from developer team members. Each team also had a product proxy owner.
- To implement communication between teams, application was divided into areas and the teams communicated at area level and with teams that were working on same features.
- Workspace was physically changed from a traditional offices where every one was in their own room into more open space where people were together to improve communication and transparency. Physical change also visualized the change of the development method.

Noted benefits and improvements that changing into agile development model were savings in administrative costs, average feature cost decreases, release interval time has decreased and positive change in the atmosphere.

Bibliography

Lassenius, C., 2013. *An interview with christian engblom at ericsson finland discussing their large-scale agile transformation*. [online]. Available from: https://lassenius.files.wordpress.com/2013/01/engblom_mastered.mp3 [Accessed 2 Feb 2018].

Sommerville, I., 2010. *Software Engineering: 9th Edition*. 9th ed. USA: Addison-Wesley Publishing Company.