

software pilots

TRIFORK.

Agile from the Outside

Jesper Boeg
VP Trifork Agile Excellence
jbo@trifork.com
April 7, 2011

In general

- Please let me know if:
 - You have questions
 - If I am not making any sense
- Comment!
 - This will be a lot more fun if it is not just me talking



WHO ARE WE?

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

**WHAT DOES AGILE MEAN TO
YOU?**

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 Manifest does not mention:

- Deliver early and often
- Fast feedback
- Customer value/satisfaction
- Quality
- Continuous improvement



THE FORGOTTEN PRINCIPLES BEHIND THE MANIFESTO

**HOW MANY OF THE
PRINCIPLES CAN YOU
REMEMBER?**

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



WHY DO WE FORGET?

THERE ARE MANY (12)



THEY ARE INFLEXIBLE

*BUSINESS PEOPLE AND DEVELOPERS MUST WORK
TOGETHER DAILY THROUGHOUT THE PROJECT*

THEY ARE OLD (2001)

DELIVER WORKING SOFTWARE FREQUENTLY, FROM A
COUPLE OF WEEKS TO A COUPLE OF MONTHS, WITH A
PREFERENCE TO THE SHORTER TIMESCALE

**THEY ARE 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

AGILE REQUIRES EARLY DELIVERY

It seems we need two things

1. A shorter and clearer definition to help us remember **what** Agile is
2. Deeper knowledge of the theory behind Agile principles to help us understand **why** they work

**MAYBE ACADEMIA CAN HELP
US FIND OUT **WHAT** AGILE IS?**

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



**DOESN'T QUITE WORK FOR
ME 😊**

What I Try to Remember Each Day

- 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 Valuing Simplicity**

DEFINITION OF AGILE



**BUT WHY IS IT SO
IMPORTANT THAT WE
UNDERSTAND WHY?**

LET US USE SCRUM AS AN EXAMPLE

Scrum Does It All

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

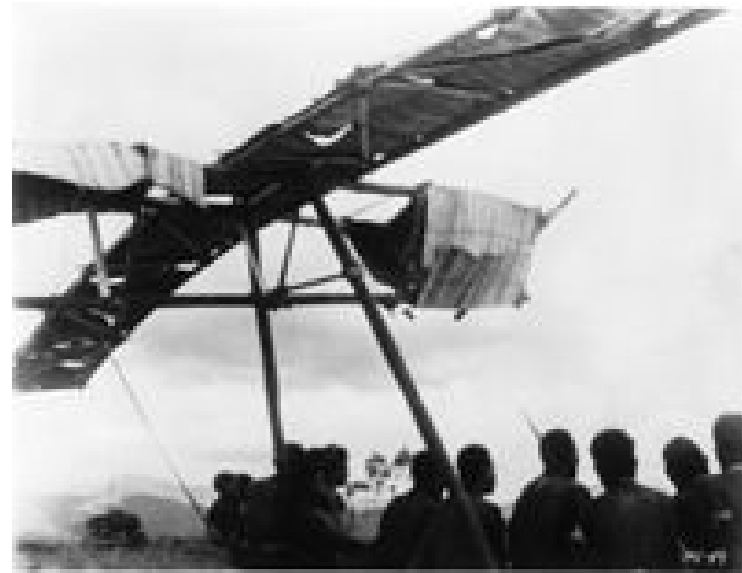
Yet!

- Many Scrum teams have historically proven to suffer from all or some of the following:
 - Unsustainable Pace
 - Poor Quality
 - Sub Optimization
 - Suboptimal Synchronization
 - Too Much Work In Progress
 - Plan Driven Focus

**NOT BECAUSE SCRUM IS A
BAD TOOL BUT BECAUSE WE
DO NOT UNDERSTAND WHY
SCRUM DOES OR DOES NOT
WORK IN OUR CONTEXT**

Too Avoid Cargo Cult Agile

- We Need to Understand the Theory Behind Agile Principles

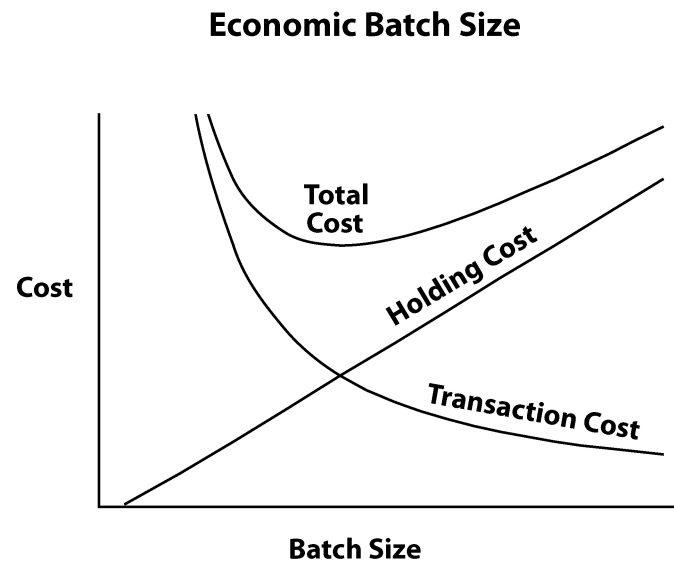


**LET US LOOK AT A FEW
ELEMENTS FROM LEAN
THEORY**

Identify value from the customer's perspective

- Put yourself in your customer's place
- Use value stream mapping to visualize your workflow
- Anything that is not a value adding activity is waste
 - When you don't know -“Go See”

- Batch size is a U-curve optimization, not faith based and context independent.



From "The Principles of Product Development Flow," by Donald G. Reinertsen.
Celeritas Publishing: 2009. Copyright 2009, Donald G. Reinertsen

But Toyota taught us that transaction costs are not fixed

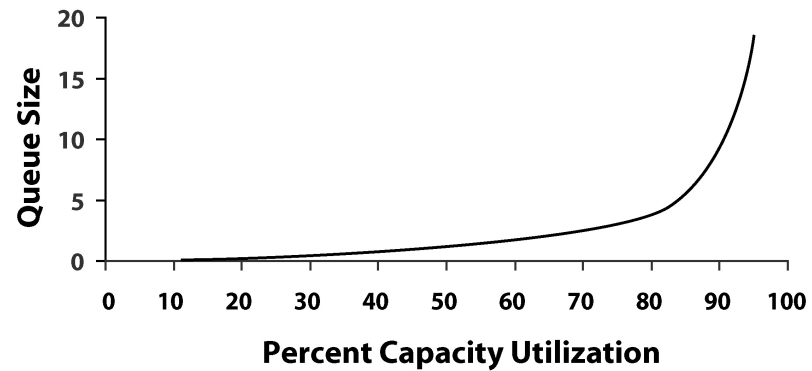
-
- Reduce WIP to improve flow and time to market
 - Reducing queues is far more effective than reducing variability

Don't stress the system to increase capacity utilization:

- It leads to
 - Unsustainable pace
 - Low quality
 - Increased queues
 - More breakdowns

- Maximum capacity utilization is a failure mode

Queue Size vs. Capacity Utilization



Note: Assumes M/M/1/∞ Queue

From "The Principles of Product Development Flow," by Donald G. Reinertsen.
Celeritas Publishing: 2009. Copyright 2009, Donald G. Reinertsen

Product Development = High Variability

- Product Development is not manufacturing!
- 0 Percent Variability = 0 Percent Value Creation
- Shorter iterations = higher variability
- Detailed plans WILL fail

Optimize the whole

- Enable pull across your entire value stream
- Development is not always the bottleneck and sub optimization will often stress the real bottleneck even more
- Focus on effective silos is batch optimization

Train for flexibility

- Flexibility is often a good investment in high variability environments like Product Development
- Flexibility means that you are able to sustain flow while handling uneven workloads (bursts)

-
- Not knowing the details is not necessarily a bad thing
 - We should always seek the optimal balance between expected payoff and probability

Choise	Cost	Payoff	Probability	Value
A	20.000	60.000	30%	12.000
B	30.000	60.000	50%	15.000
C	50.000	60.000	90%	9.000

Feedback

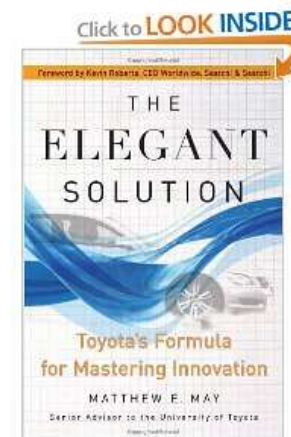
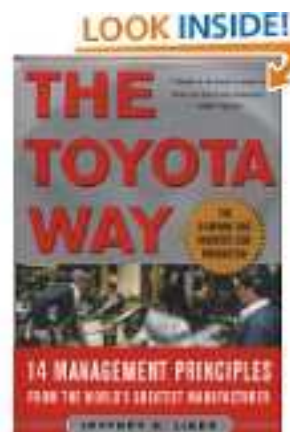
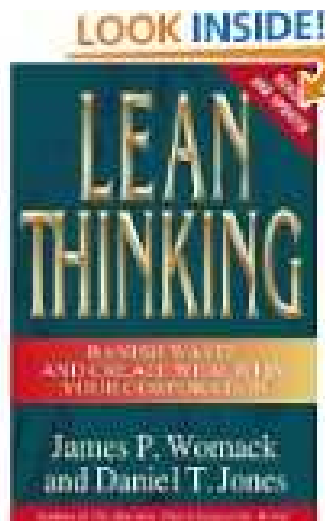
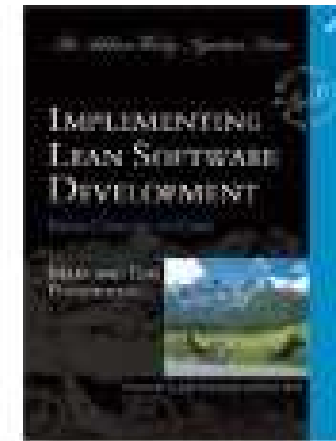
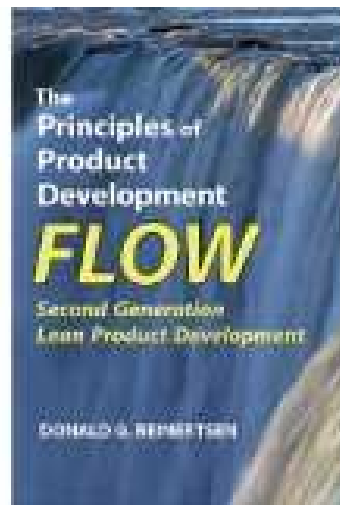
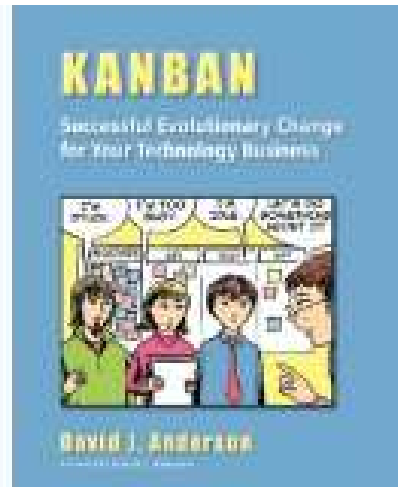
- Fast feedback cycles are the cheapest way to buy information

Visualize workflow

- Use Physical Cards and Visual Boards to Coordinate Work
 - Use visual controls to easily identify errors and inefficiencies in your activities and processes
 - Kanban Pull systems are a simple and easy method for scheduling that will outperform any automatic scheduling system

**GOTO; COPENHAGEN TO
LEARN MORE!**

Further Reading



Key takeaways

- 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 definition of Agile can be beneficial
- Lean theory can help us better understand Agile principles

What can Trifork offer?

- Scrum Master Certification
- Scrum Product Owner Certification
- 1 day Scrum Introduction for up to 15 people
- Scrum Kick-Starts
- Agile Reviews
- Kanban Training with David Anderson
- 1 Day Kanban Introduction for up to 15 people
- Lean in Product Development
- Experienced Agile Coaches
- Agile in Distributed Settings

Kanban tools

- Target Process
- Agile Zen
- Rally
- Leankitkanban
- Jira with Greenhopper plugin