

Implementing Traceability in Agile Software Development

Marcus Jacobsson (traceability@hanabi.se)

2009-01-25

Introduction

How traceability is handled in agile methods vary from organization to organization, and project to project. When working with traditional development traceability is an important part of the process. However for some reason this is not the case when working with agile methods. For example in Scrum there is no real definition of how configuration management is supposed to be done. Can we add traceability to Scrum and still call it Scrum?

The initial problem for this thesis was to look into how traceability was supposed to be performed in agile methods, do we need traceability? Another problem that came up during the research is the question if traceability needs to be heavy, does it have to add administrative overhead for the team? Does the team need to write documents that risk not being up to date? Is it better not to have tracing information then having incorrect information?

Results

The studies have resulted in a list of practices that can be used to add traceability to the agile methods. These practices can be used by themselves. But they will give more value for the money if they are combined with each other. In the thesis report there are some examples of how the practices can be combined.

A big part of the studies is the discussions about in what way and under what circumstances traceability should be added. Adding traceability in the wrong way till only result in an overall cost for the project while doing

it the right way will result in an overall benefit for everyone involved in the project.

Another focus was on the support of tools and how they can help reduce the administrative overhead of adding traceability to the project. If tracing is supposed to succeed in agile methods such as Scrum and XP, it needs to add some value to the project. Adding something that will only result in a cost in the end needs to be avoided.

Conclusion

Tracing done right will result in higher product quality as well as less time working on the project (time spend from gathering requirements until the product is faced out). There is however no silver bullet when it comes to tracing and it should be adapted to each project and organization. Most of the job with tracing will be done by the team and one of the claims that are relevant to this is "*Write once, read many*" [1]. The information saved during development could save the time for many team members, testers, managers and maintenance teams.

Traceability should not be implemented in every project and especially not fully fledged traceability. In order to get the most out of the agile methods, traceability needs to be adapted. There are several considerations that should be taken into account when adding traceability.

References

- [1] **Programmer as Reader**
IEEE Software 1987
Adele Goldberg