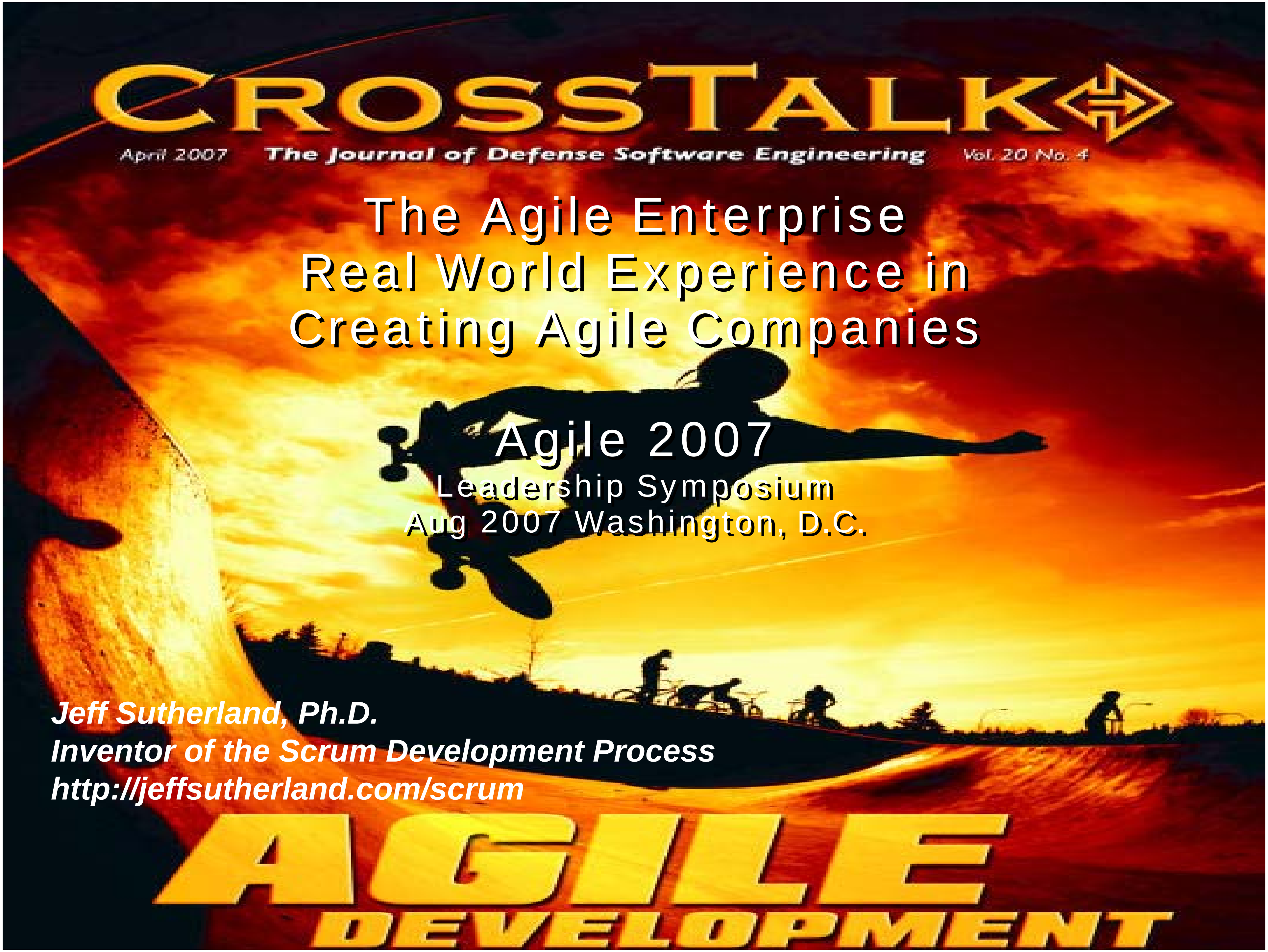


The Agile Enterprise Real World Experience in Creating Agile Companies

Agile 2007

Leadership Symposium
Aug 2007 Washington, D.C.

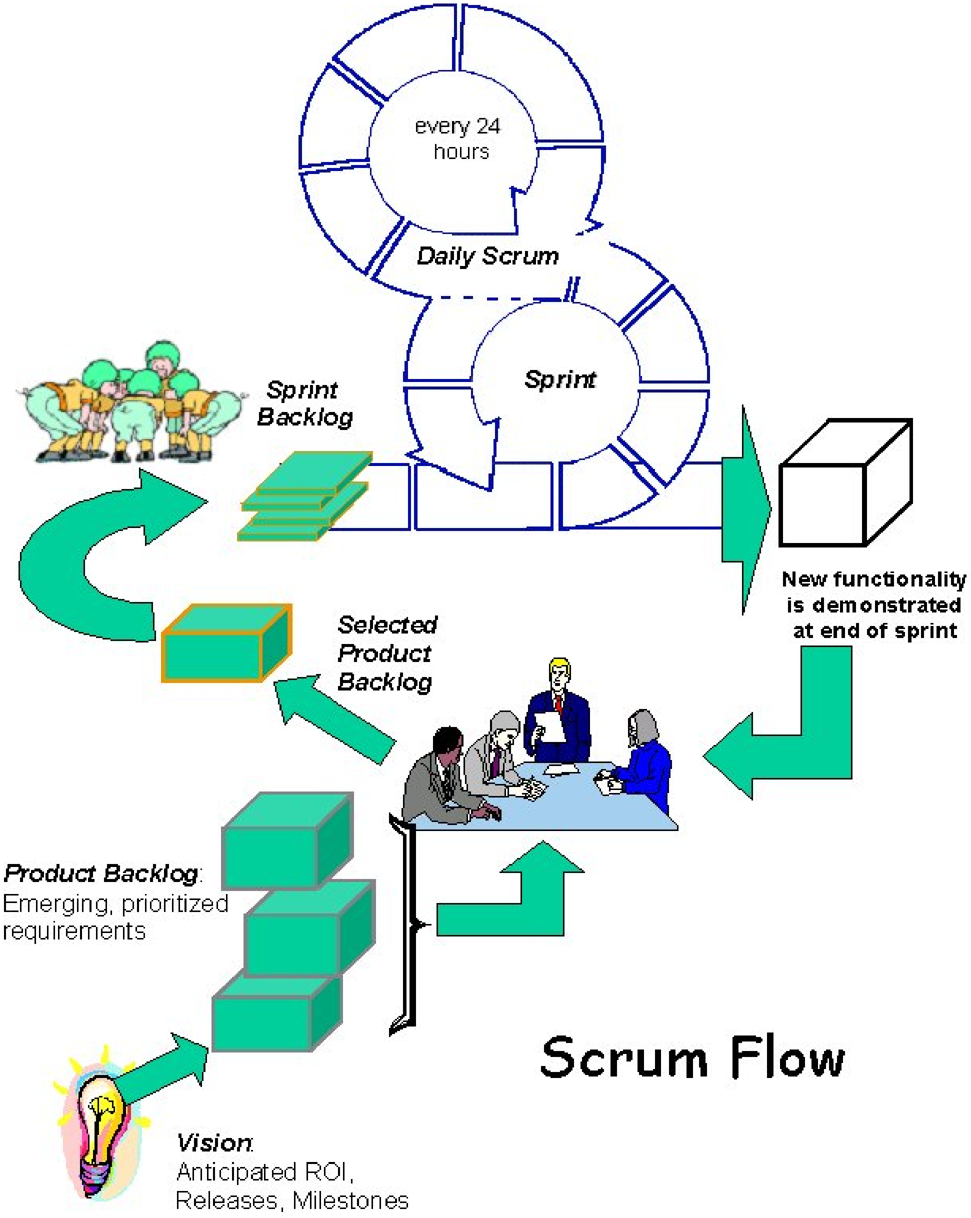
Jeff Sutherland, Ph.D.
Inventor of the Scrum Development Process
http://jeffsutherland.com/scrum

Jeff Sutherland jeffsutherland.com/scrum

CERTIFIED Scrum Trainer

Agile Alliance founders

- **CTO/VP Engineering of 9 software companies**
  - Prototyped Scrum in 4 companies
  - **Conceived and executed first Scrum at Easel Corp. in 1993. Rolled out Scrum in next 5 companies**
  - Scrum consultant to leading companies in Europe, North and South America, and Russia.
  - Senior Advisor, OpenView Investments, LLC
- **Signatory of Agile Manifesto**
- **CTO of PatientKeeper and Senior Advisor, OpenView Investments LLC**
  - ***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, author of Lean Software Development.***


every 24 hours
Daily Scrum
Sprint
Sprint Backlog
Selected Product Backlog
New functionality is demonstrated at end of sprint
Product Backlog: Emerging, prioritized requirements
Vision: Anticipated ROI, Releases, Milestones
Scrum Flow

Scrum

- Three Roles
  - Product Owner
  - ScrumMaster
  - Team
- Three Ceremonies
  - Sprint Planning
  - Sprint Review
  - Daily Scrum
- Three Artifacts
  - Product Backlog
  - Sprint Backlog
  - Burndown Chart

Scrum Operational Requirements

- Iterations
  - Fixed time
  - Working software that is tested
  - Agile specification
- Product Owner has Product Backlog
  - Prioritized
  - Estimated by Team
- Team has Burndown Chart
  - Knows velocity
  - No disruptions

Disruptions go to:
Next Sprint
Support allocation
Stop the line

Characteristics of Real World Agile Companies

- Agile is strategic initiative
- Scrum and/or XP is institutionalized
- Teams pass the Nokia test for Scrum
- Senior management and developers are totally involved
- Scrum is used in all areas of the company, not just development
- Companies have dramatic growth in size and revenue. Products are best of breed and projects are best in class.
- Product companies move into GartnerGroup magic quadrant.

Getting there: Xerox Parc innovations

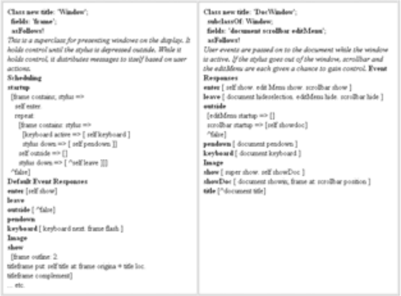
Personal Workstation

Mouse

Ethernet

Windows Interface

Laser Printer



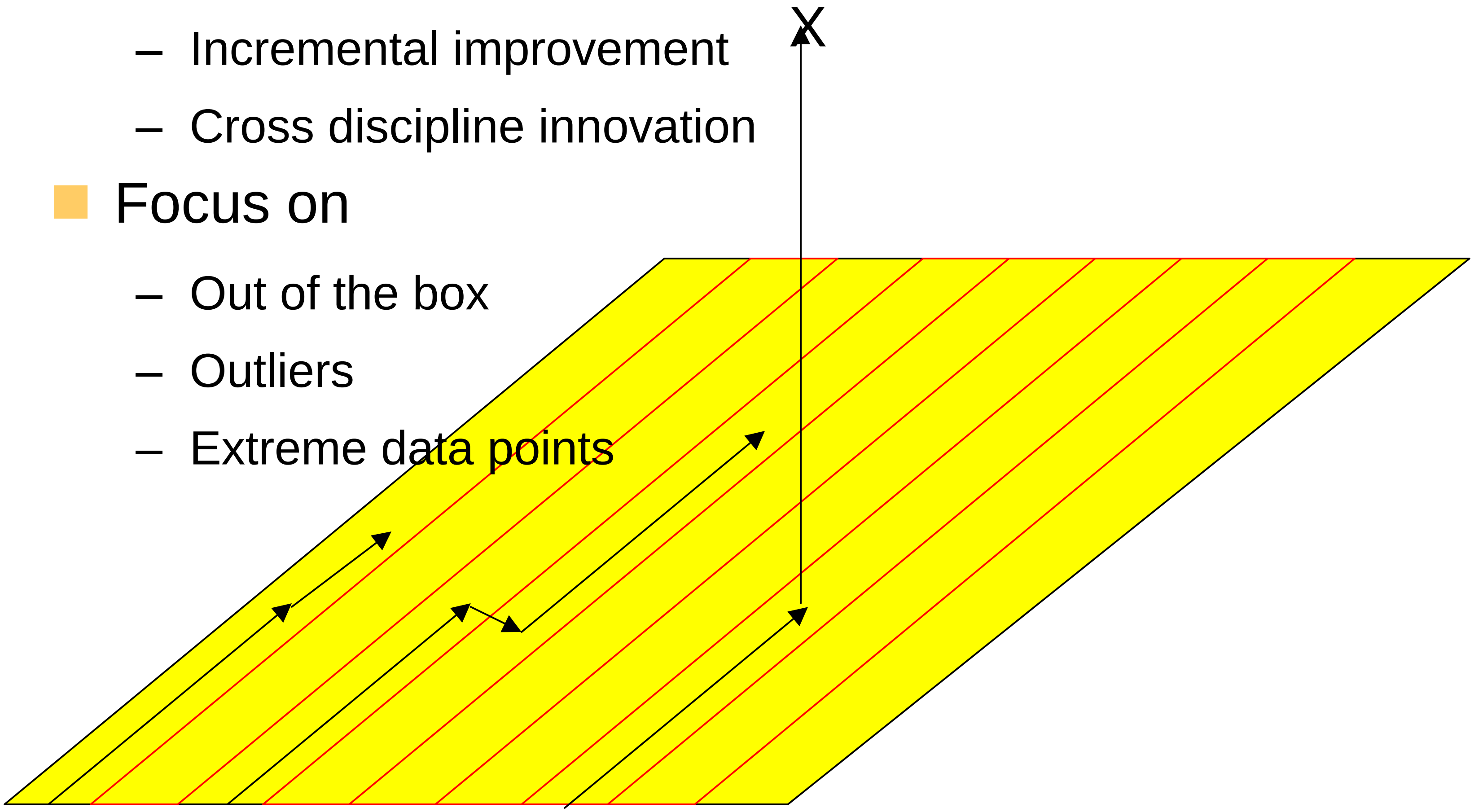
```
Class new title: 'Window';
  fields: 'frame';
  asFollows!
This is a superclass for presenting windows on the display. It
holds control until the stylus is depressed outside. While it
holds control, it distributes messages to itself based on user
actions.
Scheduling
startup
  [frame contains: stylus =>
    self enter.
    repeat
      [frame contains: stylus =>
        [keyboard active => [ self keyboard ]
        stylus down => [ self pendown ]]
      self outside => []
      stylus down => [ ^self leave ]]]
  ^false]
Default Event Responses
enter [self show]
leave
outside [ ^false]
pendown
keyboard [ keyboard next. frame flash ]
Image
show
  [frame outline: 2.
titleframe put: self title at: frame origina + title loc.
titleframe complement]
... etc.
```

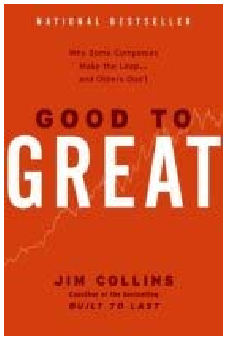
```
Class new title: 'DocWindow';
  subclassOf: Window;
  fields: 'document scrollbar editMenu';
  asFollows!
User events are passed on to the document while the window
is active. If the stylus goes out of the window, scrollbar and
the editMenu are each given a chance to gain control. Event
Responses
enter [ self show. edit Menu show. scrollbar show ]
leave [ document hideselection. editMenu hide. scrollbar hide ]
outside
  [editMenu startup => []
  scrollbar startup => [self showdoc]
  ^false]
pendown [ document pendown ]
keyboard [ document keyboard ]
Image
show [ super show. self showDoc ]
showDoc [ document showin; frame at: scrollbar position ]
title [^document title]
```

Smalltalk

Alan Kay's Strategy

- Forget about
  - Incremental improvement
  - Cross discipline innovation
- Focus on
  - Out of the box
  - Outliers
  - Extreme data points

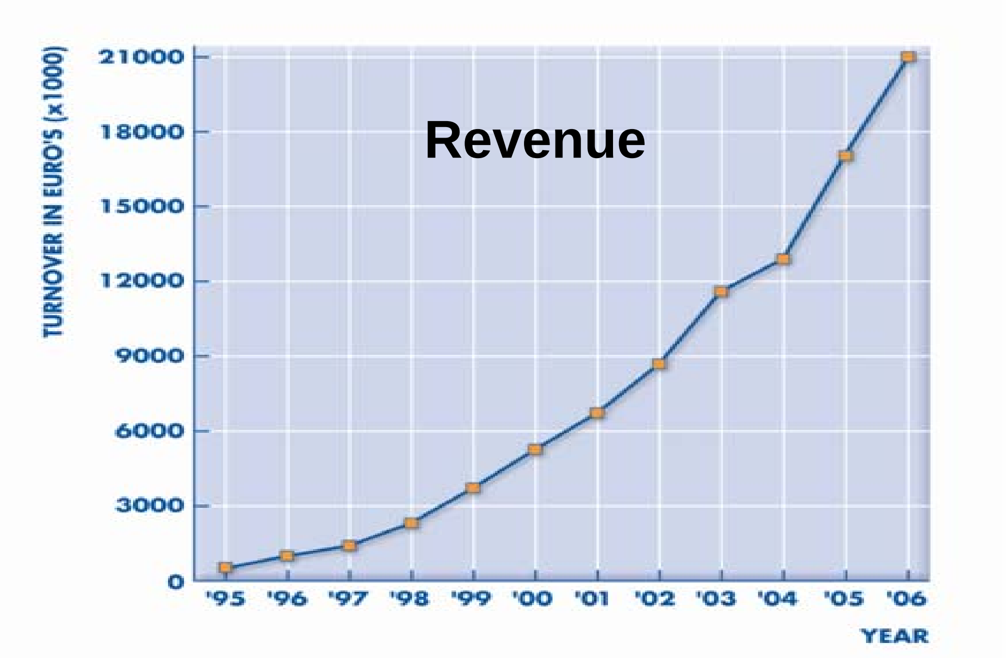


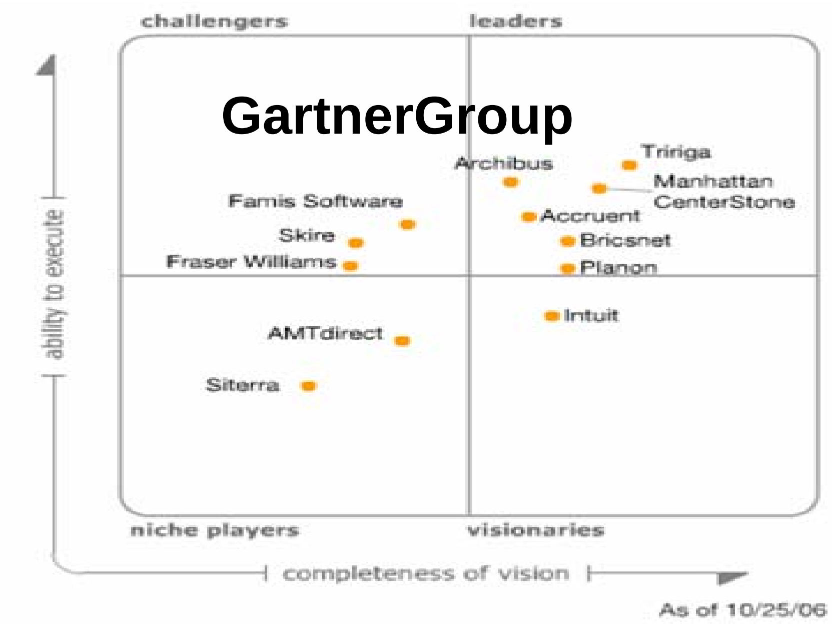

NATIONAL BESTSELLER
Why Some Companies Make the Leap... and Others Don't
GOOD TO GREAT
JIM COLLINS
Coauthor of the bestselling BUILT TO LAST

Out of the Box

- Scrum looked at projects that were off the plate
  - IBM surgical team
  - Borland Quattro Project
  - Takeuchi and Nonaka new product development strategies
- *Scrum: A Pattern Language for Hyperproductive Software Development*
  - By M. Beedle, M. Devos, Y. Sharon, K. Schwaber, and J. Sutherland. In Pattern Languages of Program Design. vol. 4, N. Harrison, Ed. Boston: Addison-Wesley, 1999, pp. 637-651.
- First Scrum was a hyperproductive Scrum. Management and sales asked the team to slow down.
- Latest Scrum is a hyperproductive revenue generating Scrum. The Board asked the company to slow down revenue recognition.

Planon Type B Scrum


Revenue


GartnerGroup

- Great means you are the industry leader in your market and revenue is skyrocketing
- Anyone can aspire to be great!
- That aspiration will make you and your company better

Are you doing Scrum?
The Nokia Test by Bas Vodde

NOKIA
Connecting People

- First, you must be doing iterative development
  - Iterations must be timeboxed to less than six weeks
  - Software must be tested and working at the end of an iteration
  - Iteration must start before specification is complete
- Then you must meet the Nokia Scrum test

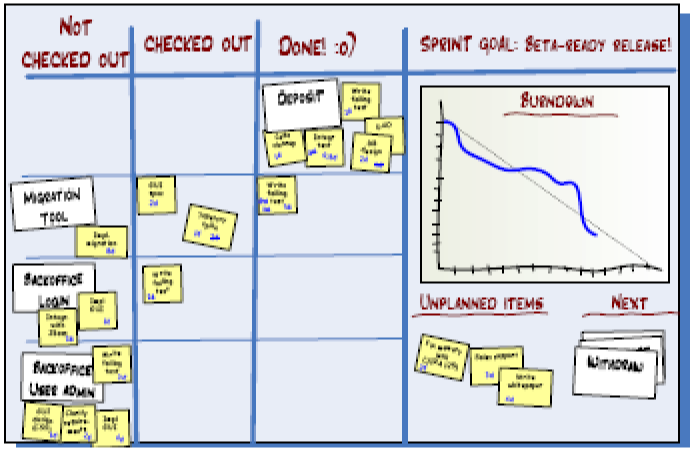
1969 - Earliest published reference to Iterative Incremental development

Robert Glass. Elementary Level Discussion of Compiler/Interpreter Writing. ACM Computing Surveys, Mar 1969

See Larman, Craig and Basili, Vic. Iterative and Incremental Development: A Brief History. IEEE Computer, June 2003 (Vol. 36, No. 6) pp. 47-56

For those doing Scrum

- You know who the product owner is
- There is a product backlog prioritized by business value
- The product backlog is has estimates created by the team
- The team generates burndown charts and knows their velocity
- There are no project managers (or anyone else) disrupting the work of the team



Kniberg, Henrik. Scrum and XP from the Trenches: How We Do Scrum. Version 2.1, Crisp, 5 Apr 2007.

OpenView Venture Partners

- Invest only in Agile companies
- Scrum and XP are Oracle and SQL Server of Agile processes. Portfolio companies must use them.
- Portfolio companies must pass the Nokia test
  - One product owner, one product backlog, estimated by developers
  - Fixed iterations with software done (working and tested)
  - Teams have burndown charts and know their velocity
  - No external disruption of teams during Sprint

Experiences with people doing Scrum

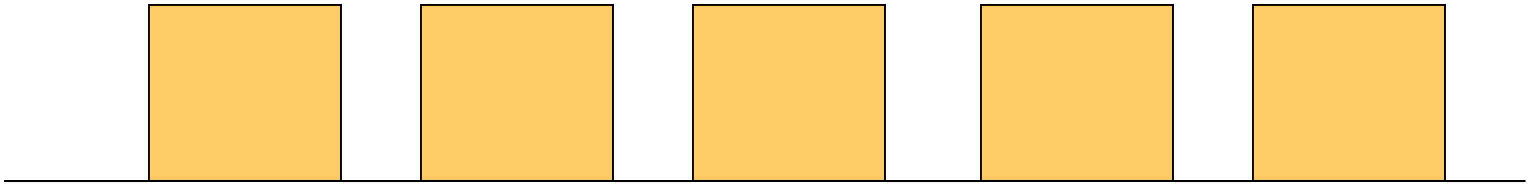
Hyperproductive Scrum Teams

- It is easy to double productivity with Scrum by implementing only high business value features.
- To quadruple productivity (Toyota effect) requires surfacing impediments and removing them (inspecting and adapting).
- Scrum was designed for 5-10 times productivity improvement. This has been experienced in three types of teams:
  - – *The first Scrum team and similar colocated teams.*
  - – *The first distributed Scrum team.*
  - – *Large distributed/outsourced projects.*
- Understanding hyperproductive Scrum teams can help improve your software development with Scrum.

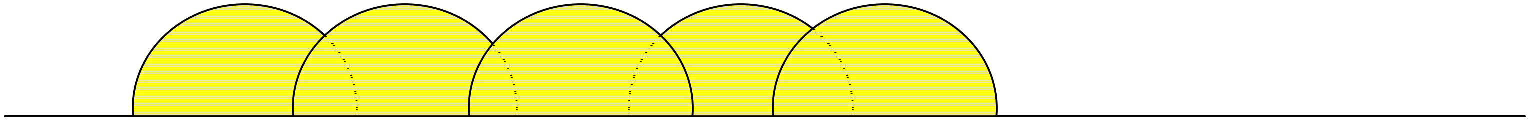
Outsourcing

- Outsource $2M development
- Outsourcing costs - $1.6M
  - Industry data show 20% cost savings on average
- Introduce Scrum locally
  - 240% improvement at IDX, for example
- Local Scrum costs – $0.83M
- SirsiDynix radically reduced outsource costs making oursourcing reasonable for:
  - Gaining expertise that is unavailable locally
  - Expanding and contracting development staff without layoffs

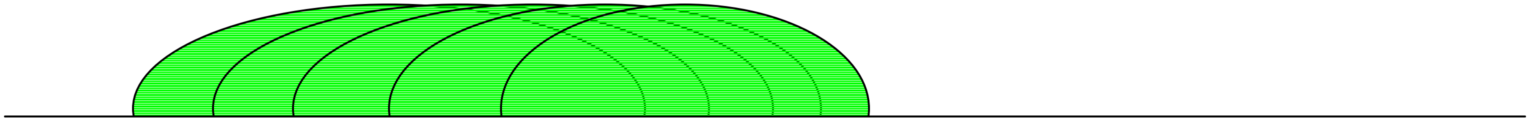
Distributed Scrum Styles



Isolated Scrums



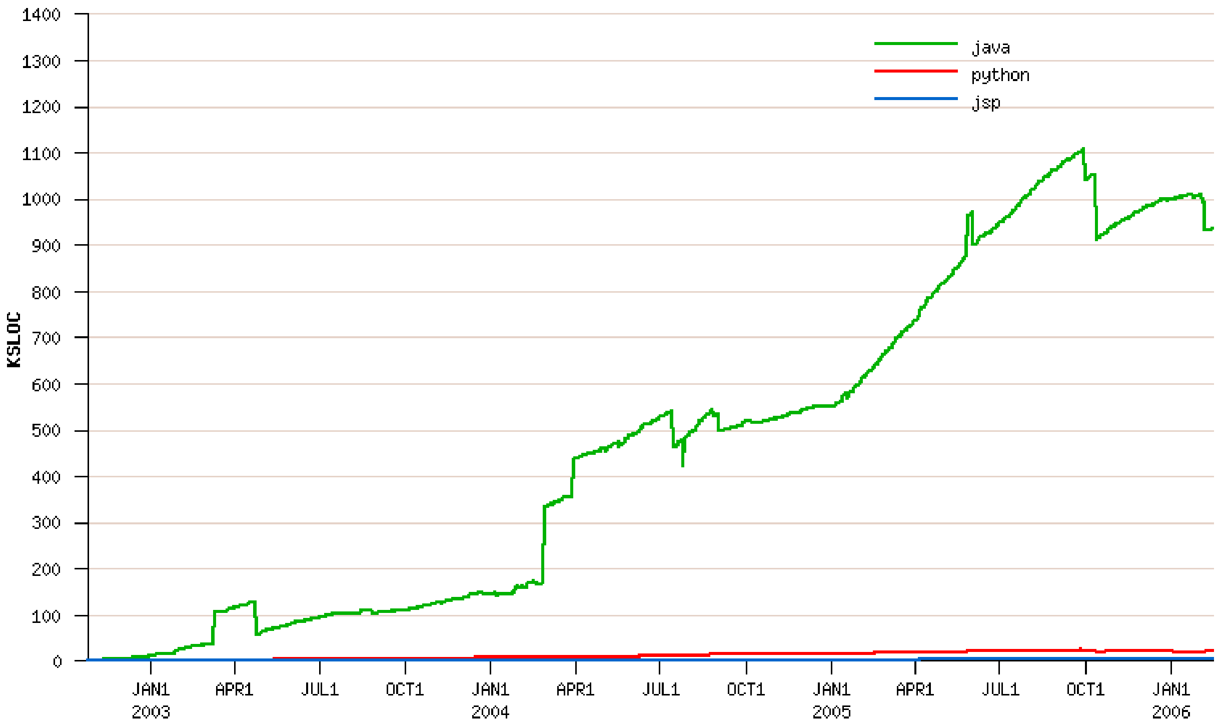
Distributed Scrum of Scrums



Totally Integrated Scrums

SirsiDynix Distributed Scrum

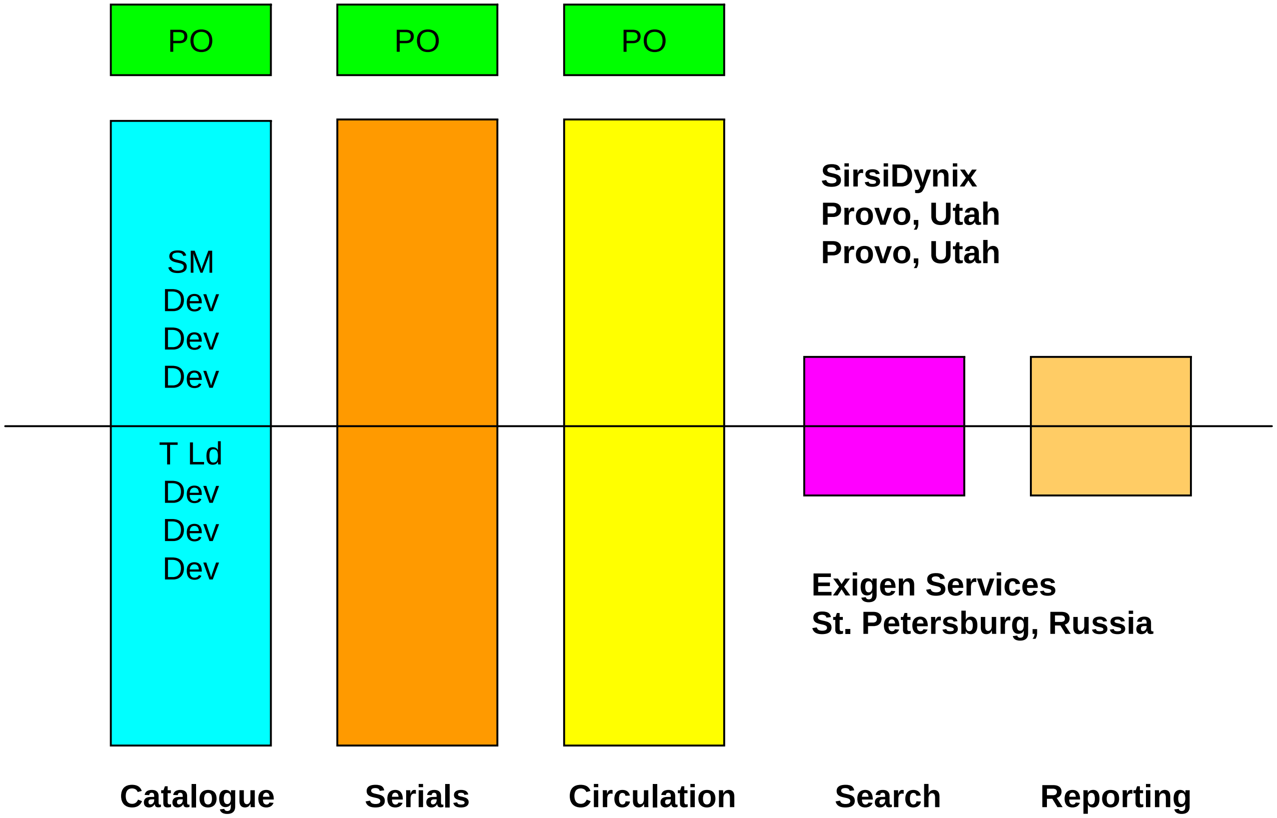
- Over a million lines of Java code



patientkeeper worldwide **scrum consulting** practice

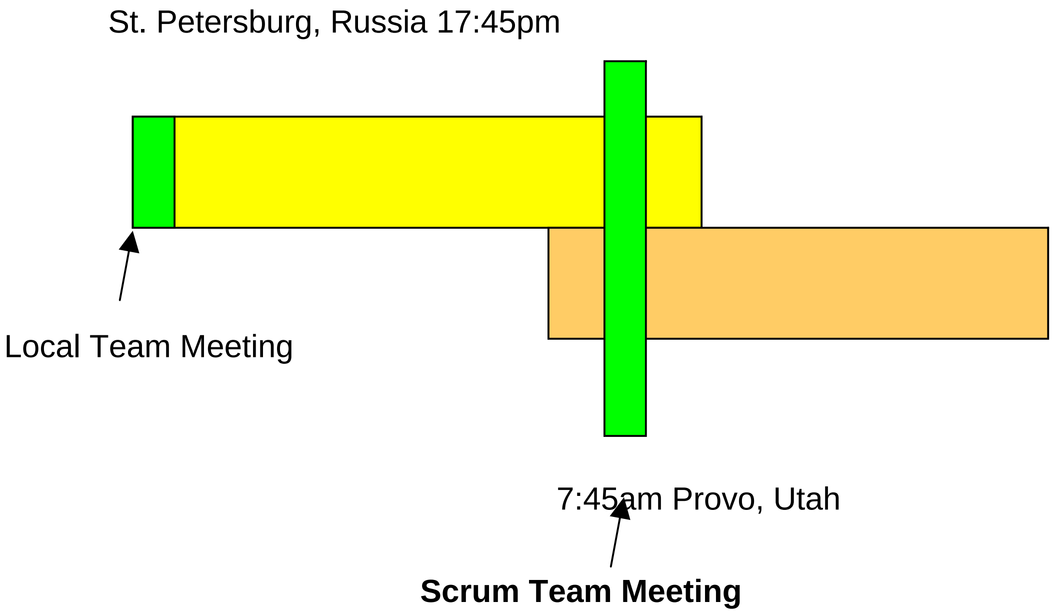
SirsiDynix Distributed Scrum

- 56 developers distributed across sites



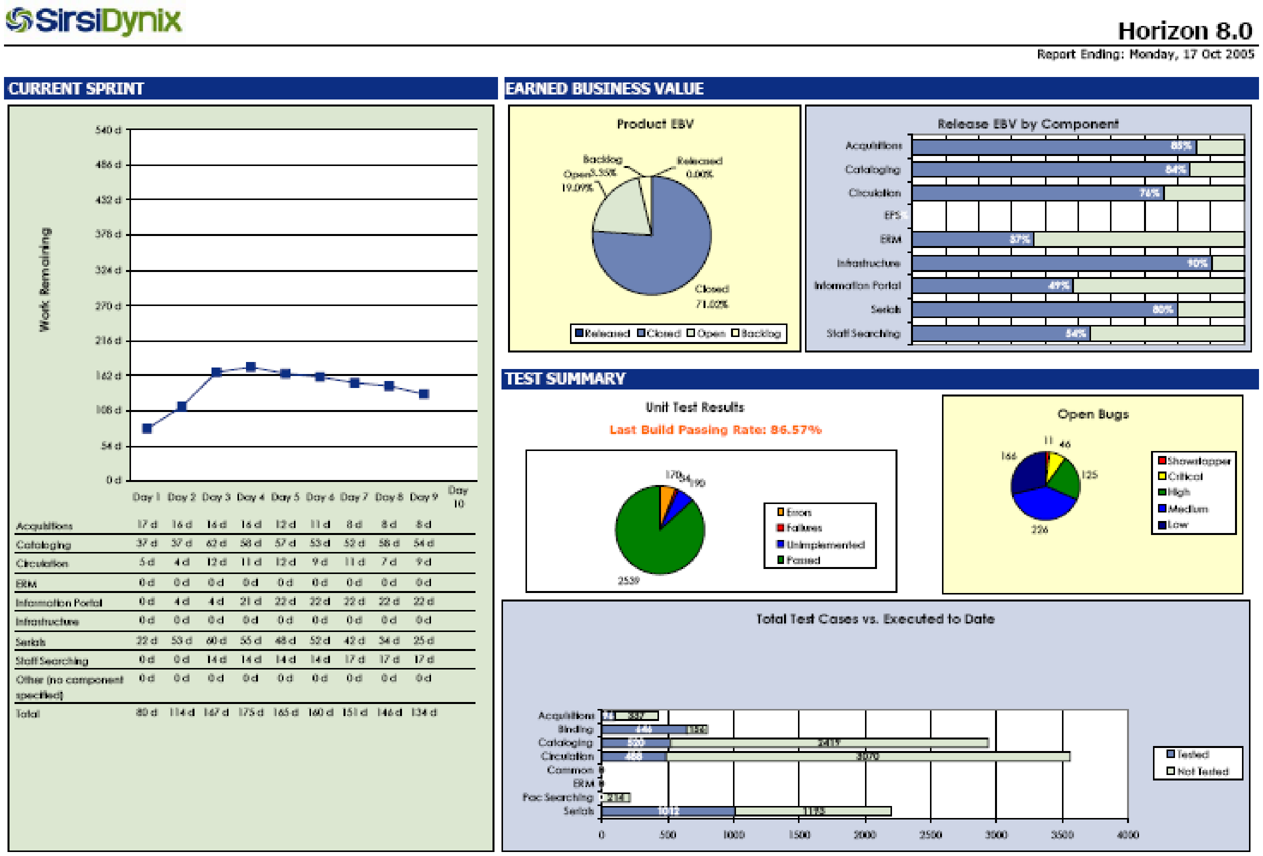
SirsiDynix Distributed Scrum

- Scrum daily meetings



SirsiDynix Distributed Scrum

- Common tools



SirsiDynix

Horizon 8.0

Report Ending: Monday, 17 Oct 2005

CURRENT SPRINT

	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Day 8	Day 9	Day 10
Acquisitions	17 d	16 d	16 d	16 d	12 d	11 d	8 d	8 d	8 d	
Cataloging	37 d	37 d	62 d	58 d	57 d	53 d	52 d	58 d	54 d	
Circulation	5 d	4 d	12 d	11 d	12 d	9 d	11 d	7 d	9 d	
ERM	0 d	0 d	0 d	0 d	0 d	0 d	0 d	0 d	0 d	
Information Portal	0 d	4 d	4 d	21 d	22 d	22 d	22 d	22 d	22 d	
Infrastructure	0 d	0 d	0 d	0 d	0 d	0 d	0 d	0 d	0 d	
Serials	22 d	53 d	60 d	55 d	48 d	52 d	42 d	34 d	25 d	
Staff Searching	0 d	0 d	14 d	14 d	14 d	14 d	17 d	17 d	17 d	
Other (no component specified)	0 d	0 d	0 d	0 d	0 d	0 d	0 d	0 d	0 d	
Total	80 d	114 d	147 d	175 d	165 d	160 d	151 d	146 d	134 d	

EARNED BUSINESS VALUE

Product EBV

Release EBV by Component

TEST SUMMARY

Unit Test Results

Last Build Passing Rate: 86.57%

Open Bugs

Total Test Cases vs. Executed to Date

patientkeeper worldwide **scrum consulting** practice

SirsiDynix Distributed Scrum

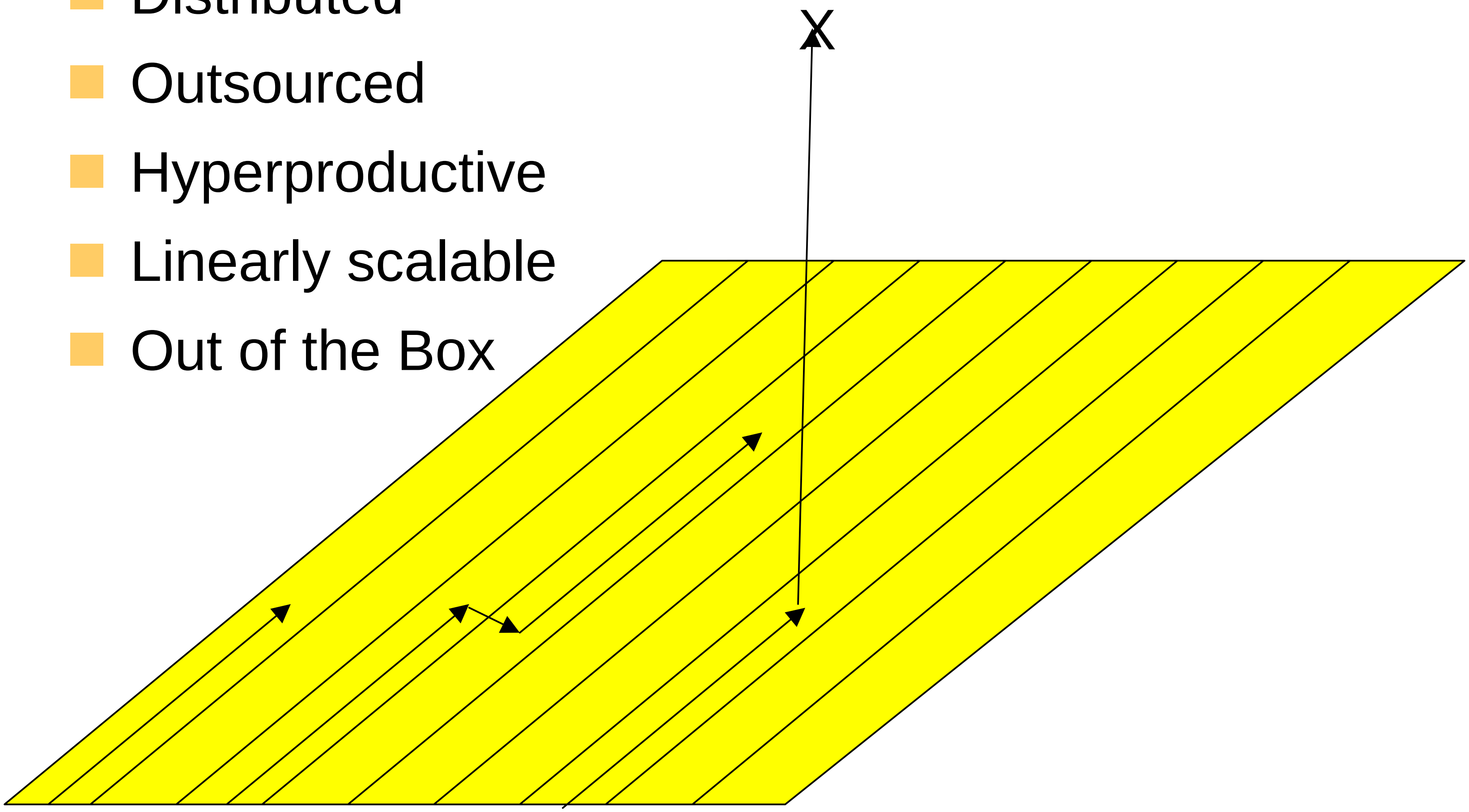
- Uncommon performance

	Colocated Scrum*	Waterfall*	SirsiDynix Distributed Scrum**
Person Months	54	540	827
Lines of Java	51,000	58000	671,688
Function Points	959	900	12673
FP per dev/month	17.8	2.0	15.3

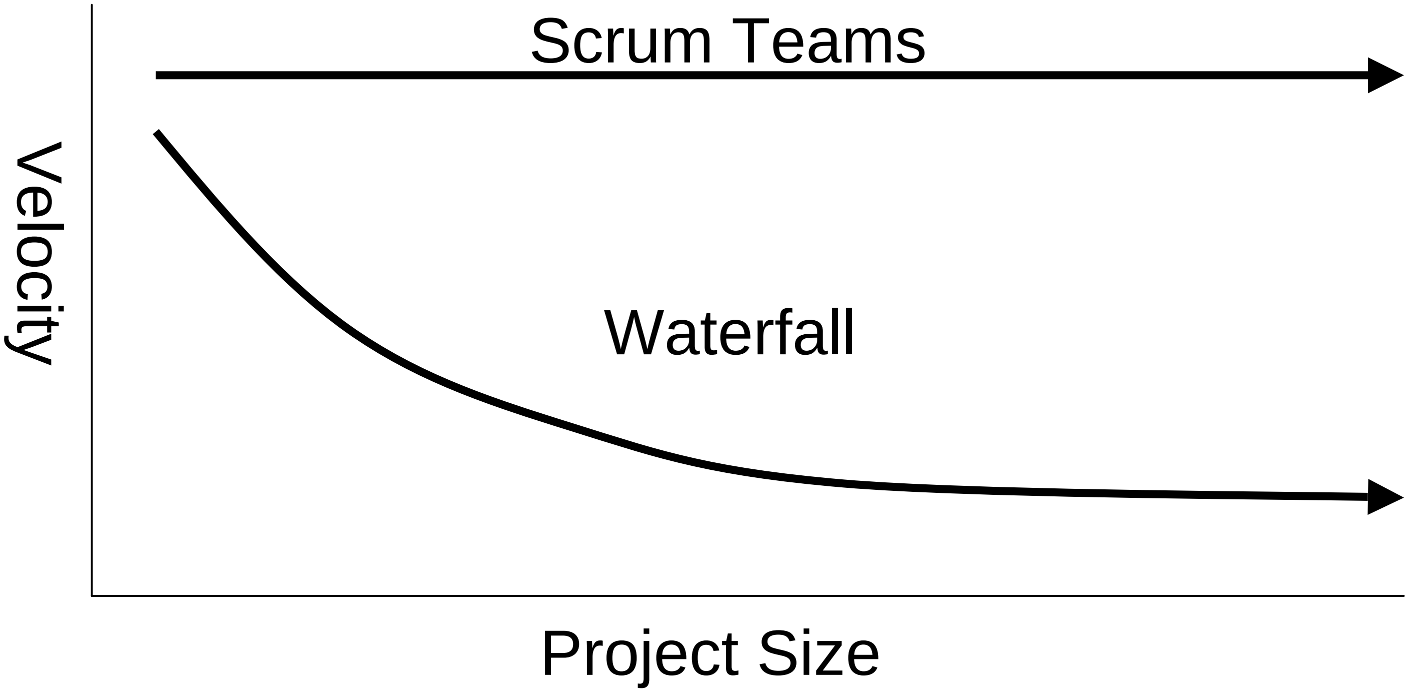
*M. Cohn, User Stories Applied for Agile Development. Addison-Wesley, 2004
**J. Sutherland, A. Viktorov, J. Blount, and N. Puntikov, "Distributed Scrum: Agile Project Management with Outsourced Development Teams," in HICSS'40, Hawaii International Conference on Software Systems, Big Island, Hawaii, 2007.
.

SirsiDynix was off the charts

- Distributed
- Outsourced
- Hyperproductive
- Linearly scalable
- Out of the Box



First Demonstration of Linear Scalability

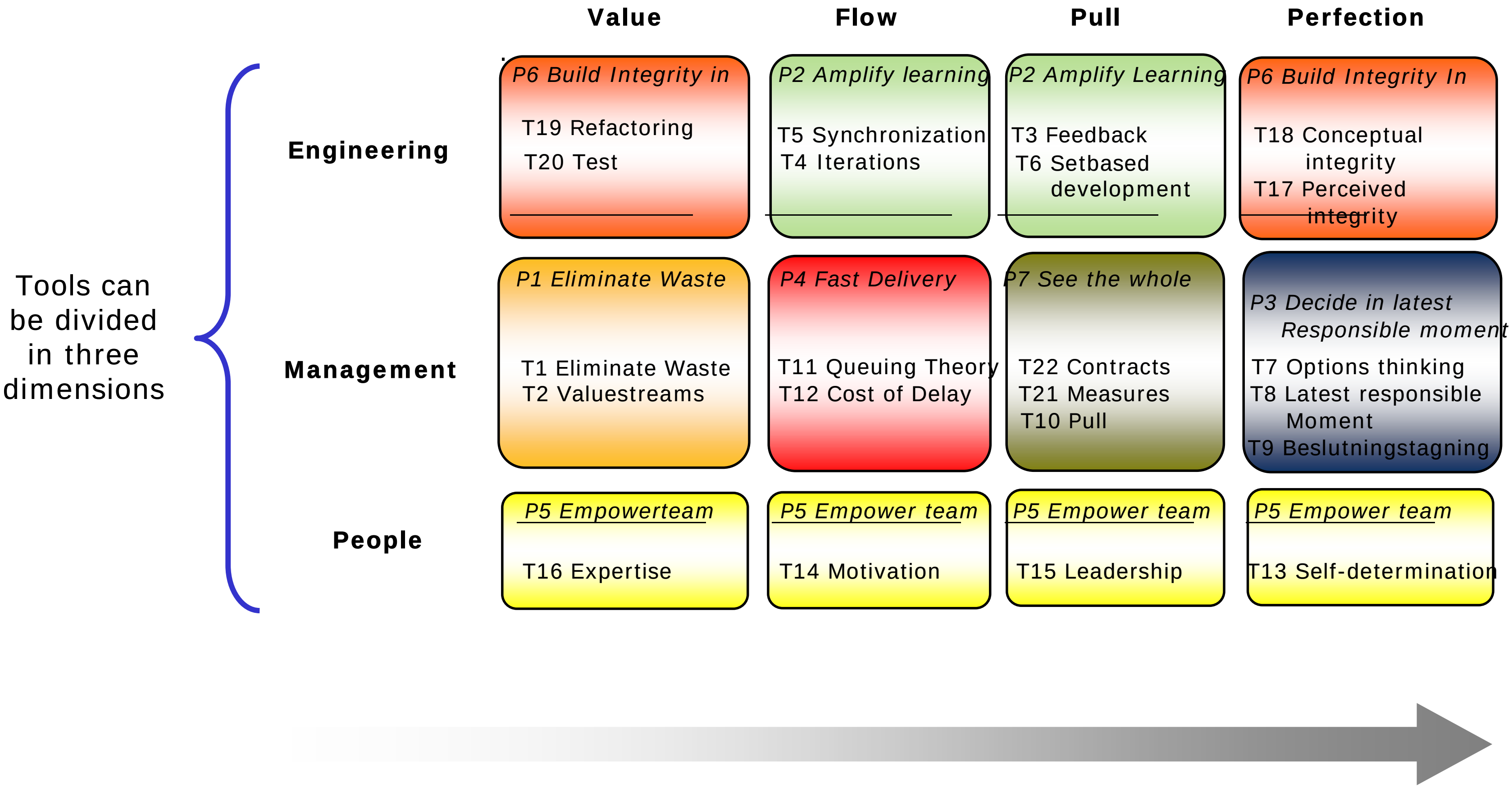


- J. Sutherland, A. Viktorov, J. Blount, and N. Puntikov, "Distributed Scrum: Agile Project Management with Outsourced Development Teams," in HICSS'40, Hawaii International Conference on Software Systems, Big Island, Hawaii, 2007.
- J. Sutherland, C. Jacobson, and K. Johnson, "Scrum and CMMI Level 5: A Magic Potion for Code Warriors!," in Agile 2007, Washington, D.C., 2007.

Poppendieck Lean Thinking Tools

P1 Eliminate waste	**P2** Amplify Learning	**P3** Responsible decisions	**P4** Fast Delivery	**P5** Empower team	**P6** Buil integrity in	**P7** See the Whole
Tool 1: Eliminate Waste	**Tool 3:** Feedback	**Tool 7:** Options Thinking	**Tool 10:** Pull	**Tool 13: S**elf-determinatoion	**Tool 17:** Perceived integritet	**Tool 21:** Measures
Tool 2: Value Stream Mappig	**Tool 4:** Iterations	**Tool 8:** Latest Responsible Moment	**Tool 11:** Queue Theory	**Tool 14:** Motivation	**Tool 18:** Conceptual Integritet	**Tool 22:** Contracts
	Tool 5: Synchronization	**Tool 9:** Decision Making	**Tool 12:** Cost of Delay	**Tool 15:** Leadership	**Tool 19:** Refactoring	
	Tool 6: Setbased development			**Tool 16:** Expertise	**Tool 20:** Test	

Systematic's new model for Lean SW development

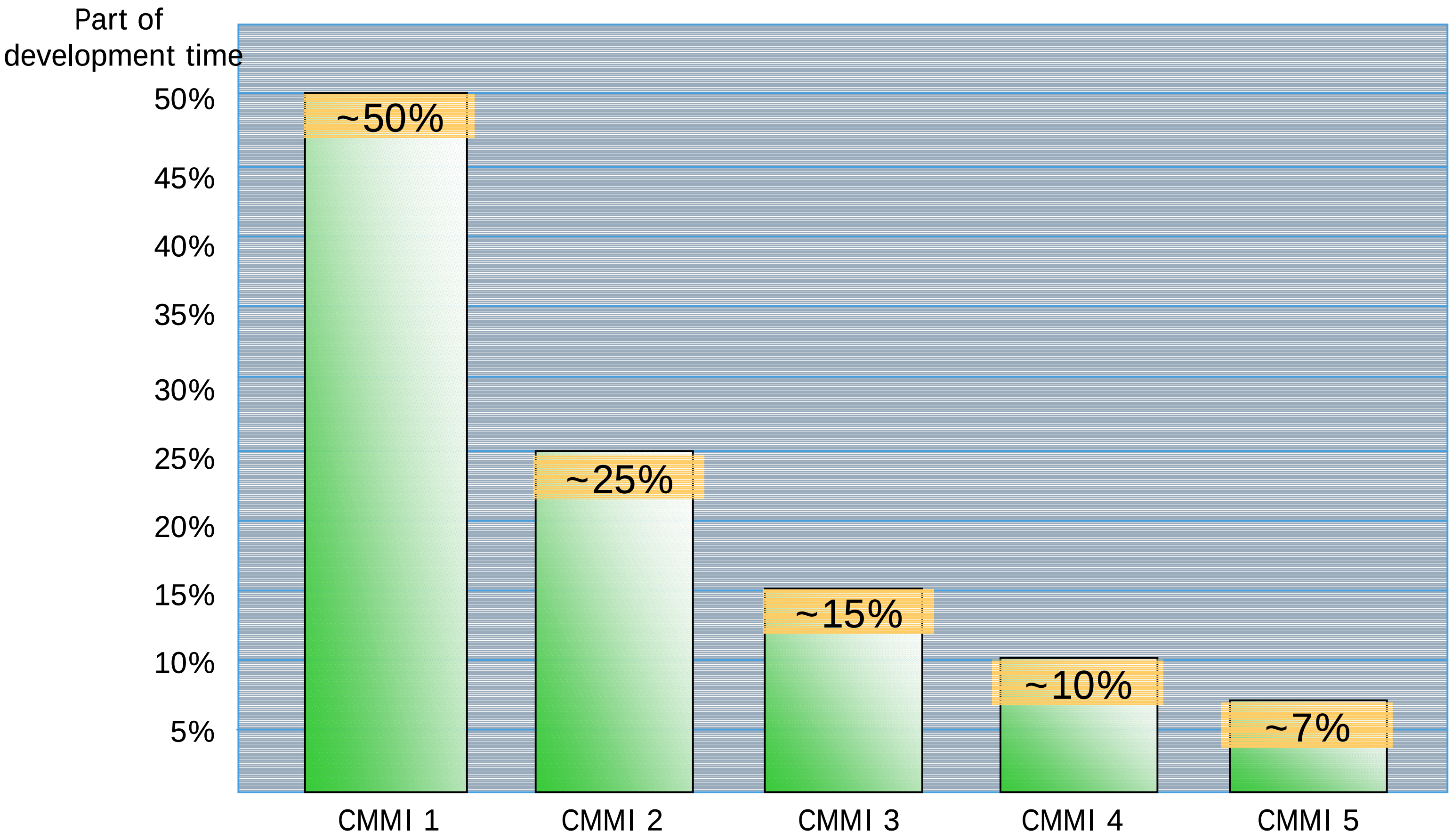


Tools can be divided in three dimensions

	Value	Flow	Pull	Perfection
Engineering	*P6 Build Integrity in* T19 Refactoring T20 Test	*P2 Amplify learning* T5 Synchronization T4 Iterations	*P2 Amplify Learning* T3 Feedback T6 Setbased development	*P6 Build Integrity In* T18 Conceptual integrity T17 Perceived integrity
Management	*P1 Eliminate Waste* T1 Eliminate Waste T2 Valuestreams	*P4 Fast Delivery* T11 Queuing Theory T12 Cost of Delay	*P7 See the whole* T22 Contracts T21 Measures T10 Pull	*P3 Decide in latest Responsible moment* T7 Options thinking T8 Latest responsible Moment T9 Beslutningstagning
People	*P5 Empowerteam* T16 Expertise	*P5 Empower team* T14 Motivation	*P5 Empower team* T15 Leadership	*P5 Empower team* T13 Self-determination

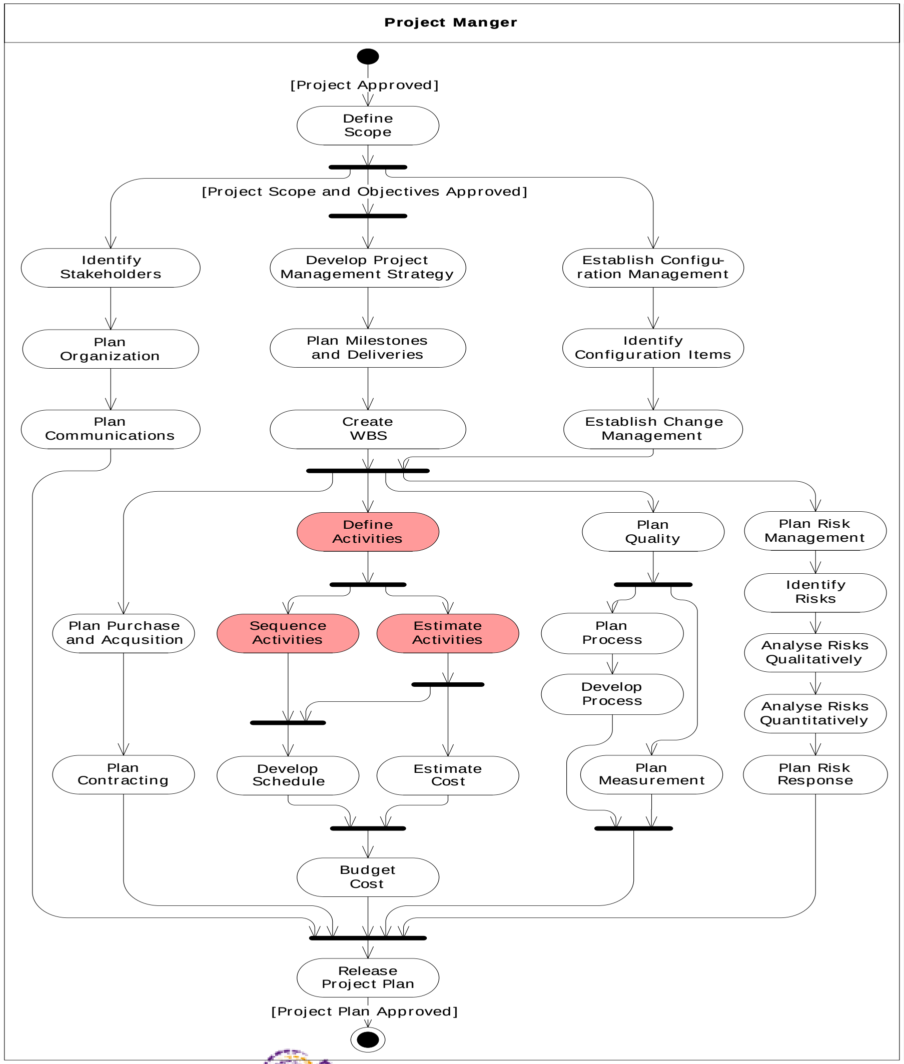
These are thinking tools – Projects and employees knows best how to transform them

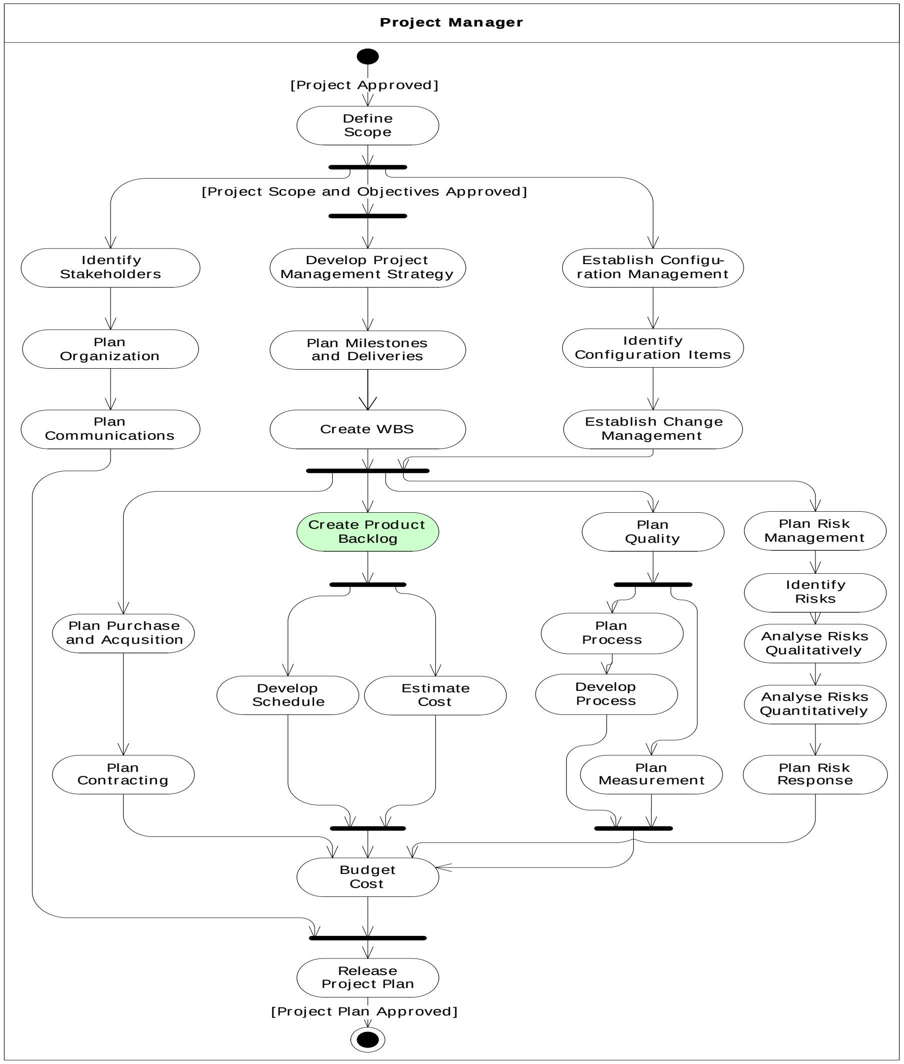
Published experiences with ”rework”



Source: Krasner & Houston, CrossTalk, Nov 1998
Diaz & King, CrossTalk, Mar 2002

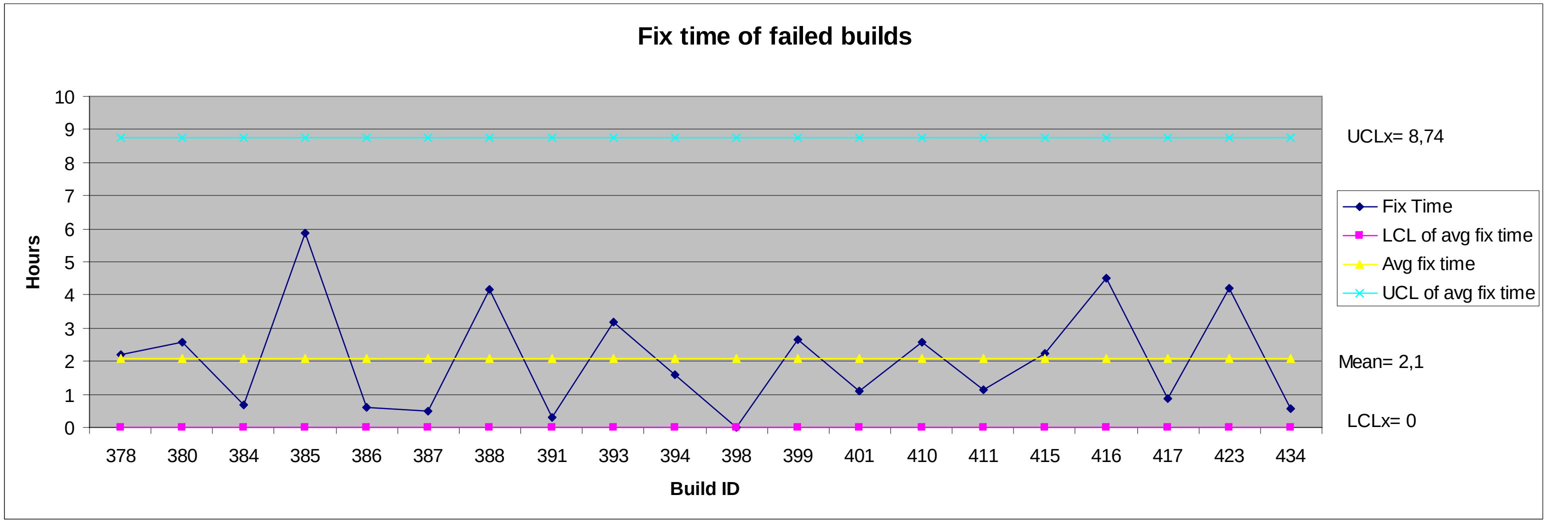
SCRUM and PDP-Common





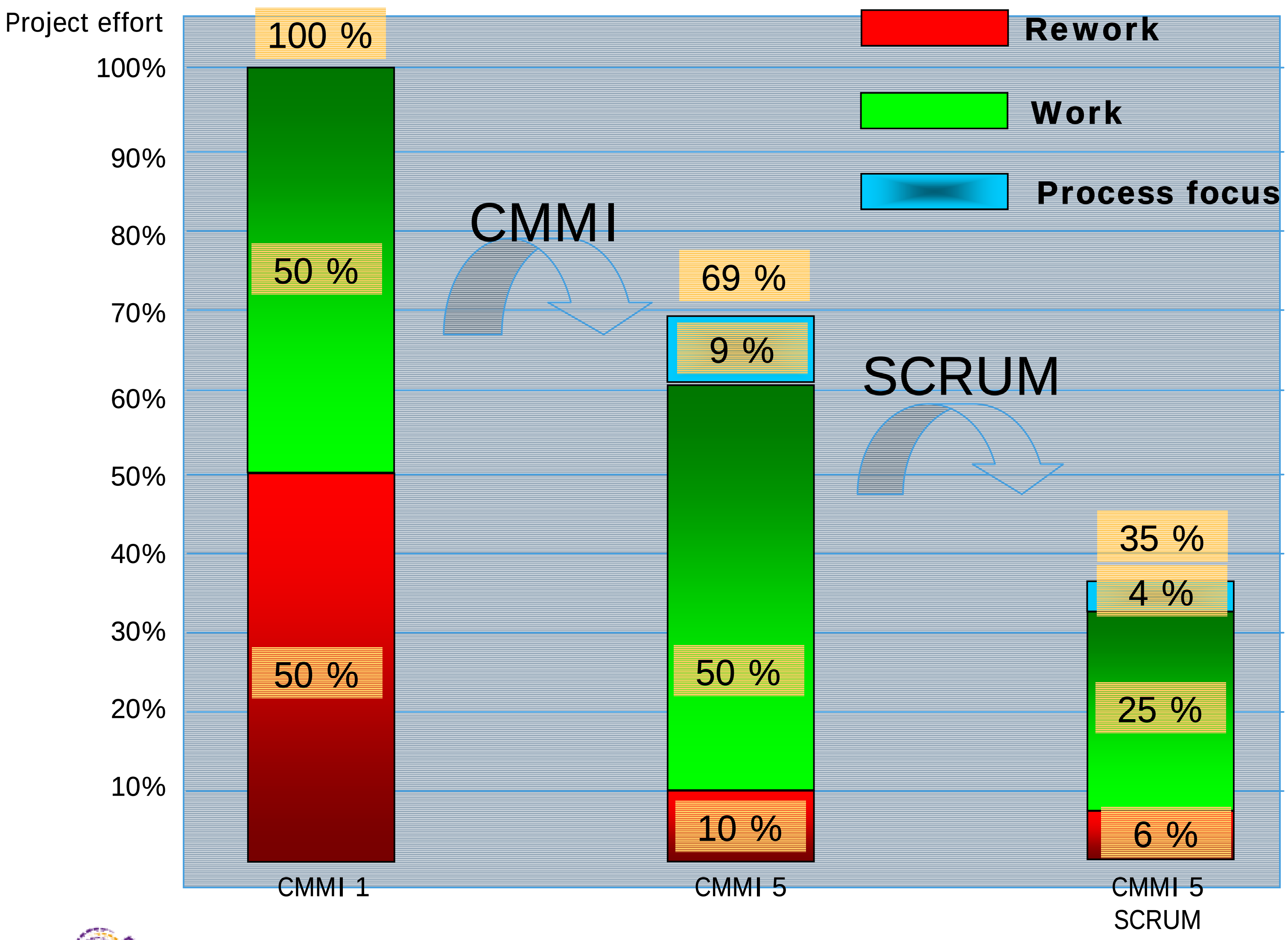
Control chart of build process

Build errors
Unit test errors
FXCop failed



Scrum applied to CMMI Level 5 company

– *6 month results*



© Jeff Sutherland 1993-2007

Systematic CMMI 5 Analysis
First six months of Scrum

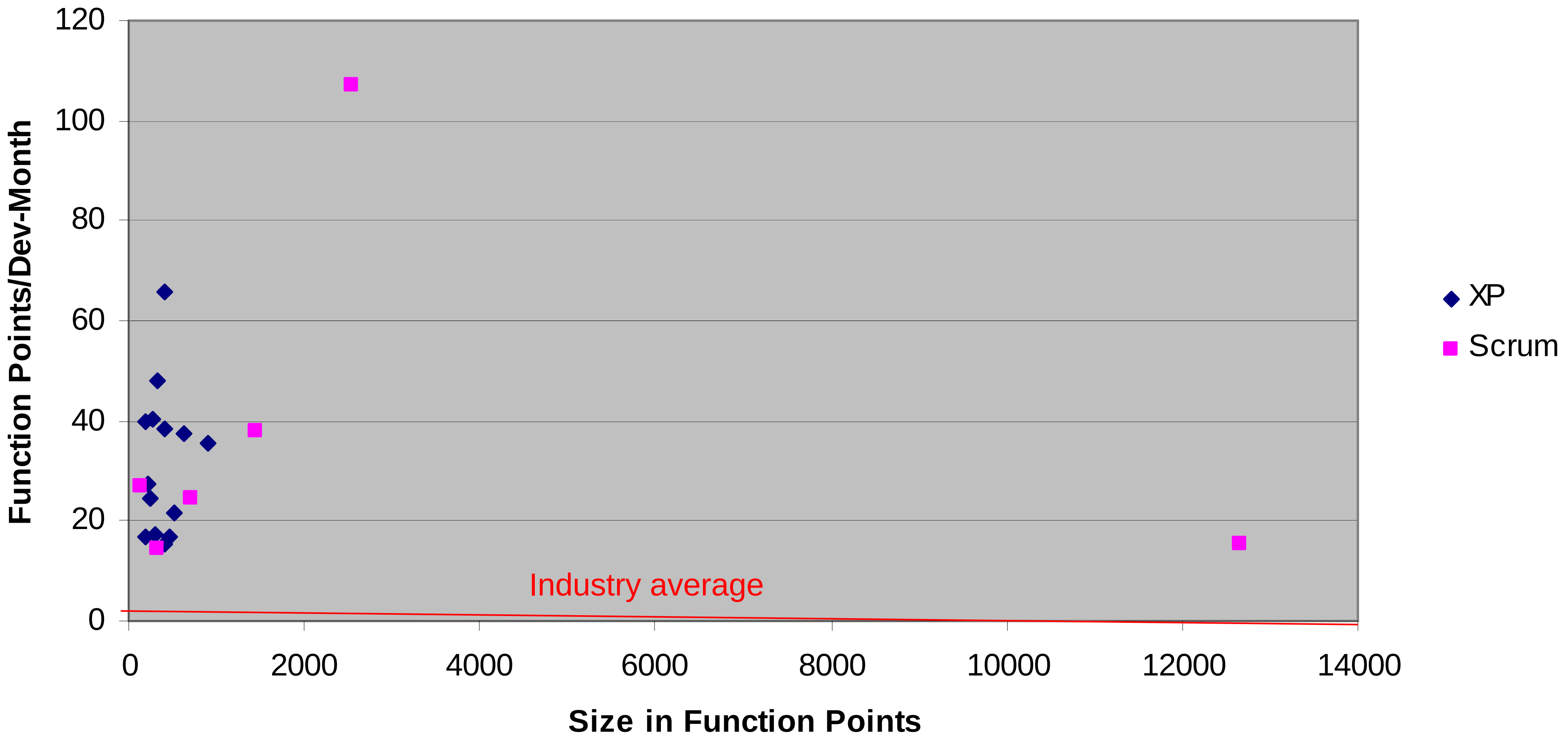
- 80% reduction in planning and documentation costs
- 40% reduction in defects
- 50% reduction in rework
- 100% increase in overall productivity
- Systematic decided to change CMMI Level 5 process to make Scrum the default mode of project management
- When waterfall project management is required, they are now need to be contracted for twice the price of Scrum projects
  - Required by some government agencies
  - Lower business value with fewer features
  - Lower customer satisfaction
  - Lower quality
  - Twice the cost

AGILE2007 Conference

Sutherland, J., C. Jacobson, et al. (2007). Scrum and CMMI Level 5: A Magic Potion for Code Warriors! Agile 2007, Washington, D.C., IEEE.

Exigen Services Productivity

August 2007 Data



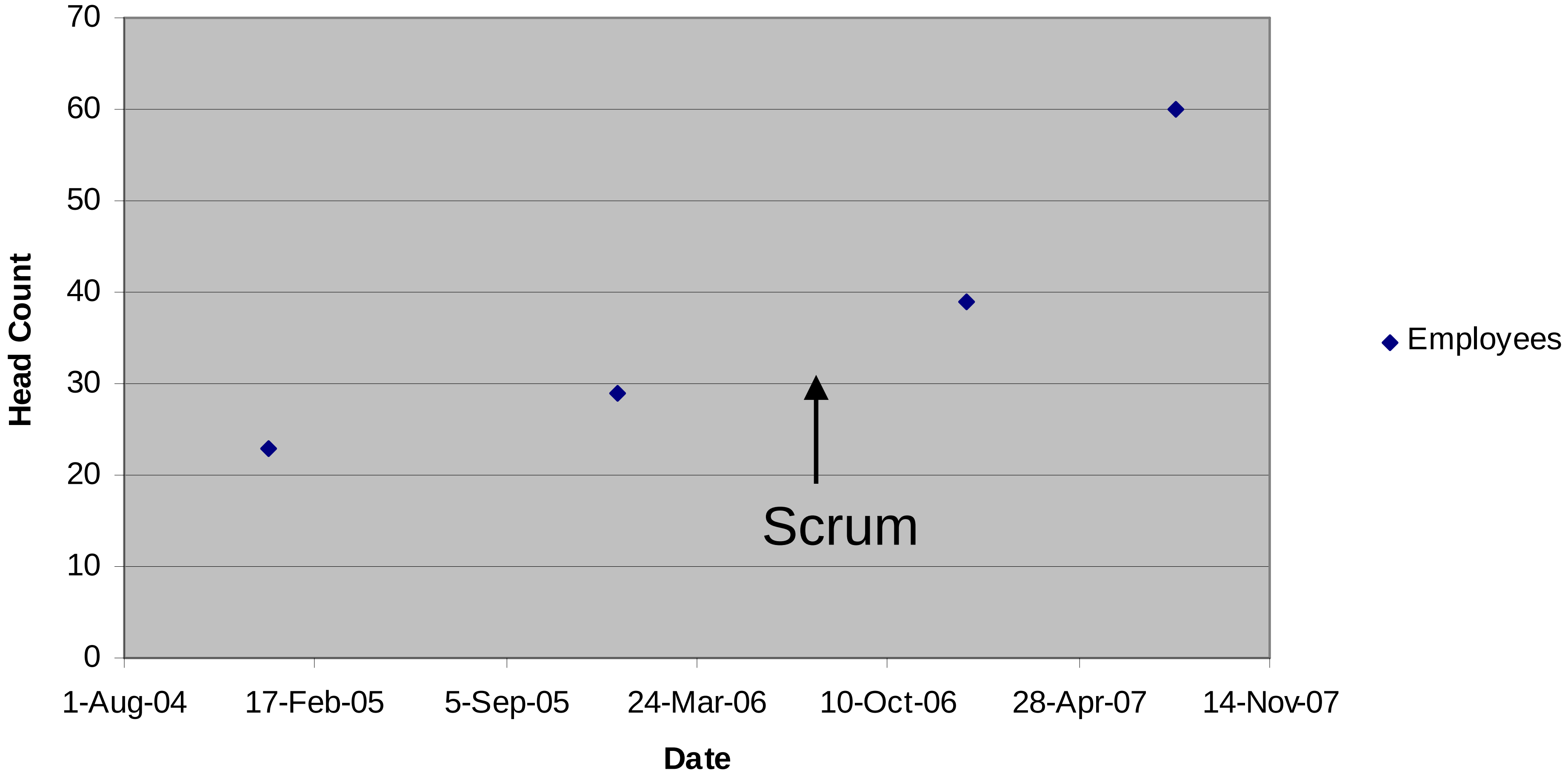
Trifork

JAOO conference 2007

September 23 - 28, Aarhus, Denmark

- Company wide Scrum
- Goals
  - Every employee a Certified ScrumMaster
  - Every part of company run by Scrum
- Sales force are Certified ScrumMasters and only execute Agile contracts
- JAOO run by Scrum – bigger than Agile 2007

Trifork 2004-2007

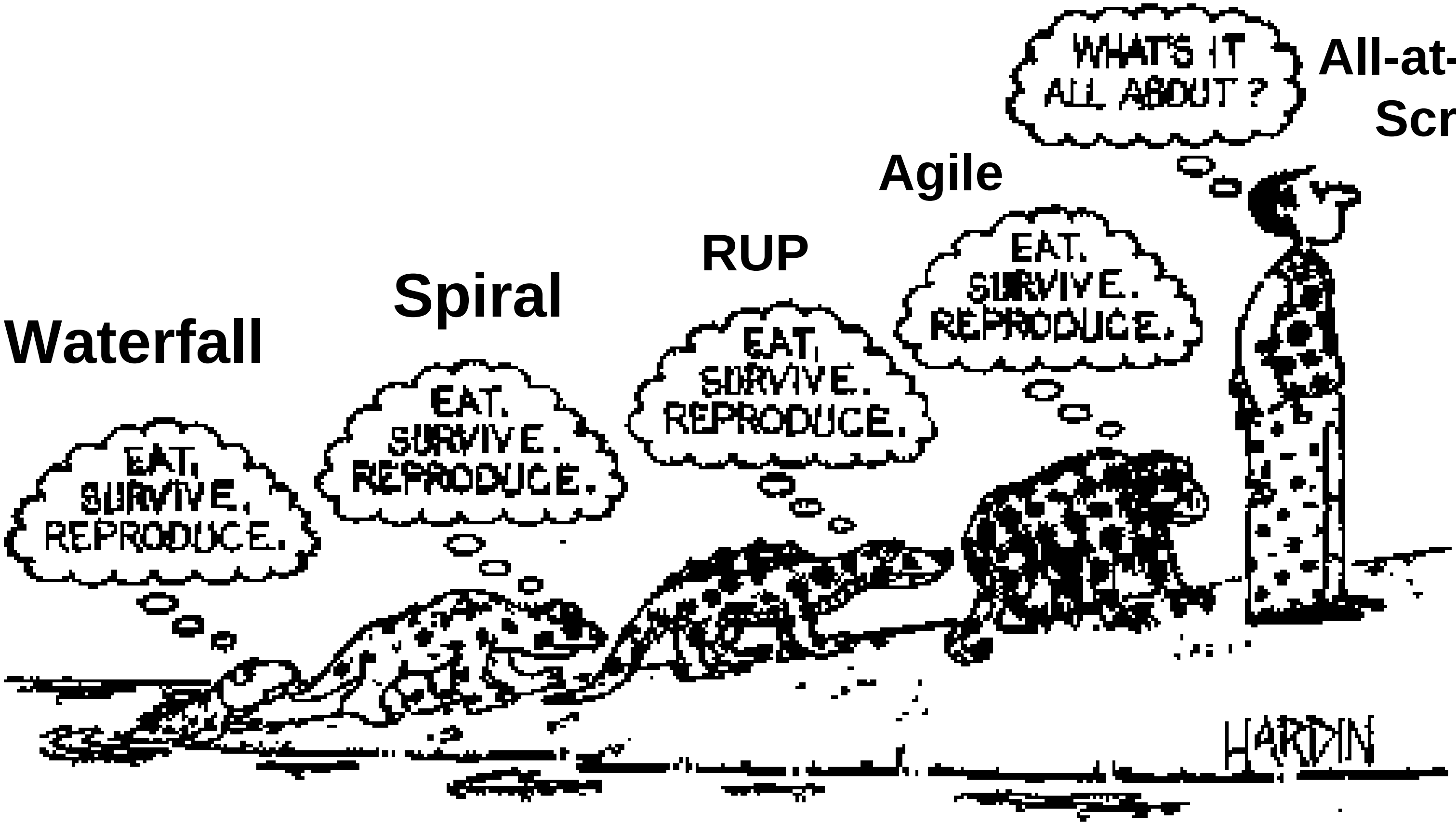


Company Structure

	Coercive	Empowering
Disciplined	**Bureaucracy** •Rigid rule enforcement •Extensive written rules and procedures •Hierarchy controls	**Leadership** •Empowered employees •Rules and procedures as enabling tools •Hierarchy supports organizational learning
Whimsical	**Autocracy** •Top down control •Minimum rules and procedures •Hierarchy controls	**Organic** •Empowered employees •Minimum rules and procedures •Little hierarchy

Adapted from Liker, JK (2004) The Toyota Way. McGraw Hill.

Climbing out of the tar pit ...



Waterfall

Spiral

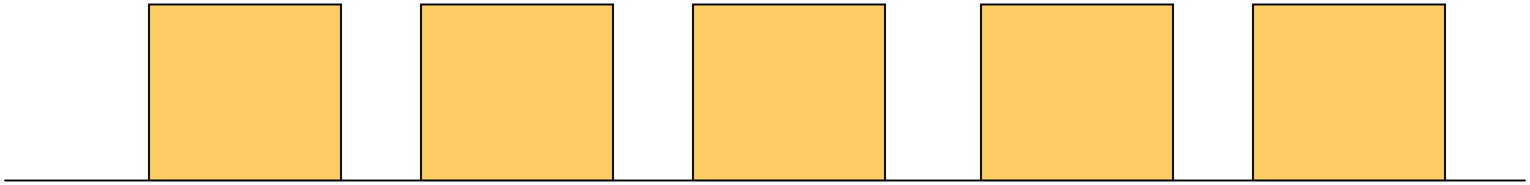
RUP

Agile

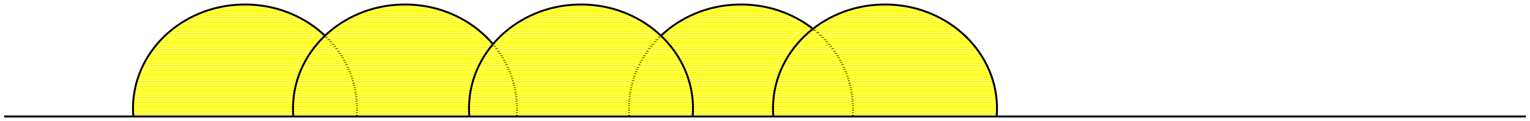
All-at-Once Scrum

Theory: Scrum Evolution

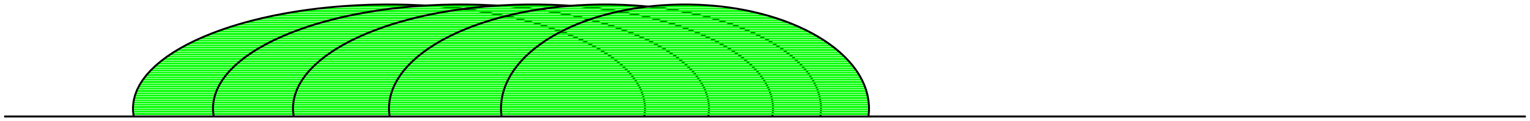
Type A, B, C Sprints



Type A – Isolated cycles of work



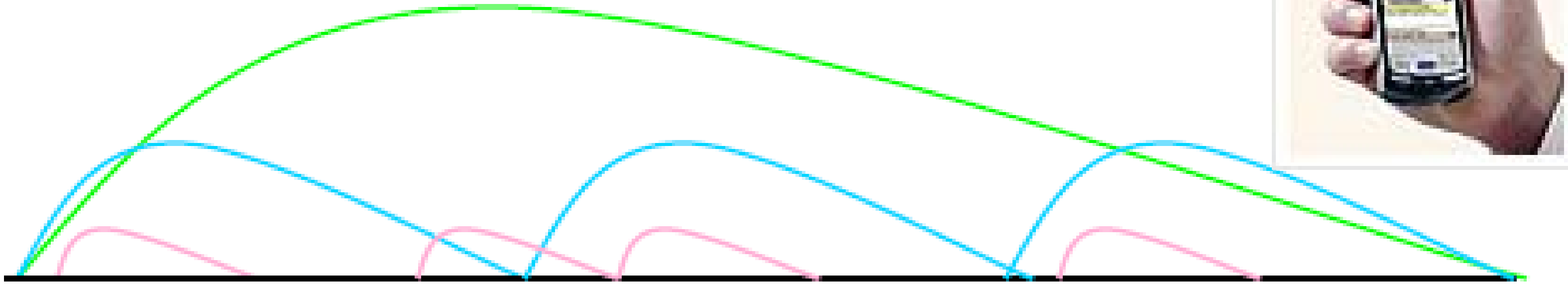
Type B – Overlapping iterations



Type C – All at once

The overlapping of phases does away with traditional notions about division of labor. Takeuchi and Nonaka (1986)

Simultaneous Overlapping Sprints

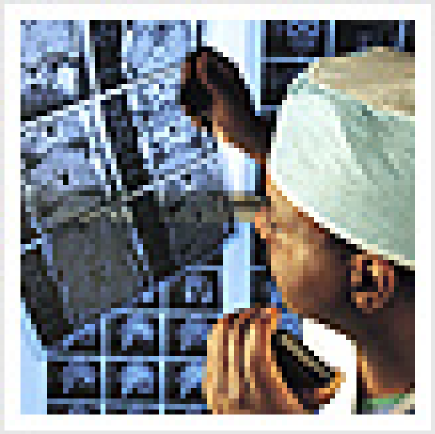


Red - weekly
Blue - monthly
Green - quarterly

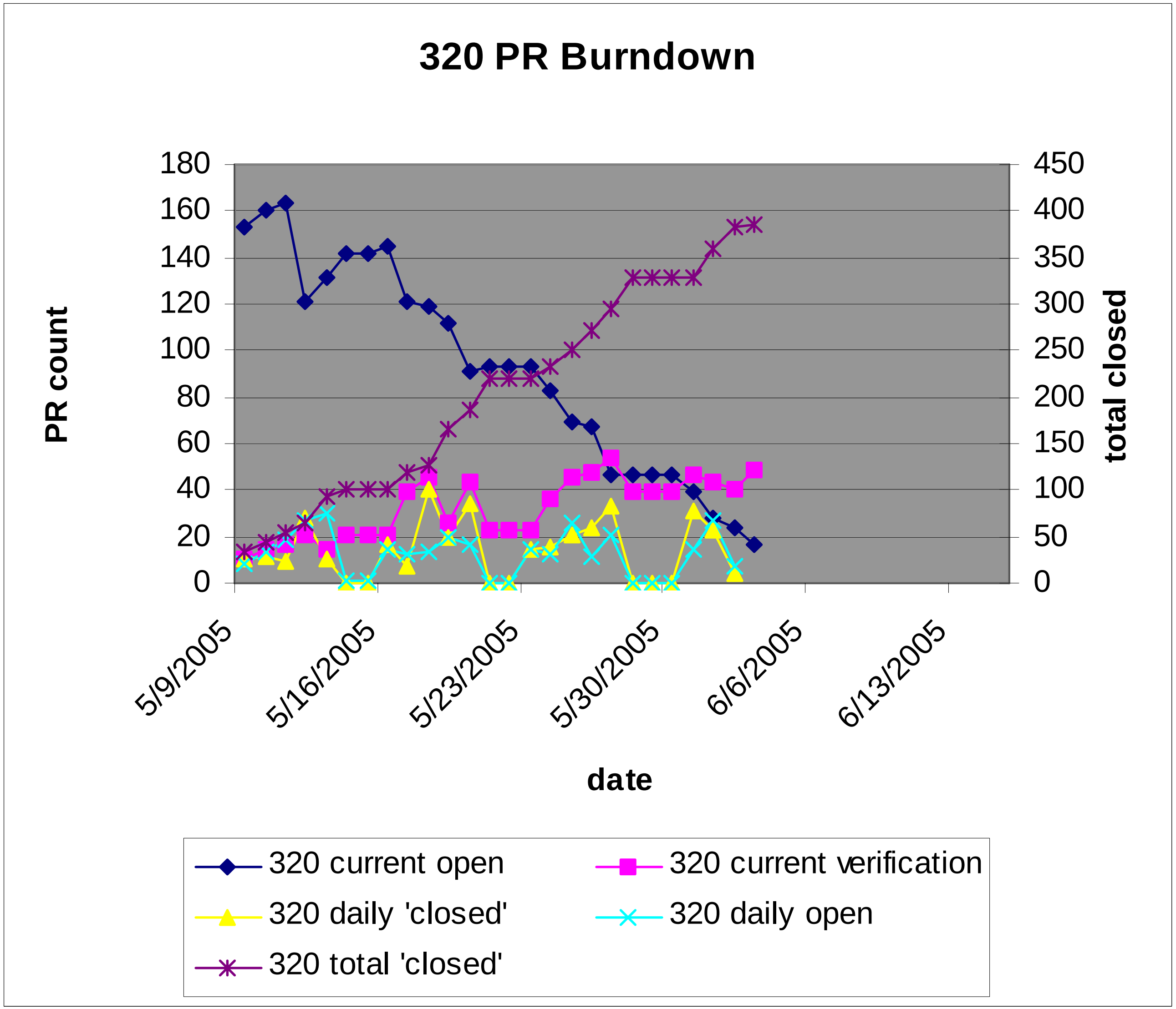
PatientKeeper delivers 45 production releases of quality code to tens of thousands of physician users in large healthcare systems every year. Largest client is HCA with 176 hospitals.

PatientKeeper Strategy for Done, Done, Done, Done

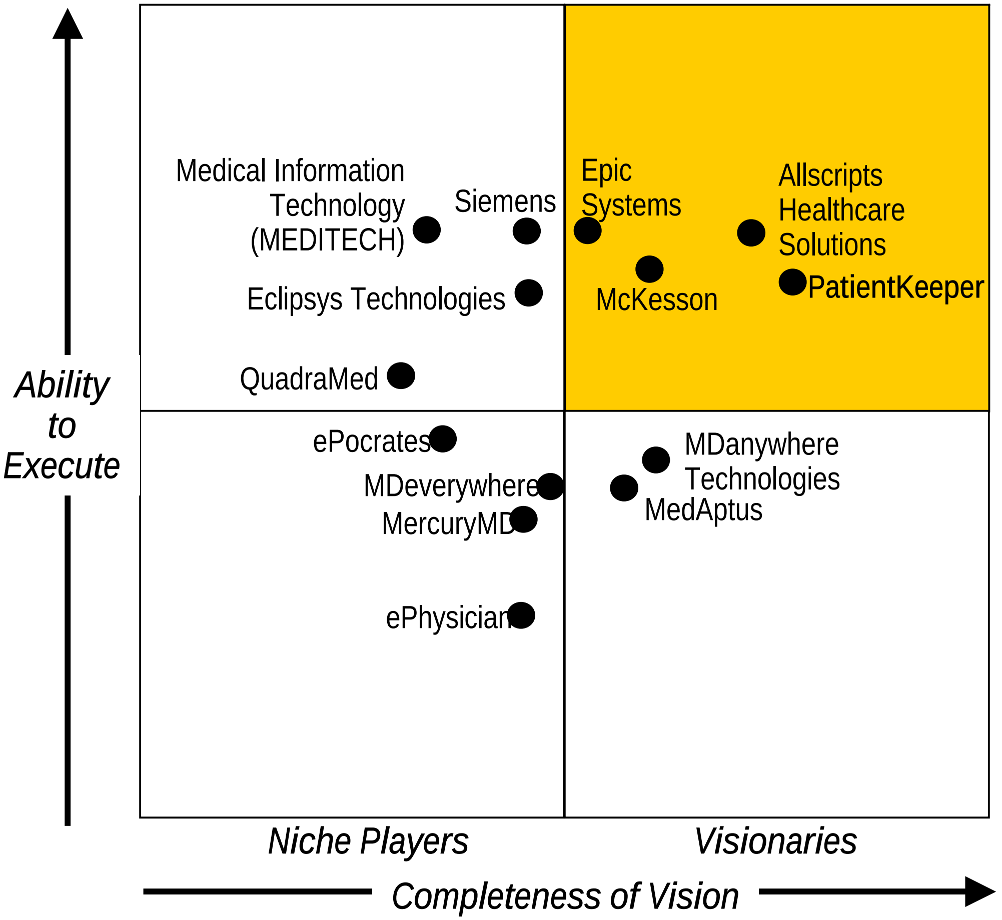
- Sprint Planning
  - Product Backlog must be “ready” (estimable and testable)
- Week 1
  - Develop Agile technical specifications
  - Testers update test plans with delta
  - Developers put code in build
  - Smoke test ensures build never fails
  - Testers immediately test completed stories
- End of Week 2
  - Install Release Candidate 1 at customer sites
- End of Week 3
  - Install Release Candidate 2 at customer sites
- End of Week 4
  - Complete all customer requests, bugs, testing
  - Install final Release Candidate at customer sites
- Last day of Sprint
  - Train users and go live!
- Sprint Review
  - Did customers complain/rejoice?



PatientKeeper Burndown



PatientKeeper All-at-Once Scrum



- First Scrum showed hyperproductive software development - NOT revenue
- PatientKeeper first company to achieve hyperproductive revenue state driven by Scrum in July 2007
- All-at-Once Type C Scrum designed for hyperproductive software *AND* revenue

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**

Sutherland, J., A. Viktorov, and J. Blount. **Adaptive Engineering of Large Software Projects with Distributed/Outsourced Teams.** In *Proceedings of the International Conference on Complex Systems*. 2006. Boston, MA, USA.