



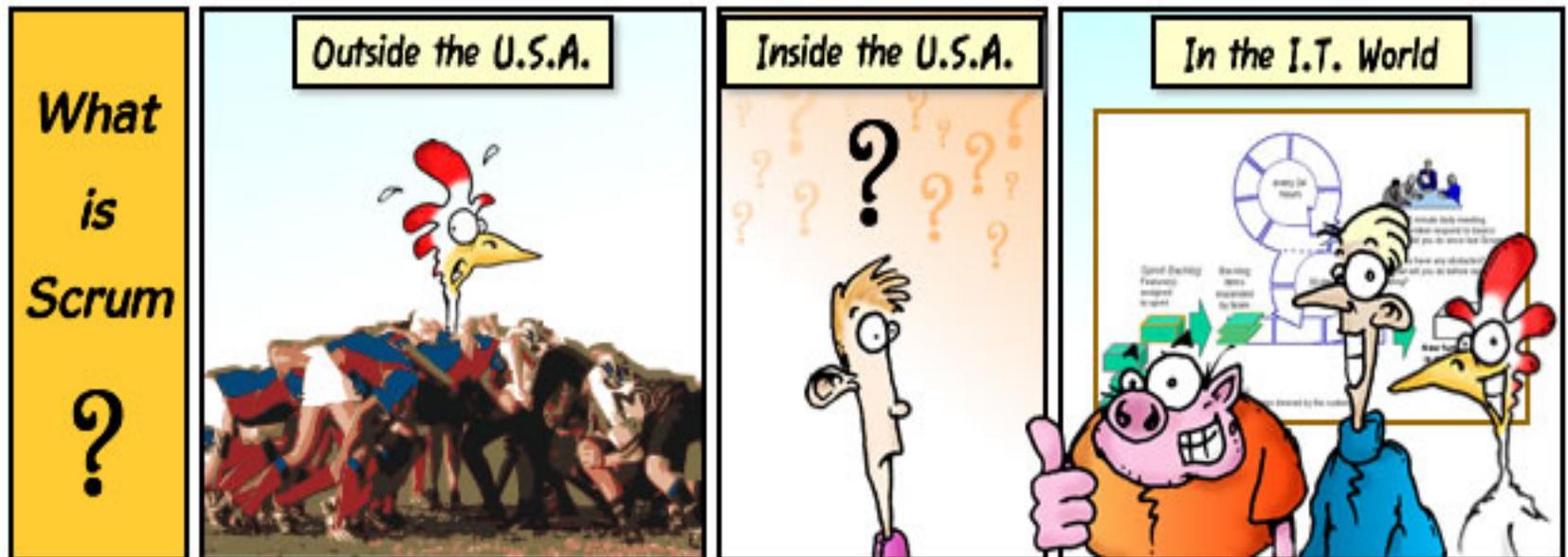
Helping Pigs Fly

**Stories, Experiences, and
Lessons of Scrum In The Large**

Tom Mellor and Lowell Lindstrom

Beginnings...

Expect to get muddy...and have fun!



By Clark & Vizdos

© 2006 implementingscrum.com

All cartoons copyright by Mike Vizdos *Implementing Scrum* www.implementingscrum.com Used with permission

A Bit About Tom



- Certified Scrum Trainer teaching Scrum inside and outside his company
- Started using Scrum in 2004 – without proclamation or fanfare (and rather secretly 😊)
- Ken S – “Read the book that Beedle and I wrote, apply the practices and let me know how it goes.”
- Has now worked on 13 Scrum projects and helped and coached many others
- Still a work in process. Overall, very fulfilling and rewarding, but has had it’s personal detriments

A Bit About Lowell



- Founder of The Oobeya Group and 21 year software development industry veteran
- A Certified Scrum Trainer – taught and coached agile software development since 1999
- Chairman for Agile 2001 and co-Programming Chairman for Agile 2004
- Founding Board member of the Agile Project Leadership Network (www.apln.org)

Beginnings...Pre-Scrum 2003

- Came to IT as a BA in 2001 and became a project manager in March 2003 after spending 20 years in P/C claims
- “Gee, this way of working seems inordinate. It expects that you can predict everything up front and discourages adapting to almost certain change.”
- “You are accountable for delivery of all of the requirements to an acceptable state of quality within the specified time and budget...But I have little discretionary control over those things...Get over it.”

Beginnings...Little Project 2004

- “I think I know a way we can deliver this in 60 days, but it is completely different from how we normally do work.”
- Your risk isn't failure...it's success.”
- “Wow, this feels good and it works!”
- “Can you do Scum on a big project?”

Beginnings...Big Project 2004

- “We’ll do Scrum if you keep your 3 promises”
 - *You will really protect us*
 - *You really get noise out of our way*
 - *You won’t abandon us if things go badly*
- “What are you doing? We are building a methodology that will solve our problems...”
- You’re the “real” product owner, why don’t you take on the role?”
- “I am not the PM, I am the ScrumMaster...Then who is responsible for the success?”

Beginnings...Big Project 2004

- “We aren’t going to make the date...but we have an alternate temporary solution”
- “That was fast and we even got what we wanted!”
- “We have more time and you have money left – do you want more developed?...You can do that??”
- “Why don’t we give the team a sum of money and let them divide it?...That will never work.”

Lessons in the Beginning...

- Tom's "aha moment" – the Product Owner of Big Project says "Wow, this is great" at a Review
- Early adoption was bottom up. Now, it's bottom up and top down – leverage both
- When it's bottom up, people risk careers and being villanized and ostracized – be careful
- When it comes top down, it risks being adulterated and politicized
- Either way, it's the front line that does the work and they have to buy in and be protected

Lessons in the Beginning...

- The political environment can't be avoided. Be respectful, sensitive, and empathetic – that's being political. Find allies and work with skeptics
- People have their own agendas and underlying motivations. They can appear supportive and may actually be subversive.
- Learn to have *crucial conversations*. If someone is disingenuous, have a crucial talk with them
- Keep the organization and teams in the focus and in front, not individuals

Lessons in the Beginning...

- Understand and appreciate the organization's limitations and ability to consume change
- Pull the dragon along, but always keep your eyes on its head. It's fire can fry you instantly!
- Be transparent – admit challenges and failures and investigate why
- It will be easy to blame Scrum – be pragmatic and rational, not defensive

Lessons in the Beginning...

- Promote the benefits always:
 - Fast delivery with quality built in
 - Customer satisfaction and involvement
 - Team satisfaction and productivity
 - Efficiency and reduction of waste
- Understand the concerns

The 07 *Pure Scrum* Experiment

- 10,000 hour project originally budgeted to 1.3 million dollars
- Idea was to see what would happen with a project that was exempt from “traditional project controls” and used Scrum “without any modifications”
- No Quality Control checks
- Team of 8 worked cross-functionally, with a couple of people working off-shore

The 07 *Pure Scrum* Experiment

- I'll give you back \$500,000 right now at the start of the project
- "I'm going to become a welder."
- "It's nice that we can actually be trusted to figure things out."
- "Were the customers satisfied? From Day 1 through now 30 days post production, not a single complaint or defect."

The 07 *Pure Scrum* Experiment

- Results:
 - A highly energized and productive team that delivered a high quality product
 - Project came in at 2/3 of the budget and in the allotted time
 - Started middle management thinking hard about the validity of the agile approach

Current Projects – Good and Bad

- “They got nothing *done* in Sprint 2, but patted themselves on the back anyway.”
- “The team lost the velocity bet – they needed to do 80 points per Sprint to deliver the functionality demanded by the Product Owner. Their capacity was 40 points. The only thing they really gained was technical debt.
- “When are we going to get this tested?...When we can.”
- “Scrum didn’t fail them. They failed Scrum. That’s what happens when you do *Scrum, but.*”

Current Projects – Good and Bad

- “After you complete the build, you have no idea of the integrity of your code, do you?...Not really.”
- The stakeholders are demanding that we produce something for the Sprint Review. It’s ok for us to show something even if it is full of defects – they don’t need to know it is full of defects.”
- “You have a work flow problem, capacity problem, and velocity problem. Other than that you are doing fine.”

Current Projects – Good and Bad

- “The PM saw no benefit to the retrospectives, so he cancelled them.”
- “I’m the PM, this team will do what I want it to do, when I want it done, and how I want it done.”
- “I’m not too enthusiastic about using Scrum. I have to work harder [than with traditional methods] and I don’t get paid anymore to do that.”
- “When the truth hurts, people don’t want to hear it.”

Current Projects – Good and Bad

- “These 2 projects did it right. They held to the values, principles and practices of Scrum and agile. No short changing and they dealt with impediments honestly and effectively.”
- “The team effectively managed the Product Backlog and delivered a amount considered *good enough* to the Product Owner in 7 months rather than 12 months.”
- The team had an abnormal Sprint termination, regrouped and delivered fast. That floored us – we have never seen that happen.”

Other Humorous Quotes

- “Iterative development means the development is iterative, not the requirements”
- “Acceptance criteria: as a non-agent user, I want the ability to not see the work item icon on the screen.”
- “He’s a Scrum guru!!...Really? Has he ever actually used Scrum on a project? No, but he’s read a lot about it.”
- “What is the minimum passing percentage for the Nokia test...60%?”

Helping People Learn...and Do

- First, make sure they understand Shu Ha Ri:
*Learn to do agile correctly and effectively.
Then, master it. Then, maybe, do as you will.
Many people want to ignore steps one and two and jump to step three.*
- If they ignore Shu Ha Ri, see if they can respect Yoda: *Do, or do not. There is no try.*

Helping People Learn...and Do

- You cannot expect to change the overall structure of the organization – at least not without lots of pain and anxieties
- The best is to work within the structure and advocate for a long term strategy. This will force the organization to face up to whether it has the fortitude and chutzpah to really commit to the change
- Support the effort with analysis and conclusions from other large organizations

Helping People Learn...and Do

- Large organizations have their own complex eco systems and they value differentiation and uniqueness. You have to be sensitive to uniqueness, but leverage common traits and experiences
- In highly regulated industries, anti-trust and competitive issues often impede the sharing of information between companies in the same industry
- Gather information and results that you can and take them with a grain of salt

Helping People Learn...and Do

- Lessons:
 - Solidly support the foundation of the adoption with an effective coaching organization
 - Bring in people to coach who have done it in organizations of similar size (if possible)
 - Work with internal people to become coaches, but be sure they have actual experience doing Scrum and agile (or get the experience)
 - Experience means actually working on a Scrum project

Helping People Learn...and Do

- Lessons:
 - Understand that *Scrum, but* will happen and that you cannot turn a large organization into an agile organization overnight – it is a journey that will take *years*
 - Discourage teams from compromising on the values, principles and practices; they will be inclined to do so if they find doing Scrum hard (and they will)
 - Set expectations as to what a Scrum project is – e.g. the Nokia test, etc.

Helping People Learn...and Do

- Lessons:
 - Expect to deal with a full range of individual reactions to Scrum – from full embracement and enthusiasm to complete resistance and even subterfuge
 - Divide your focus to 2 levels: the team level and the organization level. Find people who are adept at coaching at each level and leverage their expertise accordingly.

Helping People Learn...and Do

- Lessons:
 - Balance dogma and pragmatism. Being uncompromisingly dogmatic can push the team and the organization to delusional and dysfunctional behavior.
 - Some (many??) organizational impediments and problems will not be overcome instantly (or perhaps even in the mid-term.) Teams will need to be given the latitude of divergence to deal with them.

Helping People Learn...and Do

- Lessons:
 - While there should be tolerance for divergence from strict adherence to practice, there must still be boundaries. This is essentially the tolerable limit of pragmatism before the behavior or action becomes *non-agile*
 - Be careful of opacity and delusion – make sure that bad news isn't hidden or figmented

What About Culture??

- “It’s the culture, stupid...and believing you can change it is simply naïve.”
- “You don’t change culture. You prod it, influence it, and hope you can mitigate it’s inherent flaws.”
- “You want to change the culture??? That’s easy: just change people’s norms, behaviors, beliefs, and attitudes. It’s that simple.”

What About Culture??

- “I want accelerators; we can simply select and use those things that can make us faster...If you pick and choose practices, and dismiss the whole of it, you will negate the symbiotic benefits.”
- “We set up an entire testing organization around traditional development. Agile will require a complete change in that structure and I don’t think we can do that or are up to it.”
- “You are endangering yourself and PMs as a whole. And that is personally threatening to me. You ought to stop.”

Lessons from Experiences

- If the organization is up for it, it can do Scrum and agile. Politics will be part of the equation.
- It will surface the organizational inadequacies – engineering, process, structure, etc. Those will embarrass people and make those people uncomfortable and even angry.
- People can feel threatened, especially in organizations where people's value is seen from a role perspective where the role is narrowly defined and utilized

Lessons from Experiences

- Common Issues:
 - Organizational work structure and roles
 - Processes created to support a waterfall methodology
 - Traditional measures and metrics
 - Testing in the agile space – especially automation
 - Adoption of agile development practices – TDD, etc.

Lessons from Experiences

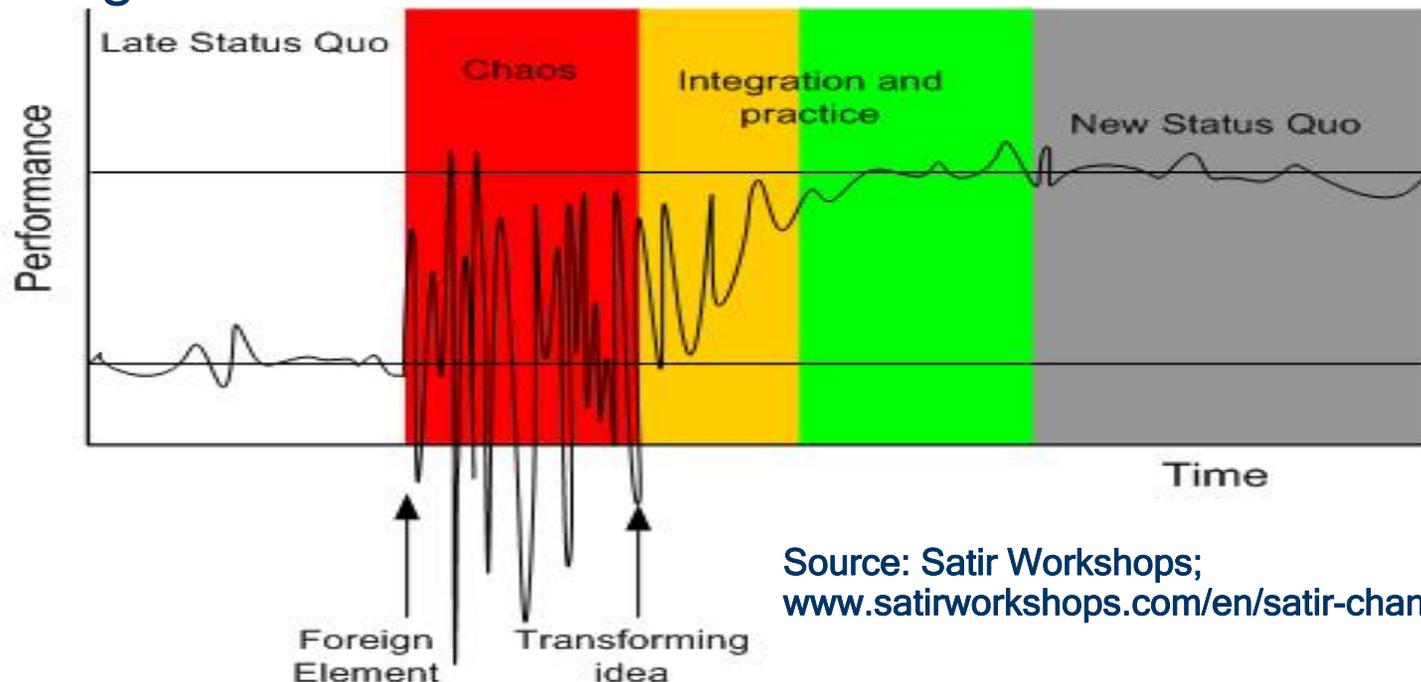
- Common Issues:
 - Tradition of command and control management versus servant leadership
 - Doubt and discomfort about self-organization and self-direction
 - Physical environment and the lack of collaborative space
 - Myths, legends, and innuendo...and basic lack of understanding and misunderstanding of Scrum and agile

Lessons from Experiences

- Common Issues
 - Individual performance measurement and rewards systems
 - Inability within the culture to have *crucial conversations*
 - The practice of *accountabalism* – the assumption that systems are perfect and deciding that problems that people create can be solved by specifying another level of detail and/or with another addition of a role

Lessons from Experiences

- There will be “aha moments” that indicate realization of the *Transforming Idea* – at individual, team and organizational levels



Source: Satir Workshops;
www.satirworkshops.com/en/satir-change-model

Endings...”We need to throw the pendulum out and continuously improve”

- Eventually, improvement will happen...or not. It will only happen if the organization really commits to and understands continuous improvement (especially in its practices and processes)
- Customers’ satisfaction will drive the wider adoption – they will demand it
- Expect to continue to deal with pointy-haired things; they will continue to exist and the absurd will not disappear
- Expect to see good things and improvements along with persistent challenges