

Agile og EVISOFT bedriftssamling på Kvittfjell, 17-19 mars 2009



Program, deltakere, presentasjoner og oppsummering

AGILE

EVISOFT
Evidence based Improvement of Software engineering

Organisasjonskomitè:

Tor Ulsund, Geomatikk

Haakon Spilde, KnowIT Objectnet

Nils B. Moe, SINTEF IKT

Torgeir Dingsøy, SINTEF IKT

Geir K. Hanssen, SINTEF IKT

Program

Tirsdag 17. mars - middag 20.00

Onsdag 18. mars

0900 – Fellesesjon

Kieran Conboy – Method Tailoring. Findings from Tailoring
XP - Current Practices and Problems

Brian Hanley – Efficient self-organizing teams

Ben Hoskins – Test Driven Development

Jan-Erik Sandberg – Moving to enterprise Agile - At the
breaking point

1130 – Lunsj

1230 – Ski for de som vil

1600 – Tre paralleller med lyntaler fra bedrifter, forskere og inviterte
foredragsholdere. Tema er:

- 1) Smidig produktutvikling i et organisasjonsperspektiv. Hvilken
rolle har produkteier, hvordan skreddersy smidig utvikling til
produktutvikling, hvordan planlegge releaser
- 2) Smidig produktutvikling i prosjektperspektiv. Hvordan
organisere support og vedlikehold, hvordan få til et
selvstyrende team,
- 3) Smidig testing

1900 – Slutt

1930 – Afterski/idemyldring på temaer til open space

2030 - Middag

Torsdag 19. mars

0900 - Åpen diskusjon (open space technology)

1200 – Lunsj

1300 – 1700 Ski/Workshop/open space/andre møter/avreise

Deltakere

Jan-Erik Sandberg	Det Norske Veritas
Finn Prytz	DNV Software
Brian Hanly	Exoftware
Ben Hoskins	Exoftware
Tor Ulsund	Geomatikk
Magnus Grøtan	Geomatikk
Torbjørn Meistad	Geomatikk
Haakon Spilde	Know IT Obejctnet
Anders Breivik	Know IT Objectnet
Arne Nylund	Kongsberg Spacetec
Nils Jakob Villmones	Kongsberg Spacetec
Frank J. Øynes	Kongsberg Spacetec
Børge Punsvik	Kongsberg Spacetec
Jessica Hildrum	Programutvikling
Nils Brede Moe	SINTEF
Børge Haugset	SINTEF
Geir Kjetil Hanssen	SINTEF
Torgeir Dingsøyr	SINTEF
Emil Røyrvik	SINTEF
Kieran Conboy	University of Galway

(Muligens ikke 100% siden vi hadde noen endringer i siste liten.)

Openspace-referater *(fra de som har sendt inn)*

What can we learn from research on agile development?

Facilitator: Torgeir Dingsøy

Some studies have been conducted on agile software development methods, mainly on extreme programming. However, most of the studies to date are done on teams that have less than one year of experience with agile development.

Findings from the studies concentrate on introduction and adoption of agile methods, human and social aspects, perceptions of agile development, and comparative studies (traditional versus agile development).

Some of the comparative studies show productivity data for traditional and agile teams. Such data can be very misleading, a high number of lines of code produced per hour is not necessarily good. Comparative studies need to incorporate broader indicators of success, from quality of code to developer motivation. A central idea in agile development is iterations, and investigating the effect of releasing parts of the product to customers could be an interesting direction. Identifying other core aspects of agile development, and connecting to research in other fields can be ways of making more clear lessons from research that the software engineering industry could benefit from.

Estimation Techniques Using Delphi.

Facilitator: Ben Hoskins

How to quickly expose risk and get team commitment to estimates through wisdom of crowds, blind ballot, high-low spreads and the median. a.k.a. "Try to guess the weight of my motorbike."

How are requirements specified by/with the customer?

[This was more of a talk with topics wandering quite heavily, and this also means notetaking was close to impossible. If nothing else, the notes will serve as reminders for the attendees.]

How are requirements specified by/with the customer?

Attending: Spacetec, Geomatikk, Exoftware (after a while), SINTEF

Facilitator: Børge Haugset

Usually, both for Spacetec and Geomatikk, their customers arrive with quite rigorous requirement specifications.

Geomatikk had one project where the customer wrote user stories - that was a success.

According to Brian, customers actually 'hate the huge spec' that is traditionally used. He claims it is still possible to do BDD in a rigorous (as in fixed scope/fixed cost) specification, you had to focus on internal values and be sure to answer the tender as acceptance tests.

[We wandered into a talk on governmental contracts using fixed scope/cost, and the status of 'agile contracts' in Norway]

What are design documentation used for? Quite often very little. At Spacetec they had projects where new developers would go through the user manuals, tests, and other code. They skipped reading any design docs (IIRC they didn't have any?) Brian says, 'the code is the only part of the documentation that can't lie'

[On developers writing tests..] People (devs) are by nature lazy, and hence don't want to write tests. It is seen as extra work. One must in some way teach devs that they need to write their code to be able to hook up tests more easily.

Last item we talked about was Spacetec, which explained about a 'test and training' document to the customer. When the customer was done with the training, the system was also accepted. (I realize there are a lot of logic jumps lacking to get to this conclusion, but I added it in case it seems interesting for someone.)

What barriers exist preventing the agile paradigm to be embraced beyond the software development community?

Facilitator: Finn Prytz

Agile thinking contrasts markedly with the traditional software development processes in omitting the requirements specification.

Purchasers are unsettled with the absence of a specification and a plan as they are used to see. They are used to asking: “What will I get, when will I get it and at what price?” With agile, the answer will not be as expected.

We are up against a massive management and legal culture in engineering, construction and procurement in which the requirements specification is an essential reference basis for contracts and agreements. With the absence of a specification the customers tend to feel lured into a scheme, free to spend their money without any promise of what will come in return.

An explicit definition of the contract/agreement reference documentation to replace the requirements specification might be a clue to approaching the prevailing purchasing culture.

A scientific, quantitative demonstration of the advantages of agile with respect to ROI and risk management, which might have an impact on this culture, is not knowingly yet produced. In the meantime, we are given to argue that agile provides unprecedented project transparency and flexibility, and thereby a great risk management opportunity.

Tilbakemeldinger

What was the best about these days

- Etter ski på onsdag
- Open space
- Location
- Alle har de samme problemene
- Foredragene
- Isolasjon - folk er her hele tiden. vanskelig å stikke av

Kommentar. Vi kom hit på grunn av programmet, og ikke fordi vi skulle stå på ski. Det er skiuavhengig.

Hva var ikke så bra

- De fra utsiden bør få vite litt om de som er tilstede. Alle kjenner hverandre i prosjektet.
- En liste over deltakere og deltaker lapper
- Flere bedrifter
- Dyrt