Agile management principles applied to a collaborative EU-project

In the SWITCH-ON project, we have aimed at applying best practices agile principles for the project management. While the approaches are slightly different in the various work-packages (product development, the collaborative research, and knowledge brokering), they all are based on iterative and incremental (agile) work with continuous tests and feedback.

1 The Agile Approach

The "agile approach" is the umbrella term for several iterative and incremental development methodologies used in the SWITCH-ON project. Broadly, the agile approaches share the following common core values:

"Iterative and incremental, Deliver value early and often". The project deliverables are assembled from the results of several smaller iterations. Each small iteration completes a small portion of the final deliverable. Typically, the duration of such iteration is a few weeks or a few months (with preference to the shorter time-scale). There are several benefits with the iterative approach:

- By working in iterations of working useful results, we can constantly show end-users new features and gather feedback to make sure we are doing the right thing. Each iteration is made "minimal" but not too small (the iteration still has to be "marketable" to some end-user).
- By working with shorter iterations we can reduce the long-term planning work and use the resources to create working solutions rather than planning documents.
- We can detect scheduling flaws (e.g. wrong time estimates for activities) earlier, and address these issues before they cause problems in other parts of the project.
- For WP4, demonstration of the iteration results provides an excellent progress monitoring for the project. Many argue that the only real measure for progress in development project is actual running software.
- Important requirements can be implemented first.
- At any given time in the project there exists a working, tested system.
- Time for software development is not predictable. Instead of measuring progress through written status-reports, we can see the progress through a steady sequence of new features implemented.

"If it isn't on the portal, it does not exist". Most agile approaches require that the results from iteration are made operational. It is not uncommon to underestimate the amount of work to transition from a prototype to a real operational system. By making each partial result available online, we find out problems with operational issues much earlier (rather than finding the problems during the last year of the project when it is too late to fix them). Additionally, system integration can be tested much earlier. Too many project wait until the last year of the project to integrate systems. By making results operational, we do this work much earlier and also find out what kind of problems you may encounter.

"Adapt to change": In contrast to plan-driven project, we do not assume that all the requirements are well-known at the beginning of the project. We know that new requirement will be discovered during the development phase. Especially the area of open data is continuously evolving and expanding. By quickly adapting to changes in the open data, wehope our product developers will gain a competitive advantage. Similarly, hydrology research is still evolving fast and we need to harness new scientific results in the project. Hence, responding to change is more important than strictly following outdated plans.

2 Managing development of water-information products

For the development of the 14 water-information products, we have adopted an iterative and incremental approach based on the twelve principles in the agile manifesto (http://agilemanifesto.org/principles.html) . http://en.wikipedia.org/wiki/Agile_software_development

Many project plans contain separate phases for definition of requirements, design of products, implementation, testing and finally marketing and sales (usually called waterfall project plan). Usually, the project has a fixed time-plan for the development phase but as more and more IT-professionals now admit, the time estimation of software development is notoriously difficult. The tests and feedback from the real users comes during the last year of the project. In the end, most projects experience some delays which means that the later stages of implementation, testing and marketing has to be cut short.

Studies in US and UK have shown that approximately 75% of software systems developed using the waterfall methodology were failures or never used. You cannot get the perfect results if you only get only chance. You cannot implement the perfect software if you only get one chance.

3 Managing agile research

The iterative and incremental approach was also successfully leveraged for the collaborative research. Rather than working with ambitious research experiments that run over several years, we define smaller experiments that can be carried out within a few months. Faster research cycles also allow us to better harness new open data sets made available for experiments. For the collaborative science, we have come to value:

- Interaction of scientists over formal processes
- Feedback early in the procedure of setting up an experiment
- Incremental design of protocols that allow repetitiveness and reproduction of results
- Responding to change or bad experience over following an outdated plan

4 Summary

All the principles and practices used in SWITCH-ON have been known for many years. They have proven track record giving higher quality results in shorter time compared to traditional project approaches. Main sources of inspiration have been:

- Scrum (Ken Schwaber)
- XP (Ken Beck)
- Lean software development (Mary Poppendieck)
- Adaptive software development (Jim Highsmith)
- Feature-driven development (Jeff DeLuca)

Since SWITCH-ON value sharing, collaboration, open communication, trust, and teamwork, we have found that the agile approach has worked well.

5 Cross-WP working groups

Pyramidal organisation structures inevitably lead to latency in communication between workpackages. In SWITCH-ON, we try to avoid these delays by setting up working groups containing staff from several WPs. One notable example of such a cross-WP team consists of members from WP1, WP2 and WP3.

For more info, see the following webcast: http://www.versionone.com/Webcasts/Agile_PM_for_PMPs.asp