

Getting to Done

The Secret Sauce of High Performing Teams



Hosts: *Joel Riddle*
Jeff Sutherland

Scrum Inc. is the Agile leadership company of Dr. Jeff Sutherland, co-creator of Scrum. We are based at the MIT Cambridge Innovation Center, MA.

CEO Jeff Sutherland helps companies achieve the full benefits of Scrum leading our comprehensive suite of support services and leadership training:

- Adapting the methodology to an ever-expanding set of industries, processes and business challenges Training (Scrum Master, Product Owner, Agile Leadership, online courses, etc.)
- Consulting (linking Scrum and business strategy, customizing Scrum)
- Coaching (hands-on support to Scrum teams)



Chief Content Officer JJ Sutherland maintains the Scrum framework by:

- Capturing and codifying evolving best practices (Scrum Guide)
- Conducting original research on organizational behavior
- Publishing (3 books) and productizing ScrumLab



Principal Hardware Engineer Joe Justice leads our hardware consulting practice:

- Worldwide consulting at leading hardware companies
- 700-800% performance improvement in hardware development
- Builds 100 mpg cars in his garage with help from 500 people in 32 countries



We run our company using Scrum as the primary management framework, making us a living laboratory on the cutting edge of “Enterprise Scrum”

Find out more at www.scruminc.com.

Certified Scrum Professional Earning SEU's

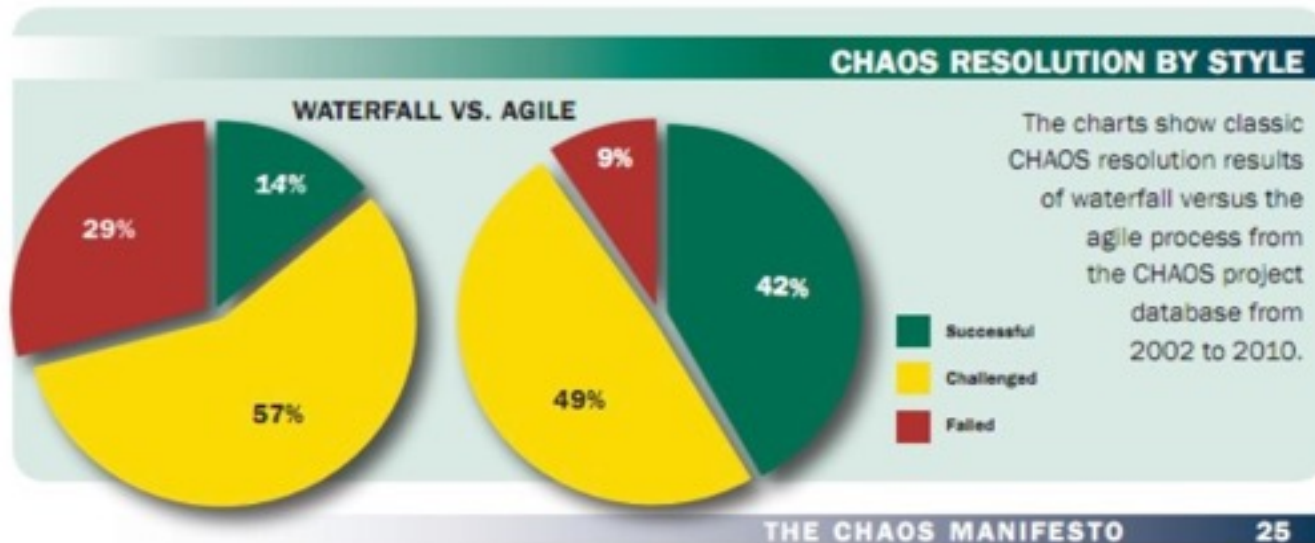
Category B: Scrum Alliance Courses or Coaching

There is no limit to the number of SEUs you can submit in this category by working with Scrum Alliance CSTs, REPs, and Coaches. However, a **minimum of 14 SEUs is required in this category for CSP certification**. A full day of training is equal to a maximum of 8 SEUs. You can earn SEUs by:

- Achieving your CSM, CSPO, and/or CSD certifications. We expect to see at least one of these certifications in Category B, and the certification must be current. CSM, CSPO, or CSD certification training is accepted even when training is more than 3 years old. All other SEUs need to have been earned within the past 3 years.
- Earning **Added Qualifications** in advanced Scrum topics.
- Attending training courses provided by a CST. Training by CSTs may include webinars, recorded training, and face-to-face courses.
- Attending training courses provided by a REP. Please note that the course *and* trainer must be listed on the Scrum Alliance website, which you may validate via the Course Search page:
<https://www.scrumalliance.org/courses-events/course>.
- Through one-on-one or small group coaching provided by a CSC. Members need not submit SEUs after each coaching session but may list them all at the time of submission.

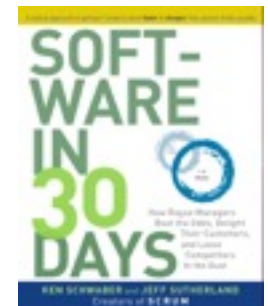
Also useful for Project Management Institute PDUs.

Do you have working software at the end of a sprint?



58% of Agile is “Bad Agile”

Source:



Reasons for no working software ...



Why Is It So Important to Have Working Product?



Teams That Finish Early Accelerate Faster!

Patterns for Coaches - ScrumPlop.org

Teams that Finish Early Accelerate Faster

- **Stable Teams** - How you get started
- **Yesterday's Weather** - How you pull backlog into a sprint
- **Daily Clean Code** - How to get defect-free product at sprint end
- **Swarming** - How you get work done quickly
- **Interrupt Pattern** - How to deal with interruptions during the sprint
- **Stop the Line** - How to deal with emergencies
- **Scrumming the Scrum** - How to ensure you improve continuously
- **Happiness metric** - How to ensure teams aren't overburdened

Teams That Finish Early Accelerate Faster: A Pattern Language for High Performing Scrum Teams

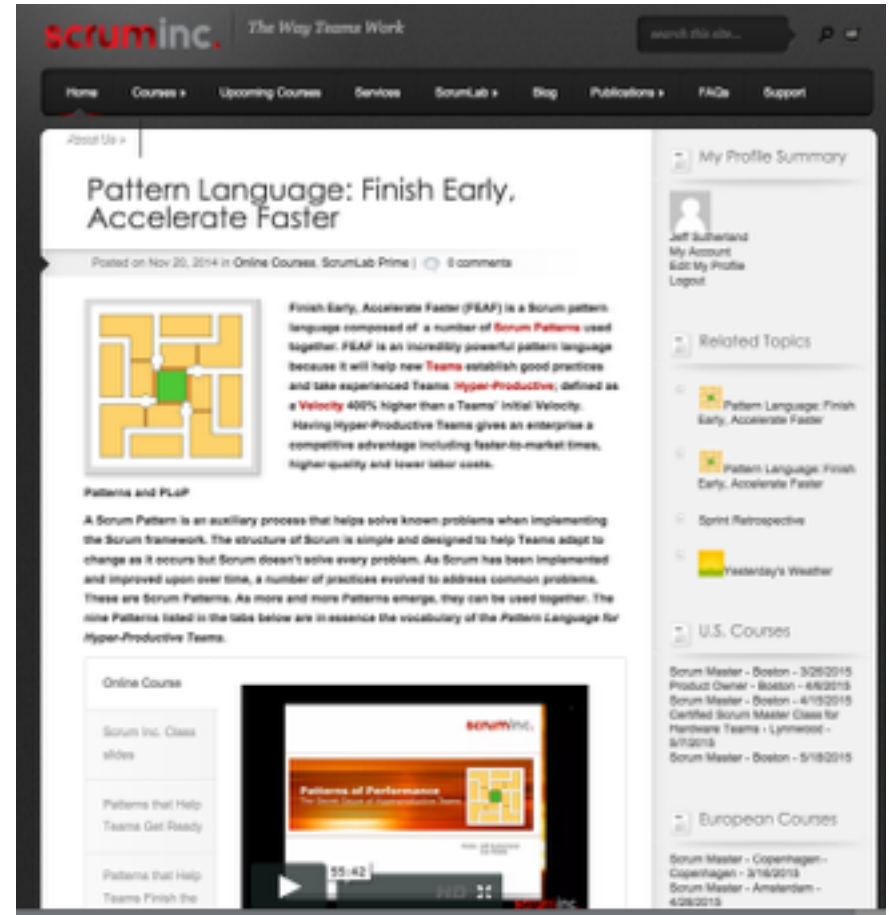
47th Hawaii International Conference on System Sciences (HICSS)

By Jeff Sutherland, Neil Harrison, Joel Riddle

January 2014

ScrumLab Vision Statement

- **FOR** experienced Scrum practitioners (Jill) who are “in the trenches”
- **WHO need** clear and actionable information to answer specific Scrum questions whenever they arise
- **ScrumLab** is the authoritative, curated on-demand source for Scrum Inc.’s leading thinking
- **That:**
 - Clearly explains Scrum and its underlying principles (i.e. why it works)
 - Shares successful advanced practices for different contexts
 - Provides actionable solutions to implement successfully
 - Is available whenever you need it
- **Unlike** other online Scrum resources
- **Our product** captures decades of successful experience and wisdom from the co-creator of Scrum and his hand-picked team of thought leaders



Why Don't Teams Have Working Software?



- Poor definition of DONE
- Stories not READY
- Dysfunctional leadership
- Technical debt
- Organizational debt
- Ineffective coaching

Source: ScrumInc/VersionOne Workshop 14 Oct 2014

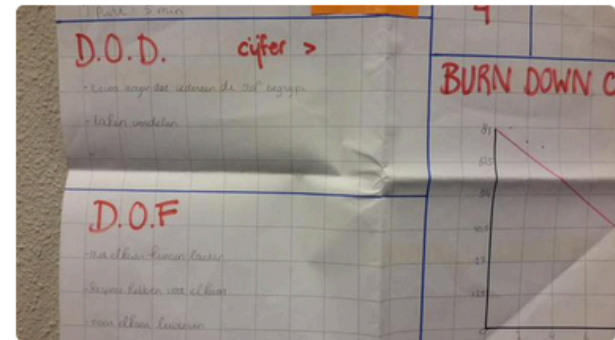
Poor Definition of DONE

- **Definition of DONE unclear**
 - It is impossible to be DONE if you don't know what DONE is.
- **Lack of consistent quality standards**
 - Definition of DONE does not include "working software".
 - Dysfunctional Product Owner accepts stories that are not done.



Ward Bergmans
@wardbergmans

Besides a Definition of Done, #eduScrum has a Definition of Fun! :-) #xpdays
pic.twitter.com/IIY1wxFhvf



Followed by Jacco Rademaker, martin wolters and Geert Bossuyt.

Stories Not Ready

- **Sizing stories**

- Bad estimates - inconsistent use of story points
- Taking stories too big into sprint
- Taking too many stories into sprint

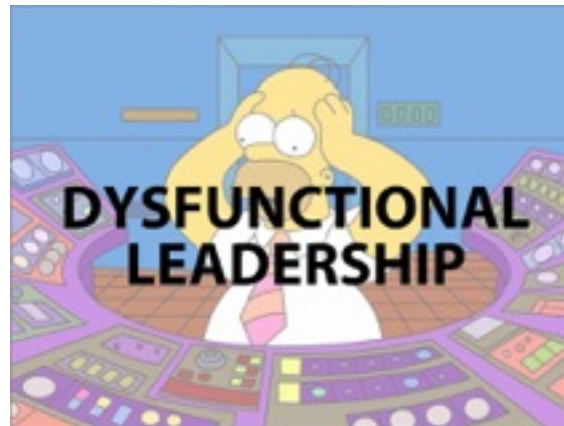
- **Poorly written stories**

- Stories not clear, particularly acceptance criteria
- Unidentified dependencies



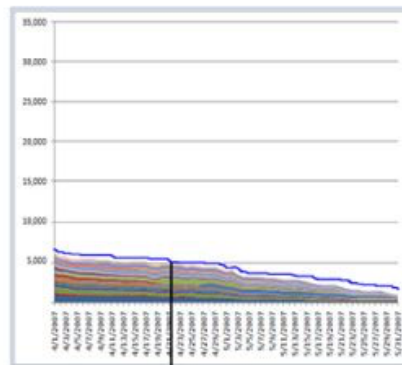
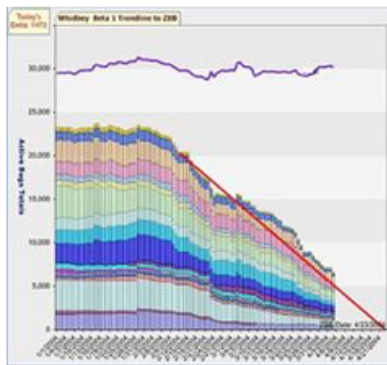
Dysfunctional Leadership

- Too many projects in pipeline (Context Switching)
- Everything is top priority
- Pressure to get things done delays projects and reduces quality
- Lack of understanding of Scrum
- Fear of exposure or change in responsibilities
- No continuous integration and/or no testing at all (Obamacare)



Technical Debt

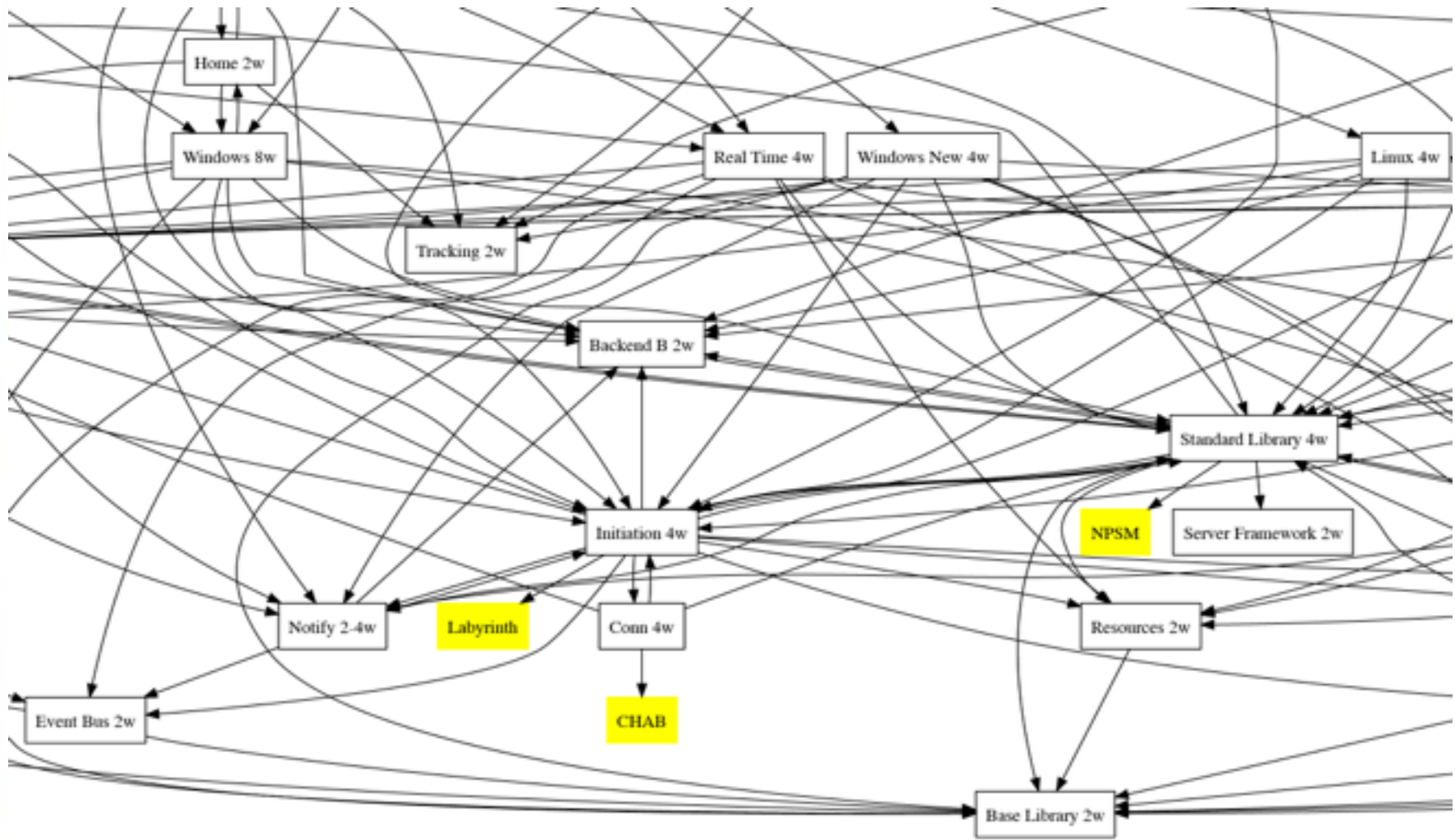
- Not finishing sprints creates bad code - 24x delay
- Legacy code is often accumulation of mountain of technical debt which reduces velocity
 - Severely aggravated by not using current technology for continuous integration and automated testing
 - Technical debt is incurred by running development too close to maximum which generates short cuts, lack of refactoring, loss of creativity, demotivation, and sloppy craftsmanship



Total bug debt at Beta 1 of VS 2008.

Microsoft TFS Mountain of Technical Debt - Scrum reduced bugs from 30000 to 2000 - Agile Software Development with Vision Studio, 2011

Organizational Debt

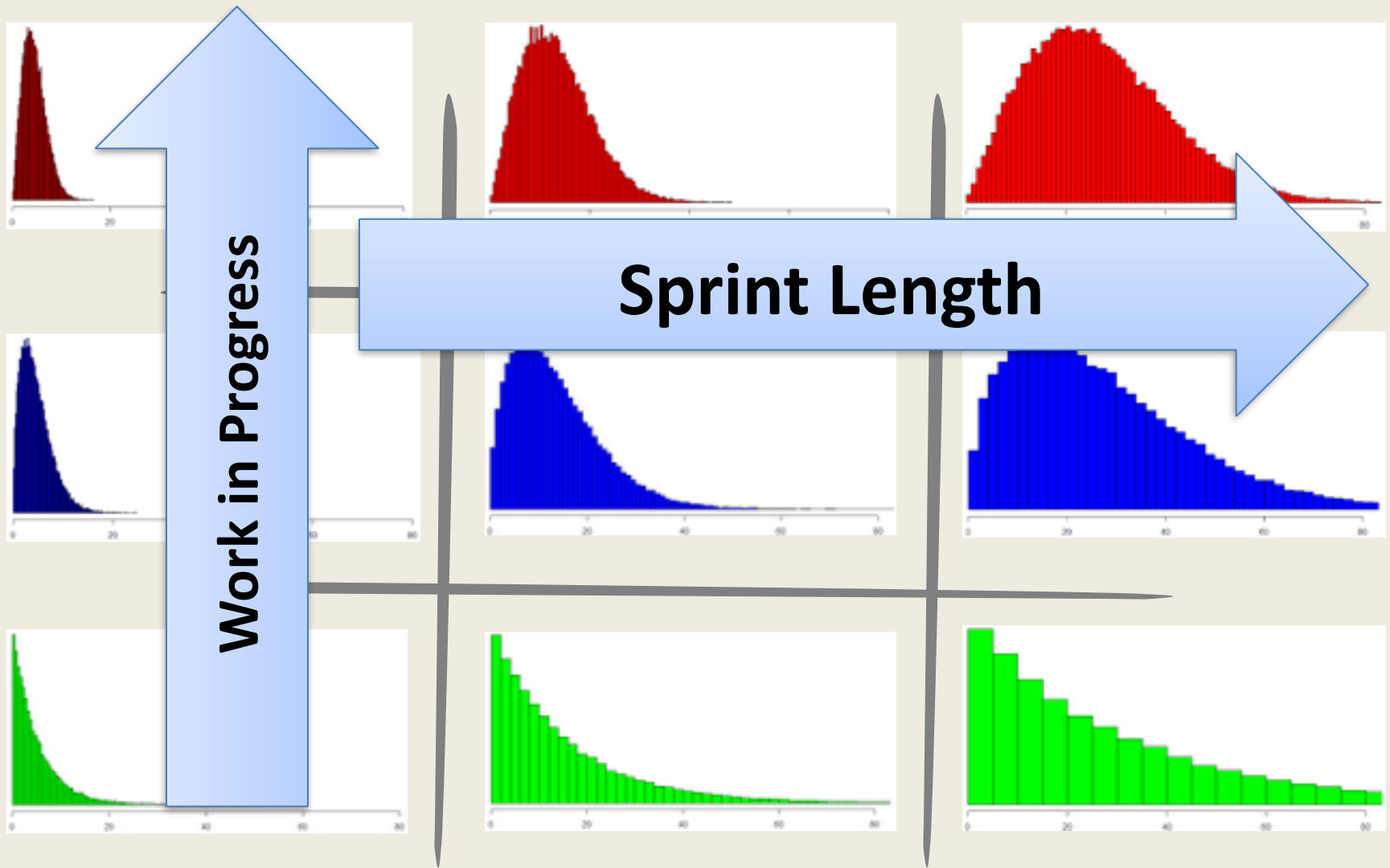


Agile Enterprise Metrics - 2015 48th Hawaii International Conference on System Sciences
Daniel R Greening, Senex Rex
dan@senexrex.com

Poor Coaching

- Silo's and specialization cripple velocity
 - specialized test teams are the worst example
- Developers not functioning as a team
 - minimal collaboration, no swarming
- No continuous improvement flatlines velocity
 - no happiness
 - no interrupt pattern
 - no scrumming the scrum
- "Pretend Agile" - no teamwork, no working software, no customer collaboration, and no effective response to change





< 1 week

~ 2 week sprint

~ 1 month

Time to Completion of a Story

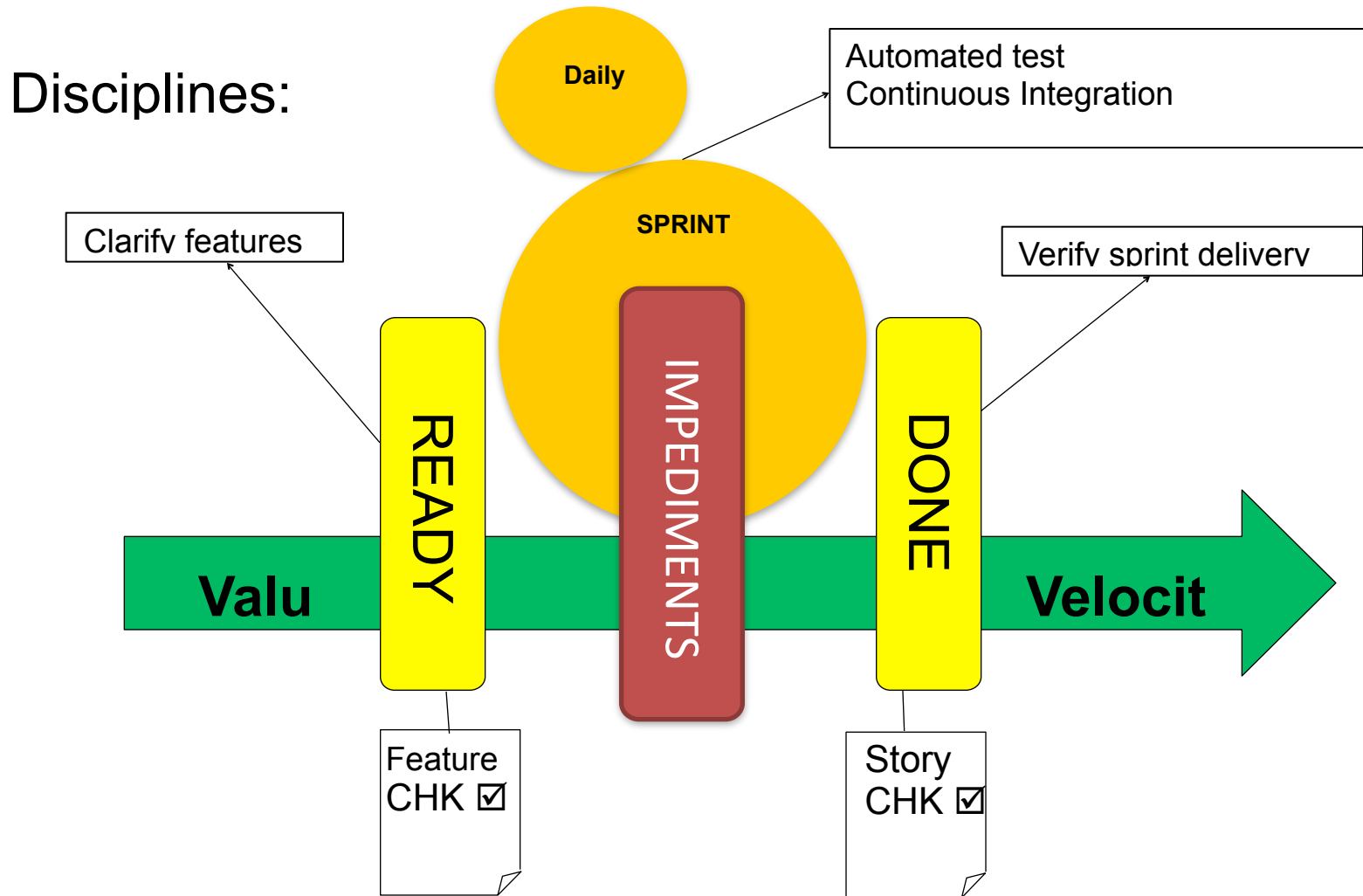


Systematic Approach to Getting To Done

- Implementing the Definition of Done
- Ensuring that backlog is Ready
- Training management
- Technical debt remediation plan
- Refactoring the organization
- Upgrading coaching and Scrum Master positions



Systematic Scrum Model



Scrum and CMMI - Going from Good to Great: Are You Ready Ready to Be Done Done
C. Jakobsen and J. Sutherland, in Agile 2009, Chicago, 2009.

Implementing Done

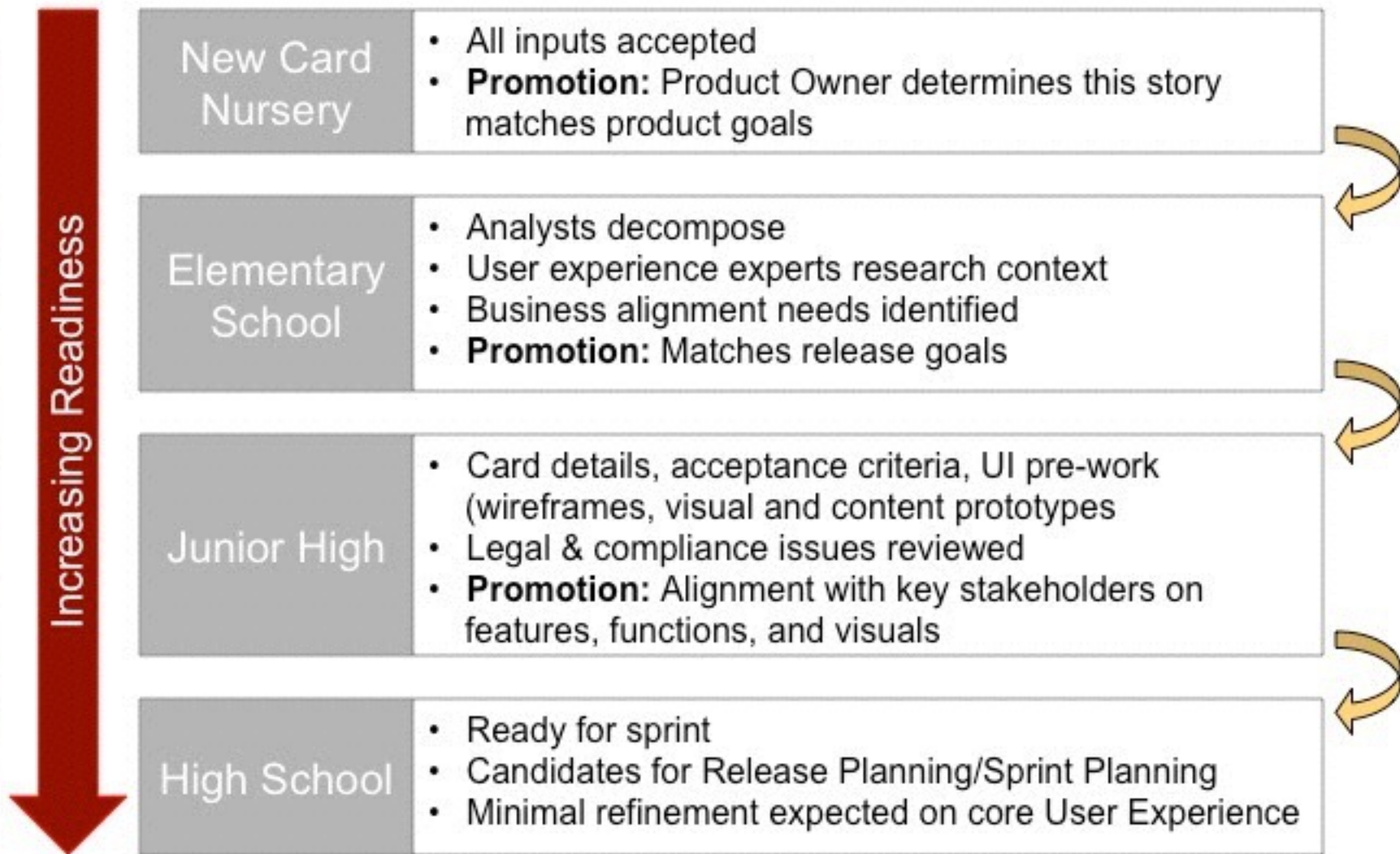
- **Definition of Done** must include integration testing and test capacity must exceed coding capacity
- Testers must be on the Scrum team - no separate test teams
- **Do not take too much into sprint.** Use Patterns.
 - Use “Yesterday’s Weather” pattern
 - “Illigitimus Non Interruptus”, and
 - “Scrum Emergency Procedure”
- **Use automated build system** combining new and old code (continuous integration)
- Systematically build **automated acceptance tests** which prevent top priority problems first
- Bugs **fixed in less than a day**
 - “Daily Clean Code”
 - 70% of defects are integration defects. Testing them later will take up to 24 times more testers!

Implementing Ready

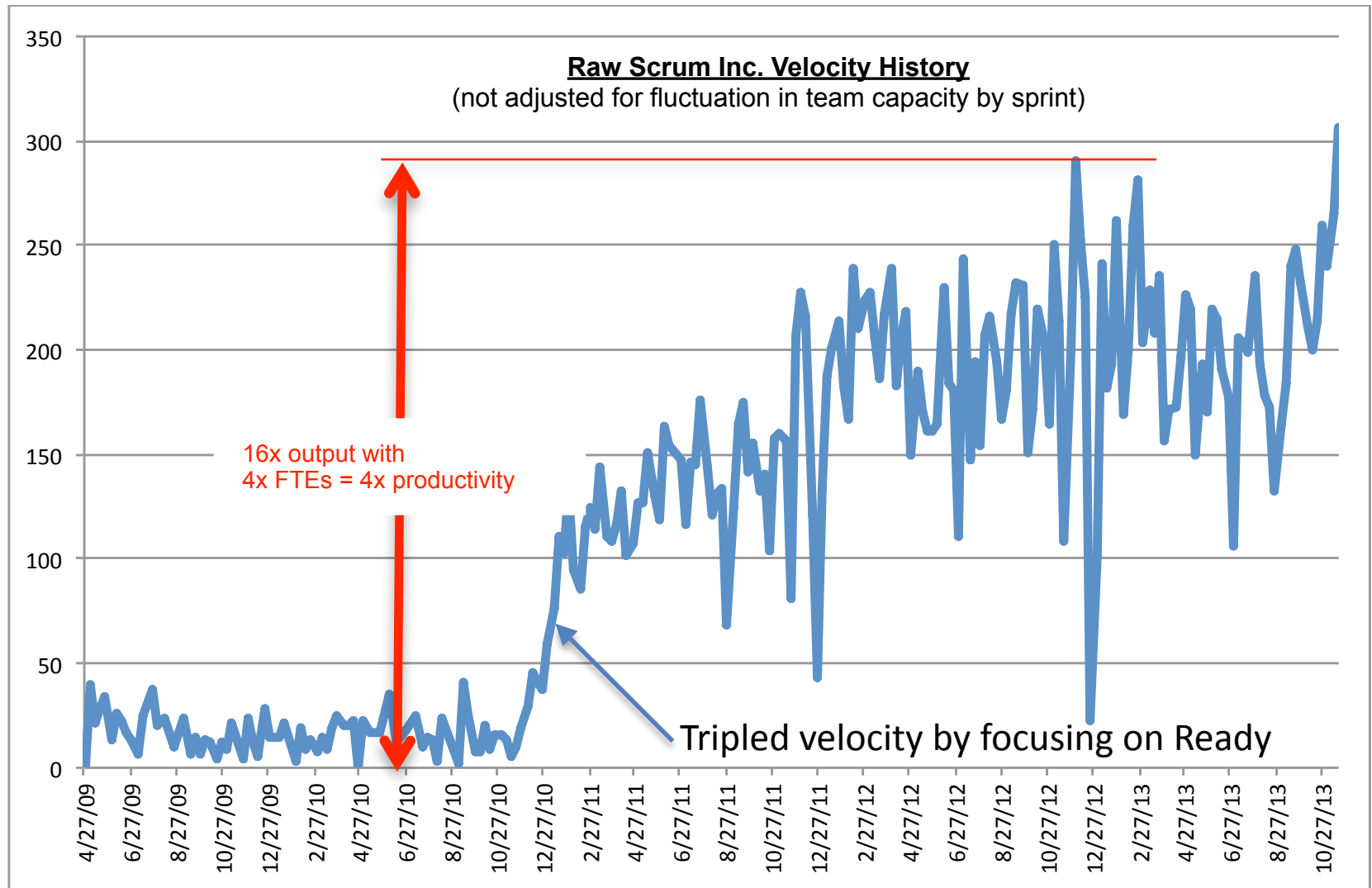
- Scrum Guide updated to include concept of **Ready**
- Team agrees on common **Definition of Ready**
- Only Ready Stories into Sprint Backlog
- **Backlog Refinement assures Ready** state.
- A good Ready state can **triple velocity**. Spend the time needed to get the backlog Ready.



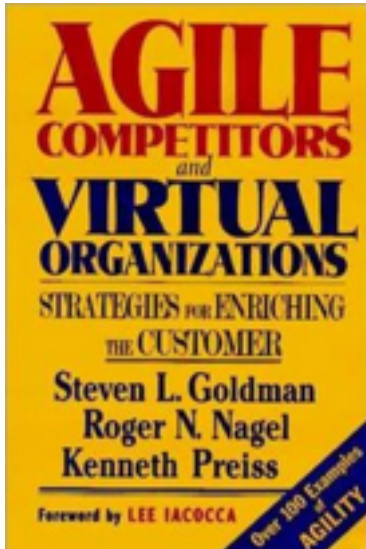
User Story Readiness Progression



Using Ready to Triple Velocity



Functional Leadership



- **Agile competition goes beyond lean manufacturing** by permitting the customer, jointly with the vendor or provider, to determine what the product will be.
- For agile competitors, the ability to individualize products comes at little or no increase in manufacturing cost. It does, however exact a cost: **It requires major changes in organization, management philosophy, and operations.**
- Managers need to be trained in how to lead Agile teams.

Leadership Responsibilities

- Provide challenging goals for the teams
- Create a business plan and operation that works
 - Set up the teams (in collaboration with teams)
 - Provide all resources the teams need
- Identify and remove impediments for the teams
 - Know velocity of teams
 - Understand what slows teams down (impediment list)
 - Remove waste (first-things-first)
- Hold P.O. accountable for value delivered per point
- Hold S.M. accountable for process improvement and team happiness



Fix Technical Debt

- **Remediate**
 - 80% of bugs come from 20% of code (or less)
 - IBM's strategy for determining remediation priorities - Mays et al. Experiences in Defect Prevention. IBM Systems Journal 29:1, 1990
- **Stop the Pain**
 - Systematically build acceptance tests into the build - highest priority first
- **Reduce the Debt**
 - Team build business case for Product Owner -
 - How many points for Tech Debt could go to value creation? (How long will it take to remove debt?)
- **Management commits to systematic improvement of product**
 - Reduce operational costs
 - Increase sales



Spotify Succeeds with Excellent Coaching

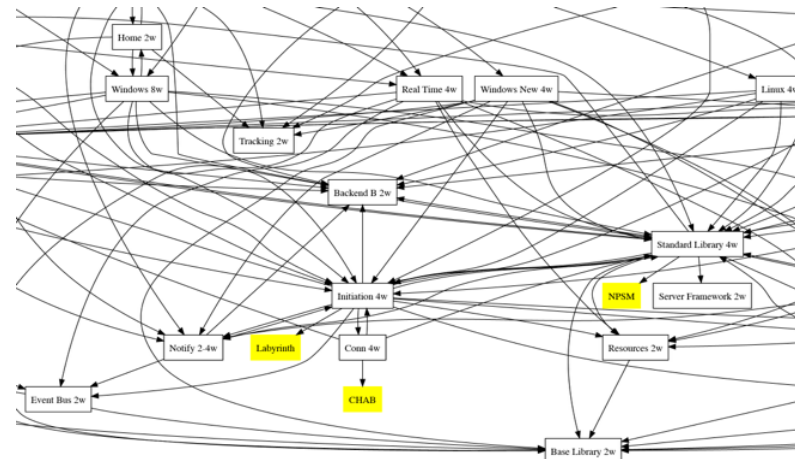


- Hires great workers
- Every team has a coach
 - Coaches are responsible for 1 process improvement every Sprint
 - Improvement backlog and they measure improvement continuously
 - Coaching has radically improved output of high performance teams.
- In the last year, 33% of all Spotify Teams have moved to continuous deployment multiple times per sprint.



Eliminate Organizational Debt

- Identify blocking teams and fix them
- Move towards cross-functional feature teams
- Train and hire T-shaped people
- Build out an object-oriented architecture for the product
- Use value stream mapping to identify queues, wait states, and other waste across the organization



Best Metrics for Coaches

- **Time to fix a defect.** If this averages less than 24 hours the team's velocity will double.
- **Measure of swarming.** How well do individuals and interactions generate performance.
 - Measure flow = actual work to do a story/ calendar time to done
 - If this is over 50% team velocity will double again



Going from Good to Great with Scrum
Are you **READY READY** to be **DONE DONE**?

Carsten Ruseng Jakobsen and Jeff Sutherland

SYSTEMATIC scruminc.

Cycle Time Improvement

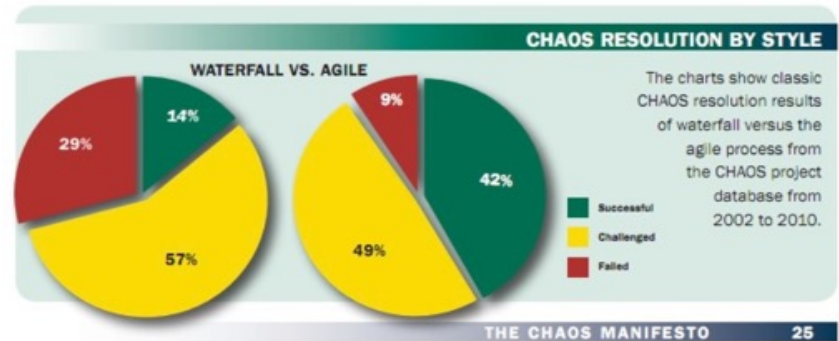
Table 2 – Example cash flow Improvement

Cycle Time	Forecast Date	Forecast Cost	Cash flow Benefit (cost savings + FY revenue)	
Current	15-Jul	1000K	$\$0 + \$60K = \$60K$	0%
10% Decrease	27-May	912K	$\$87K + \$90K = \$177K$	296% Better
20% Decrease	4-Apr	820K	$\$120K + \$145K = \$265K$	442% Better

Magennis, Troy. **The Economic Impact of Software Development Process Choice - Cycle-time Analysis and Monte Carlo Simulation Results.** 2015 48th Hawaii International Conference on System Sciences

Conclusions

- Bad Agile is caused by five primary factors
 - Poor definition of DONE
 - Stories not READY
 - Dysfunctional leadership
 - Technical debt
 - Organizational debt
 - Ineffective coaching
- Systematically focusing on remediating these issues will produce high performing teams with 200-400% improvement in production.
- Failure to focus on them will add yet another team to the 58% of teams that are “Bad Agile” leading to unhappy customers, lost revenue, and lower stock prices.



Questions?



Stay Connected

Scruminc.com

- For up coming events, new content releases, and more!

ScrumLab

- articles, online courses, tools, and papers on all things scrum
- www.scruminc.com/scrumlab

Blog

- <http://www.scruminc.com/category/blog/>

Online Courses

- advance your scrum with our online courses. Visit the Scrumlab store to view upcoming topics.

Twitter, Facebook, and G+

- @ScrumInc, @jeffsutherland, scrum and scrum inc.