

© Copyright Mountain Goat Software®

1



© Copyright Mountain Goat Software®

2

The OO-Agile connection

- Agile is the second wave of objects
- This is why so many Manifesto authors were early OOD / OOP advocates:
 - Kent Beck
 - Alistair Cockburn
 - Ward Cunningham
 - Bob Martin
 - Dave Thomas
 - etc.



© Copyright Mountain Goat Software

3

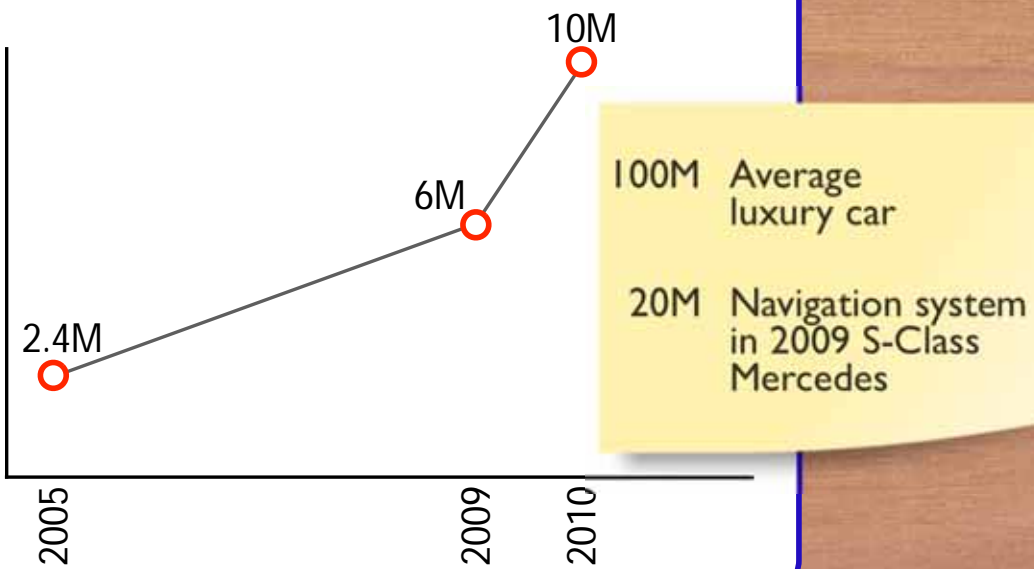
Looking back at OO

- Some said objects failed because:
 - They didn't achieve their full purpose
 - Prevented us from looking for even better ways to build software
 - Led businesses to expect miracles
- Yet: Could today's applications be written without objects?

© Copyright Mountain Goat Software

4

Lines of Code in Ford Vehicles



© Copyright Mountain Goat Software®

5

What expectations do we have of agile today that could lead us to look back on it and call it a failure?

1

Agile teams are 5–10x better.

2

There is a certain set of things you need to do or you aren't agile.

3

Becoming agile is easy.



© Copyright Mountain Goat Software®

6

Expectation #1: Agile is 5–10x better

- Very few teams have achieved the gains that some teams have shown are possible

- We can point to 5x and 10x success stories
 - These are the companies that did everything right
 - But these aren't the norm



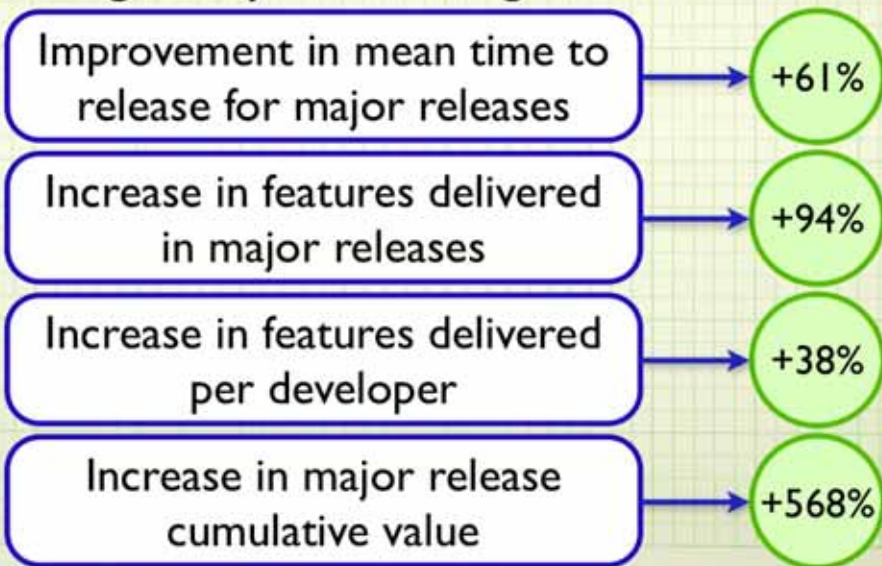
© Copyright Mountain Goat Software®

7

Salesforce.com



- During first year of using Scrum

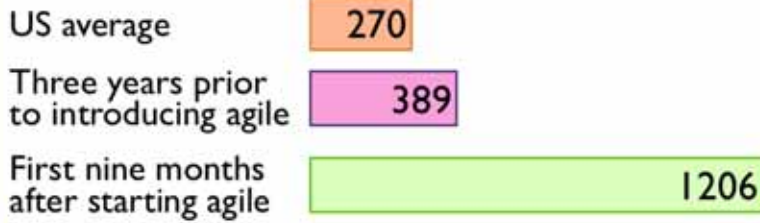


Source: Scrum Gathering, April 16, 2008.

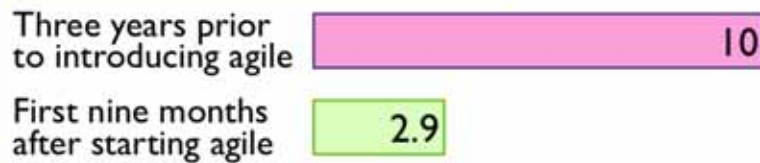
© Copyright Mountain Goat Software®

8

Productivity (NCSS / month)



Defects per KNCSS



Cosmodemonic Biotech

	Waterfall	Agile
Use Case pages	3,000	
User Stories		1,400
Calendar months	9	12
Person months	540	54
Lines of Java code	58,000	51,000
Lines of Java code per person-month	120	840



Improvements using XP

	Pre-XP	XP	Improvement
Cost	\$2.8 million	\$1.1 million	61%
Schedule	18 months	13.5 months	24%
Defects	2,270	381	83%
Staffing	18 months	11	39%

Source: "10 Tips for Successful Agile Transitions," Joshua Kerievsky, QCon 2007.
At <http://www.infoq.com/presentations/10-tips-for-agile-transitions>



© Copyright Mountain Goat Software®

11

Expectation #2: Agile is "this" set of things

- Many teams miss the mark
 - Either don't do "all" of agile or don't do it well
 - "We do Scrum, but..."
 - "we do 3-month sprints."
 - "we don't have anything shippable at the end of each sprint."
 - "we do testing in a separate sprint."



© Copyright Mountain Goat Software®

12

It's not about strict adherence

Inspect
and
Adapt



© Copyright Mountain Goat Software®

13

Expectation #3: Doing agile is easy

- “Mastery” in two days
- Most claims are realistic
 - “Although agile processes promise greater productivity once in place, productivity may decrease during the transition while the team learns new techniques.”¹
 - “Maybe we need to say, ‘Agile is hard, and you can’t master it by sitting through a two-day course.’”²
- But there are some promises of miracles:
 - “Current research shows that any team can achieve hyper-productivity in a few sprints, even in a dysfunctional company.”³

¹Cohn and Ford, “Introducing an Agile Process to an Organization,” *IEEE Software*, June 2003.

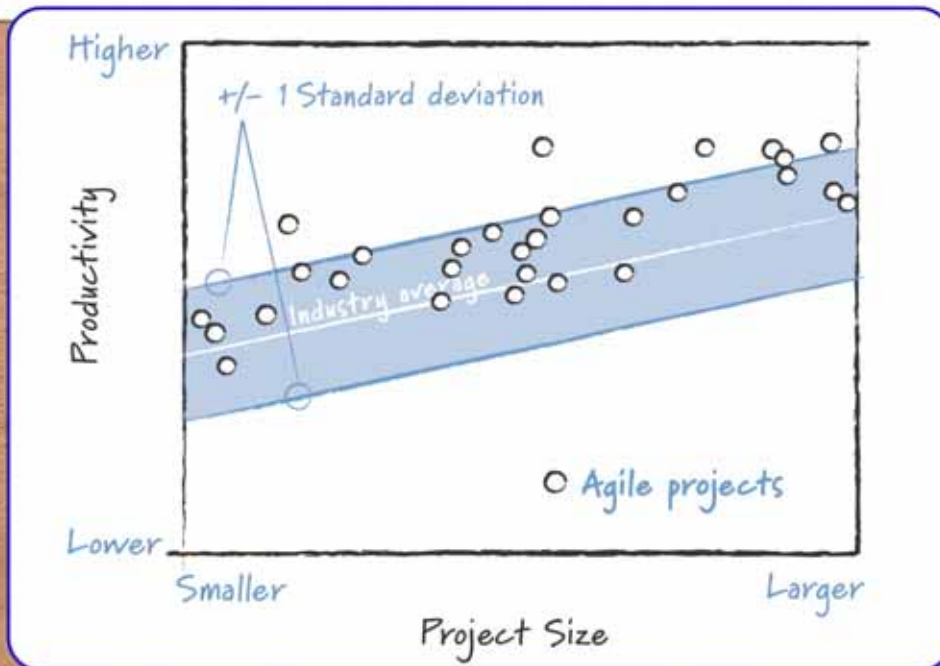
²James Shore, <http://jamesshore.com/Blog/The-Divide-and-Fall-of-Agile.html>

³Jeff Sutherland, <http://scrumjeffsutherland.blogspot.com/2009/12/jeff-sutherland-google-dec-14-2009.html>.

© Copyright Mountain Goat Software®

14

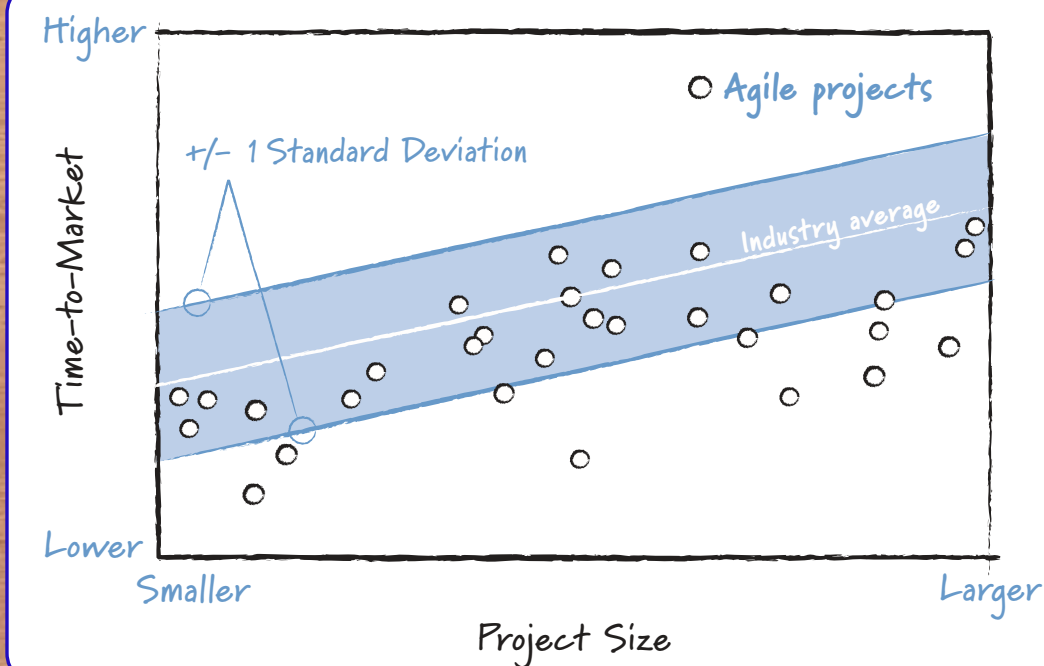
So let's see what the truth is



Mah, Michael. 2008. "How agile projects measure up, and what this means to you." Cutter Consortium

© Copyright Mountain Goat Software®

15



Mah, Michael. 2008. "How agile projects measure up, and what this means to you." Cutter Consortium

© Copyright Mountain Goat Software®

16

Will agile continue to “fail” in these ways?

- Of course, if that means
 - Failing to achieve full potential in all companies
 - Being done half-heartedly in some places
- But as an industry we’ll be far ahead of where we’d be without it



But these are invalid meanings of failure

- They pre-suppose agile as an end state
- To criticize a company as doing Scrum-but or Agile-but is to assume they’re done improving

Agile is not something you become,
it’s something you become more of.



Becoming more than you are



- I'm **more** flexible than I used to be
- You're **more** flexible than I am
- She **is** flexible



© Copyright Mountain Goat Software®

19

Succeeding with agile

- Success in agile is about becoming better than we were
- Stop obsessing over being perfect



© Copyright Mountain Goat Software®

20

How do we get there?

- We raise the bar on one another



© Copyright Mountain Goat Software®

21

Raising the bar—slowly but steadily



© Copyright Mountain Goat Software®

22



© Copyright Mountain Goat Software®

23

Continuing to raise the bar on one another



© Copyright Mountain Goat Software®

24



A

Awareness that there is room for improvement

D

Desire to change

A

Ability to work in an agile manner

P

Promote early successes to build momentum and get others to follow

T

Transfer the impact of agile throughout the organization so that it sticks



© Copyright Mountain Goat Software®

27

"I'm the ScrumMaster and ...

...the developers are not meeting expectations for code quality.

One of our challenges is that we're still hacking our way through lots of legacy code that isn't unit-testable or automated yet. This code is mission-critical and the person who has been working mostly on that area of code consistently leaves holes in the design and implementation of new pieces of that code.

How can I get her to write high-quality code with automated tests?"



© Copyright Mountain Goat Software®

28

Individual and group change

- All individuals will need to move through the Awareness, Desire, and Ability stage
 - But will do so at different rates
- Early adapters and leaders:
 - Use the Promote stage to build Awareness and Desire in later adopters
 - Need to Transfer the impact of agile to groups like Human Resources or the transition will fail



Tools for building...

Awareness

- Communicate that there's a problem
- Use metrics
- Provide exposure to new people and experiences
- Focus attention on the most important reason or two for changing



Tools for building...

Desire

- Communicate that there's a better way
- Create a sense of urgency
- Build momentum
- Get the team to take agile for a test drive
- Align incentives (or, at least, remove disincentives)
- Focus on addressing any fears
- Help people let go
- Don't discredit the past
- Engage everyone in the transition



Waterfallacies

- Mistaken beliefs or ideas about agile created from working too long on waterfall projects.

Examples

- Agile ignores architecture, which would be disastrous for the type of system we build.
- Agile is OK for simple websites, but our system is too complicated.
- Our team is spread around the world, and agile requires face-to-face communication.



Ability

- Provide coaching and training
- Hold individuals accountable
- Share information
- Set reasonable targets
- Just do it

Promote

- Publicize success stories
- Host an agile safari
- Attract attention



Transfer

- Transfer the effects of agile beyond the current group
 - A team transfers to its department
 - A department transfers to its division
 - etc.
- If you don't transfer, the transition will eventually and inevitably fail
 - Too much **organizational gravity** pulling us back toward the status quo
- Example:
 - If you don't align promotions, raises, annual reviews, those will work against you



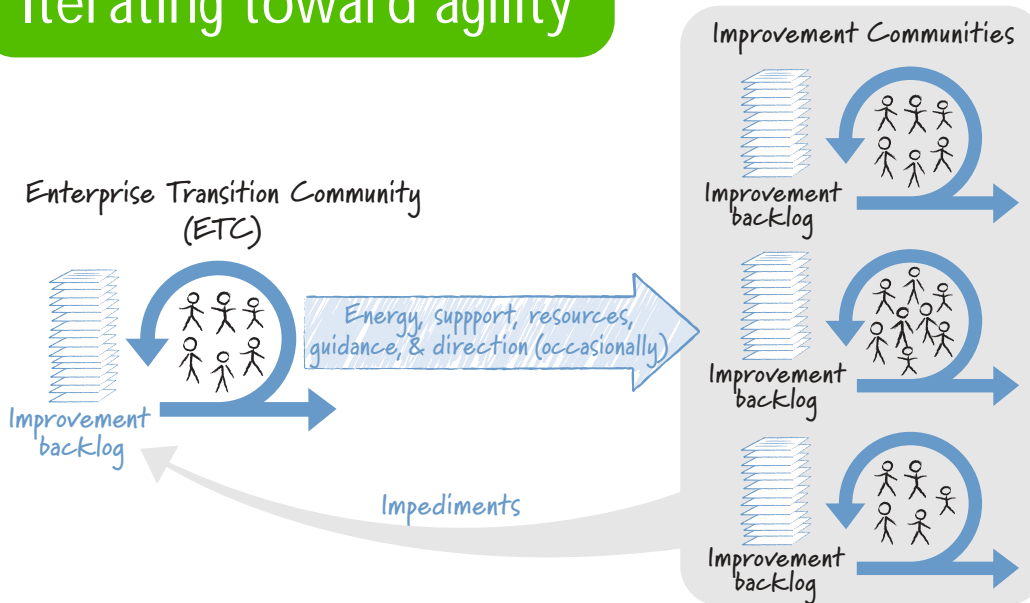
Iterating toward agility



© Copyright Mountain Goat Software®

35

Iterating toward agility



© Copyright Mountain Goat Software®

36

Enterprise Transition Community (ETC)

- Creates a culture in which passion and desire to improve can thrive
- Does not direct the transition effort
 - Provides energy, resources, support and guidance
 - Removes organizational impediments to agility
- Encourages Improvement Communities to form



ETC members

- Sponsor
 - From highest level at which change is supported
 - Not a checkbook-only commitment
- Others
 - From any level but driven by desire to improve
- Disbands when the “transition” part of adopting agile is over



ETC responsibilities

- Articulate the reasons for adopting agile
- Stimulate conversation
- Provide resources
- Engage everyone
- Set appropriate aspirations
- Anticipate and address people issues and other impediments
- Encourage simultaneous focus on practices and principles



An ETC's improvement backlog

Item	Who	Note
Create an "Agile Office" where teams can get help.		Jim (CTO) to talk this up at monthly development meeting. Let's see if there's any interest.
Establish an internal program for developing ScrumMasters.		How do we identify good internal candidates? How do we develop them?
Collect and disseminate Scrum success stories in our company.	SC	Savannah has expressed interest in this.
Resolve dispute with facilities over rearranging second floor cubicles.	JS	Jim to talk to Ursula in facilities
Get more teams to do continuous integration.	AR	Arie will summarize metrics from his project and see how many teams he can motivate.



Improvement communities (ICs)

- Form around the passion of a small number of people
 - Expand from there
- Do the real work of improving how the organization implements agile
- Focus on goals with practical relevance
- Examples:
 - ScrumMaster, Testing, Product Owner, Continuous Integration



Working with an IC

- An IC works with a project team
- Work is not done in an ivory tower
- Most ICs work in 2–4-week iterations
- Disband or refocus when goal has been achieved



ETC Improvement Backlog

...
Establish an internal program for developing ScrumMasters.
...

Not everything on an IC's improvement backlog needs to tie back to the ETC's backlog

An IC Improvement Backlog

Figure out how to identify good candidates to become ScrumMasters (in addition to those who ask to participate in this program).

Establish an internal mentoring program.

Develop some internal classroom training. Which courses? Who can teach them? Can we license courses?

Get budget for next year for external coaching. How many days? At what expected daily rate?



1

Go back to the office and create an improvement backlog

2

Create communities or not but find passionate people to attack improvement backlog items

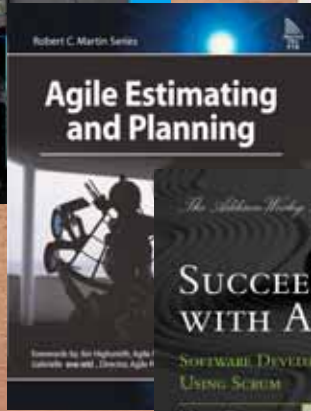
3

Return here next year more agile than you are today

4

Let's see how soon we can make it "software development" instead of "Agile Software Development"





Mike Cohn
mike@mountaingoatsoftware.com
www.mountaingoatsoftware.com
twitter: mikewcohn
(888) 61-AGILE



© Copyright Mountain Goat Software®