AGILE DEVELOPMENT IN THE ENTERPRISE

“Twice as much for half the price!”

Jeff Sutherland, Ph.D.

- Chairman, Scrum Training Institute
- CEO Scrum, Inc. and Senior Advisor, OpenView Venture Partners
  - Agile coach for OpenView portfolio companies
  - CTO/VP Engineering for 9 software companies
  - Created first Scrum at Easel Corp. in 1993. Rolled out Scrum in next 5 companies
  - Achieved hyperproductive state in all companies
  - Signatory of Agile Manifesto and founder of Agile Alliance

- http://jeffsutherland.com/scrum
- jeff.sutherland@scruminc.com
Techniques or Methodologies Used

Source: Forrester Research December 2008
Global Agile Company Online Survey

Base: 241 technology industry professionals in a variety of roles, including but not limited to development (numbers have been rounded)
Openview Venture Partners

- We invest in organizations deploying Scrum
  - one hyperproductive company out of 10 might meet investment goals for a venture group
  - two or more hyperproductive could change investment practice

- We invest in market leading, industry standard processes
  - this means Scrum and XP

- We insure the entire company implements basic Scrum practices
  - Teams pass the Nokia test
  - Management is held accountable at Board level for removing impediments
  - Maturity level assessment for management, product marketing, and development organization
Double output and cut workload in half

Maxwell Curve

Scrum

Waterfall

Story Points

Work Week

www.openviewventurepartners.com

© Jeff Sutherland 1993-2008
Scrum is a Simple Framework

- Meeting
  - Sprint Planning
  - Daily Meeting
  - Sprint Review

- Roles
  - ScrumMaster
  - Product Owner
  - Team

- Artifacts
  - Burndown Charts
  - Sprint Backlog
  - Product Backlog

- Meetings
  - Daily Meeting
  - Sprint Review

© Jeff Sutherland 1993-2008
Scrum Dynamic Model

Value

Sprint

DONE

IMPEDIMENTS

READY

Daily Meeting

Velocity
Apply Scrum Now - A 12 step program

1. Agree on a PO, SM, and full Team. And on a Product goal.
2. Set a date now for the Sprint Review in 2 weeks and send out invites.
3. Review/define a ranked Product Backlog of features
4. Estimate the Product Backlog items
5. Conduct Sprint Planning with Team and Stakeholders. Complete Sprint Backlog
6. Commit as a team to the Sprint
7. Track status and obstacles daily via the Daily Scrum
8. Track progress using the Sprint Burndown
9. Conduct a Sprint Review; demo done items
10. Conduct a team Retrospective
11. Take action on top impediment
12. GOTO 2

Source: Hubert Smits & Jean Tabaka
Key Scrum Success Factors

- Done means acceptance tests pass (testers on team)
- Product backlog ready (remove if not ready)
  - Sized and prioritized
  - Estimated in story points
- Scrum Board
  - Burn down in priority order, test immediately
- Burn Down Chart
  - Track Done
  - Execute Emergency Procedure when needed
- Know velocity
- Remove top priority impediments
- Prevent disruption of team
  - No multitasking
  - Minimize team member change
- PO/SM/Team trained properly
Avoiding ScrumBut - Nokia Test Origins
Nokia Siemens Networks

In 2005, Bas Vodde started training and coaching teams at Nokia Networks in Finland. The first Nokia test focused on Agile practices

jeffsutherland.com/scrum/basvodde2006_nokia_agile.pdf

By 2007, Siemens joined Nokia Networks to form Nokia Siemens Networks with over 60,000 employees and 15 billion Euro in revenue. Bas Vodde moved to Nokia China to and updated the Nokia Test to include Scrum practices.

In 2007, Jeff Sutherland tuned the Nokia Test for Scrum Certification and in 2008 developed a scoring system

agileconsortium.blogspot.com/2007/12/nokia-test.html
jeffsutherland.com/scrum/Agile2008MoneyforNothing.pdf

Each person on the team takes a sheet of paper and prepares to score questions on a scale of 1-10.
Question 1 - Iterations

Does the team have fixed length iterations that are four weeks or less?

Success rate of projects in 1994 - 14%
  before iterative development
Success rate of projects in 2004 - 34%
  after iterative development

Source: Standish Group
Question 2 - Testing

- Is there working software at the end of an iteration?

- Velocity increases by 100% and defects decrease by 40% with working software
Question 3 - Agile Specification

Does the team have good user stories?

Time to deliver is directly proportional to length of the specification

Rework is generated if:

- The specification has implementation details
- The specification is not clear to developers
- The specification does not include acceptance tests
Question 4 - Product Owner

Does the team have a good Product Owner?
- Vision, roadmap, product backlog ready
- Knows velocity of the team
- Works with stakeholders and works with team

Good Product Owner increases velocity by 100%

Bad Product Owner can reduce velocity to zero
Question 5 - Product Backlog

Does the team have a good Product Backlog

- Good user stories
- Clear to team
- Estimated by team
- Prioritized by product owner
- Sized properly
- Acceptance tests

Good Product Backlog will double velocity of team
Question 6 - Estimates

Is the Product Backlog estimated by the team in story points using Planning Poker

Product Owner and Planning Poker cut costs of project planning by 80%
Question 7 - Sprint Burndown Chart

Does the team have a Sprint Burndown chart that clearly shows the status of the Sprint?

Sprint success rate will double with careful monitoring of burndown
Question 8 - Team Disruption

Does the team have only Scrum roles with no disruption of the team by managers, project leaders or Product Owner?

Few roles and good daily meetings can increase velocity by 5000%

Organizational Patterns of Agile Software Development by Coplien and Harrison (2004)
Question 9 - Team

Does the team chose their own work to maximize velocity?

- Team helps each other choose the right work
- Team helps each other to implement work
- Team does highest priority story first together and tests immediately
- Team commits to Sprint goal together and aggressively removes impediments

Self-organization is the key to hyperproductivity - 400-800% improvement in velocity and quality
How we invented Scrum:
Learning about innovation from Xerox Parc

Personal Workstation

Mouse (SRI)

Ethernet

Windows Interface

Laser Printer

Smalltalk

© Jeff Sutherland 1993-2009
Alan Kay’s Innovation Strategy

- Incremental - No
- Cross Discipline - Nyet
- Out of the Box - Yes
Out of the Box

Scrum looked at projects that were off the plate
  - IBM surgical team
  - Takeuchi and Nonaka
  - Borland Quattro Project

Scrum: A Pattern Language for Hyperproductive Software Development

Going from good to great means Toyota or better.
I find that the vast majority of organizations are still trying to do too much stuff, and thus find themselves thrashing. The only organization I know of which has really solved this is PatientKeeper.  Mary Poppendieck
What’s happening with Scrum?
Multiple Team Scrum

Diagram showing teams and stakeholders linked to operations.
Scrum in Transition
Delivering to End Users
How do you scale Scrum to thousands of developers?

- Step by step
- Training and coaching is critical
  - A internal trainer at Yahoo can train, launch, and coach about 10 new teams a year
  - Teams that are not coached do not do so well. Average increase in productivity is 35% company wide.
  - Coached teams get 300-400% improvement.
- Yahoo launched over 200 teams in three years in Silicon valley where they have 2000 developers.
Rate Scrum relative to how the team was building products previously:

Business value of what the team produced in 30 days?

2%  64%

Scrum MUCH WORSE  Scrum WORSE  Scrum about the same  Scrum BETTER  Scrum MUCH BETTER
Yahoo Return on Investment

- Each Scrum Trainer starts up and coaches 10 new Scrum teams a year
- Coached velocity increase is 200-400%
- Uncoached average increase is 35%
- Conservative cost reduction per trainer is over $1M/yr

Lean Thinking Tools

- Systematic Software Engineering used the tools from Lean Software Development to develop their Scrum implementation.
- Analyzing dependencies, they produced a strategy for ordering the implementation of Lean.
Thinking tools are best transformed by people and projects.
Published experiences with "rework"

Source: Krasner & Houston, CrossTalk, Nov 1998
Diaz & King, CrossTalk, Mar 2002
CMMI/SCRUM Performance analysis

Source: Systematic Software engineering 2006
Systematic CMMI 5 Analysis
First six months of Scrum

- 80% reduction in planning cost
- 40% reduction in defects
- 50% reduction in rework
- 100% increase in overall productivity
- Estimation error < 10%
- Project completion on time > 95%
- Waterfall projects (required by some defense and healthcare contracts) are now contracted for twice the cost of Scrum projects (and produce lower quality).
Systematic is going from “beginners Scrum” to

- First doubling of velocity comes from software DONE at the end of the sprint.
- Second doubling come from product backlog READY at the beginning of the sprint.
- Systematic now has several teams executing the second doubling model successfully
- Will role this out to whole company

Carsten Jakobsen and Jeff Sutherland. Scrum and CMMI - Going from Good to Great: are you ready-ready to be done-done? Agile 2009, Chicago.
Case Study: Scrum and XP

- The first Scrum used all the XP engineering practices and set-based concurrent engineering.
- Most high performance teams use Scrum and XP together.
- It is hard to get a Scrum with extreme velocity without XP engineering practices.
- You cannot scale XP without Scrum.
Distributed/Outsourcing Styles

Isolated Scrums

Distributed Scrum of Scrums

Totally Integrated Scrums
Outsourcing

What happens if you outsource $2M of development?

- Industry data show 20% cost savings on average

Outsourcing from PatientKeeper to Indian waterfall team:

- Two years of data showed breakeven point occurs when Indian developer costs 10% of American Scrum developer
- Actual Indian cost is 30%

$2M of Scrum development at my company costs $6M when outsourced to waterfall teams

Never outsource to waterfall teams. Only outsource to Scrum teams.
SirsiDynix - Anatomy of a “failed” project

- Over a million lines of Java code
SirsiDynix Distributed Scrum

56 developers distributed across sites

PO

SM
Dev
Dev
Dev

T Ld
Dev
Dev
Dev

Catalogue

Serials

Circulation

Search

Reporting

SirsiDynix
Provo, Utah
Denver, CO
Waterloo, Canada

Exigen Services
St. Petersburg, Russia
SirsiDynix Distributed Scrum

Scrum daily meetings

- Local Team Meeting
- St. Petersburg, Russia 17:45pm
- 7:45am Provo, Utah
- Scrum Team Meeting

Monday, August 24, 2009
SirsiDynix Distributed Scrum
# Velocity in Function Points/Dev month

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Person Months</td>
<td>54</td>
<td>540</td>
<td>827</td>
</tr>
<tr>
<td>Lines of Java</td>
<td>51,000</td>
<td>58,000</td>
<td>671,688</td>
</tr>
<tr>
<td>Function Points</td>
<td>959</td>
<td>900</td>
<td>12673</td>
</tr>
<tr>
<td>Function Points per Dev/Mon</td>
<td>17.8</td>
<td>2.0</td>
<td>15.3</td>
</tr>
</tbody>
</table>

1. M. Cohn, User Stories Applied for Agile Development. Addison-Wesley, 2004
SirsiDynix Challenges

- ScrumBut
- Builds were stable only at Sprint boundaries
- No XP in U.S, only in Russia, did not have equal talent across teams
- No face to face meetings
- Low test coverage
- Poor refactoring practice
- Company merger created competitive products
Russian projects velocity data suggests high velocity is not an accident

Exigen Services Recent C# Projects

Function Points/Dev-Month vs. Lines of Code (LOC)

Industry Average = 2

© Jeff Sutherland 1993-2009
Setting up a prospective study

- Define the distributed team model before projects start
- Assure consistent talent, tools, process, and organization across geographies
- Establish high quality data gathering techniques on velocity, quality, cost and environmental factors.
- Run a consistent team model on a series of projects and look for comparable results
- Demonstrate that local velocity = distributed velocity
- Demonstrate that local quality = distributed quality
- Demonstrate linear scaling at constant velocity per developer
Case study: Building a new railway information system
ProRail PUB Example

- ProRail rescued a failed waterfall project to build a new scheduling system and automated railway station signs at all Netherlands railway stations.
- An 8 person Scrum team started the project and established local velocity (half Dutch, half Indian).
- After establishing local velocity at 5 times other waterfall vendors on the project, the Indian half of the team went back to India.
Scaling Fully Distributed Scrum
ProRail Defect Tracking

- Defect rate gets lower and lower as code base increases in size
- 95% of defects found inside iteration are eliminated before the end of the iteration
Team Characteristics

- TDD, pair programming, continuous integration. Same tools and techniques onshore and offshore.
- Daily Scrum meeting of team across geographies.
- SmartBoards, wikis, and other tools used to enhance communication.
- Indians say it feels exactly the same in India as it does in Amsterdam. They do the same thing in the same way.
# Dutch Velocity vs. Russian Velocity

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Person Months</td>
<td>827</td>
<td>125</td>
</tr>
<tr>
<td>Lines of Java</td>
<td>671,688</td>
<td>100,000</td>
</tr>
<tr>
<td>Function Points</td>
<td>12673</td>
<td>1887</td>
</tr>
<tr>
<td>Function Points per Dev/Mon</td>
<td>15.3</td>
<td>15.1</td>
</tr>
</tbody>
</table>

1. M. Cohn, User Stories Applied for Agile Development. Addison-Wesley, 2004

© Jeff Sutherland 1993-2009
Linear Scalability of Large Scrum Projects

Linear scalability

Hours/Storypoint

© Jeff Sutherland 1993-2009

Monday, August 24, 2009
Xebia’s Conclusions

- Fully Distributed Scrum has the full benefits of both local hyperproductive teams and offshoring.
- Fully Distributed Scrum has more value than localized Scrum.
- All Xebia projects of more than a few people are fully distributed today.
All the benefits of ProRail plus
– Quadrupled new user acquisition rate
– Quadrupled web site page views

Member Growth

Monthly Page View per visit

© Jeff Sutherland 1993-2009
Questions?

Emergent Architecture