

software pilots

TRIFORK.

Scrum er ikke en religion

Jesper Boeg, Agile Coach
jbo@trifork.com

January 25, 2011

In general

- Trifork
 - Software development
 - Coaching
 - Workshops, seminars and conferences
- Please let me know if:
 - You have questions (The primary goal is not to go through every single slide)
 - If I am not making any sense

**HOW MANY BELIEVE AGILE IS
THE RIGHT WAY TO GO?**

WHAT DOES AGILE SOFTWARE DEVELOPMENT MEAN?

Agile Manifesto

A painting of a group of people in a meeting, with the text overlaid. The scene is set in a room with a window in the background. Several people are gathered around a table, some looking at a document or screen. The style is somewhat impressionistic with visible brushstrokes.

www.agilemanifesto.org

We are uncovering better ways of developing software by doing it and helping others do it.

Feb 11-13, 2001

Snowbird ski resort, Utah

Kent Beck
Mike Beedle
Arie van Bennekum
Alistair Cockburn
Ward Cunningham
Martin Fowler
James Grenning
Jim Highsmith
Andrew Hunt

Ron Jeffries
Jon Kern
Brian Marick
Robert C. Martin
Steve Mellor
Ken Schwaber
Jeff Sutherland
Dave Thomas

Agile Manifesto

Feb 11-13, 2001 Snowbird ski resort, Utah

We are uncovering better ways of developing software by doing it and helping others do it.

Through this work we have come to value:

Individuals and interactions over **processes and tools**
Working software over **comprehensive documentation**
Customer collaboration over **contract negotiation**
Responding to change over **following a plan**

That is, while there is value in the items on the right (red), we value the items on the left more (blue).

**BUT THE AGILE MANIFESTO
TELLS US VERY LITTLE
ABOUT WHAT AGILE IS!**

**THE PROBLEM WITH THE
AGILE MANIFESTO IS THAT IT
IS A “MANIFEST”**

MANIFESTS = REVOLUTION



**BUT AFTER THE
REVOLUTION...**

**WHAT IF WE COULD DEFINE
AGILE WITHOUT HAVING TO
STATE WHAT IT IS NOT?**

The Manifest does not mention:

- Deliver early and often
- Decentralization of authority
- Fast feedback
- Customer value
- Customer satisfaction
- Quality
- Continuous improvement



THE FORGOTTEN PRINCIPLES BEHIND THE MANIFESTO

Agile Principles 1/2

- Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
- Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage
- Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
- Business people and developers must work together daily throughout the project.
- Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.

Agile Principles 2/2

- The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
- Working software is the primary measure of progress.
- Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely
- Continuous attention to technical excellence and good design enhances agility
- The best architectures, requirements, and designs emerge from self-organizing teams
- Simplicity--the art of maximizing the amount of work not done--is essential
- At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly

**BUT THERE IS A REASON WE
FORGET THEM**

The Principles are:

- Many (12)
- Inflexible
 - *Business people and developers **must** work together daily throughout the project*
- Long and hard to remember (182 words)
- Old (2001)
 - Deliver working software frequently, from a **couple of weeks** to a couple of months, with a preference to the shorter timescale
- Redundant
 - *Our highest priority is to satisfy the customer through **early** and **continuous delivery** of valuable software*
 - **Deliver** working software **frequently**, from a **couple of weeks** to a couple of months, with a preference to the shorter timescale
- Unable to tell you why!

WHY IS THAT A PROBLEM?

**BECAUSE WE MAKE THE
WRONG DECISIONS!**

**LET US LOOK AT A FEW
EXAMPLES!**

AGILE WILL CREATE CHANGE

AGILE RELY ON QUALITY



AGILE DEMANDS TRUST IN PEOPLE

It seems we need two things

1. A shorter, clearer and more memorable definition of Agile
2. A better understanding of theory behind Agile values

**MAYBE ACADEMIA CAN HELP
US?**

Academic definition of Agile

- The continual readiness of an ISD method to rapidly or inherently **create** change, proactively or reactively **embrace** change, and **learn** from change while contributing to **perceived** customer **value** (economy, quality, and simplicity), through its collective components and relationships with its environment.

– Kieran Conboy, 2009

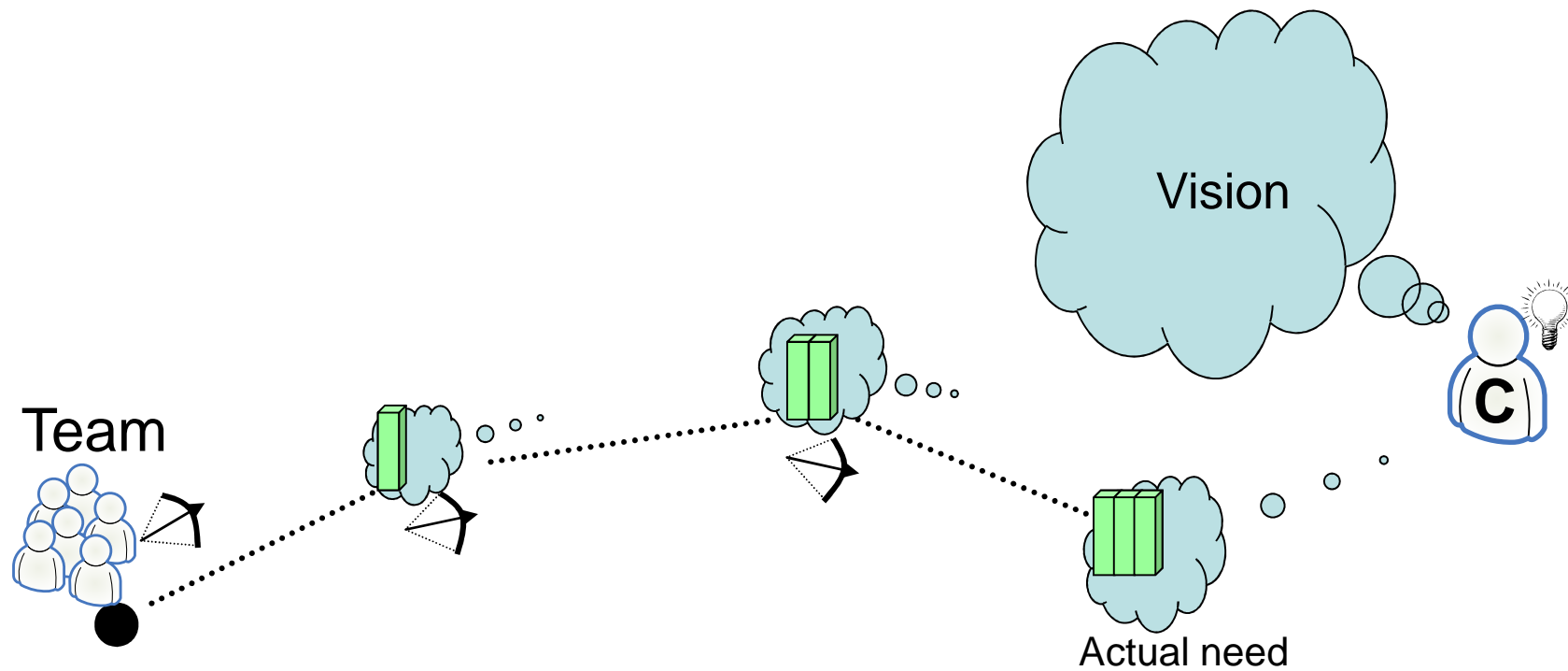
**WHAT IF WE HAD A FEW
SHORT STATEMENTS TO
HELP US REMEMBER**

Suggestion:

- 1. Satisfy the Customer through Working Software**
- 2. Deliver Early and Often**
- 3. Create and Embrace Change**
- 4. Focus on Quality**
- 5. Create Transparency through Visualization**
- 6. Endorse Sustainable Pace**
- 7. Bring People Closer Together**
- 8. Trust in People and Decentralize Authority**
- 9. Improve Continuously**

**ESSENTIALLY ALL WE WANT
TO DO IS BUILD THINGS IN A
SERIES OF SMALL STEPS**

Adaptive approach



WHY IS SCRUM A GOOD WAY TO START?

Scrum gives you:

- An easy prescription for applying Agile principles to software development
- A framework that is easy to understand and where the basics can be learned in two days
- A common ground for discussing Agile roles, artifacts and ceremonies
- A management framework with the possibility of implementing eg. XP practices

Scrum's Way of Being Agile

1. **Satisfy the Customer through Working Software** ✓
 - PO, Sprint Delivery and Backlog
2. **Deliver Early and Often** ✓
 - Sprints, Working Software
3. **Embrace Change** ✓
 - Sprint Demo
4. **Focus on Quality** ✓
 - Potentially Shippable Product Increment, Sprint Demo
5. **Visualize Workflow** ✓
 - Scrum board
6. **Endorse Sustainable Pace** ✓
 - Velocity, Sprint Planning
7. **Bring People Closer Together** ✓
 - Scrum Teams, Sprint Demo, Co-location
8. **Trust in People and Decentralize Authority** ✓
 - Self organizing teams with full Authority and Autonomy
9. **Improve Continuously** ✓
 - Retrospectives, Scrum Master

**IS SCRUM THE SILVER
BULLET THEN?**

NO!

- Many Scrum teams have historically proven to suffer from all or some of the following:
 - Break down Madness
 - Poor Quality
 - Cargo Cult Batch Sizes
 - Sub Optimization
 - Suboptimal Synchronization
 - Too Much Work In Progress

1. Breakdown madness

- Items small enough to fit a 2 week iteration are often too small to deliver real business value
 - Test becomes waste
 - Retrospectives become waste
 - Feedback becomes waste



Breakdown Madness

support booking seats

Breakdown Madness

show seat

select seat

Breakdown Madness



good sense of half victory
fake sense of half victory

2. Poor Quality

- Fixed iteration goals stress the entire system:
 - Product owners rush to prepare for upcoming cycles
 - Testers race to complete work late in the development time-box
 - Developers prioritize finishing a set of features over refactoring, TDD and pair programming



3. Cargo Cult batch sizes

- God told us that all Agile projects will fit 2-4 week development cycles without:
 - Regard to business
 - Deployment cost
 - Feedback quality
 - Competition
 - Minimal marketable features



4. Sub optimization

- Lack of focus on the entire delivery
 - Development is by definition seen as the bottleneck
 - Clear definition of roles restrict flexibility
 - PO, Development, Test and Operation become silos
 - Protecting the sprint without considering the consequences

5. Suboptimal Synchronization

- Prioritization, delivery, inspection, reflection and planning are synchronized to maximize periods of undisturbed work
 - Delaying feedback
 - Reducing flexibility
 - Applying the lowest common denominator



6. Too much work in progress

- Enormous backlogs
- All items on the sprint backlog in progress
- **Sprint is almost:** tested, reviewed, released



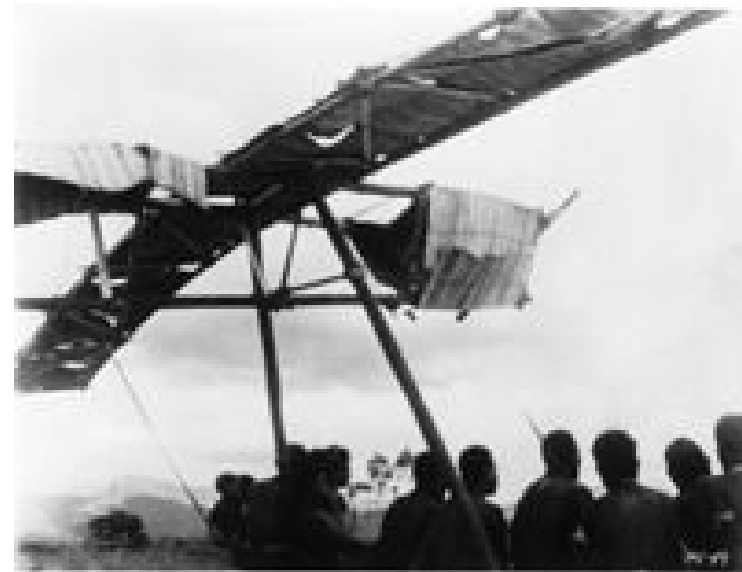
WHAT WE WANT IS TO AVOID CARGO CULT AGILE

Best practices

- Sound deceptively easy to apply
- The problem is that best practices are context independent
- Which would be fine if the project was too 😊

Cargo Cult Agile Implementations

- Once practices become faith based and cargo cult we risk loosing sight of the goal



THE THEORY BEHIND LEAN AND KANBAN CAN HELP US BETTER UNDERSTAND AGILE VALUES

AND AVOID AGILE ANTI-PATTERNS

**BUT THAT IS OUT OF SCOPE
FOR TODAY'S TALK**

Key takeaways

- Scrum is a good way to kick-start your Agile implementation
- If we focus solely on practices we will make the wrong decisions.
- If we do not understand “why” we will make the wrong decisions
- A clear, short, rememberable definition of Agile can be beneficial
- Kanban and Lean theory can help us better understand Agile principles

What can Trifork offer?

- Scrum Master Certification
- Scrum Product Owner Certification
- 1 day Scrum Introduction
- Scrum Kick-Start
- Kanban Training with David Anderson
- 1 Day Kanban Introduction
- Lean in Product Development
- Experienced Agile Coaches
- www.trifork.com