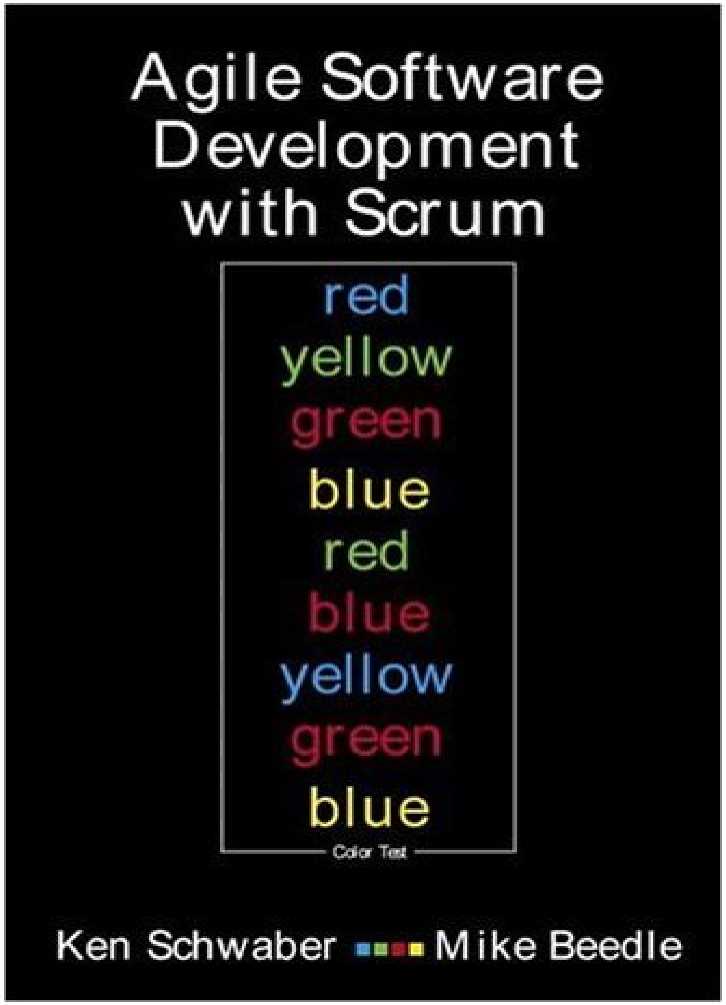
Scrum with XP and Beyond

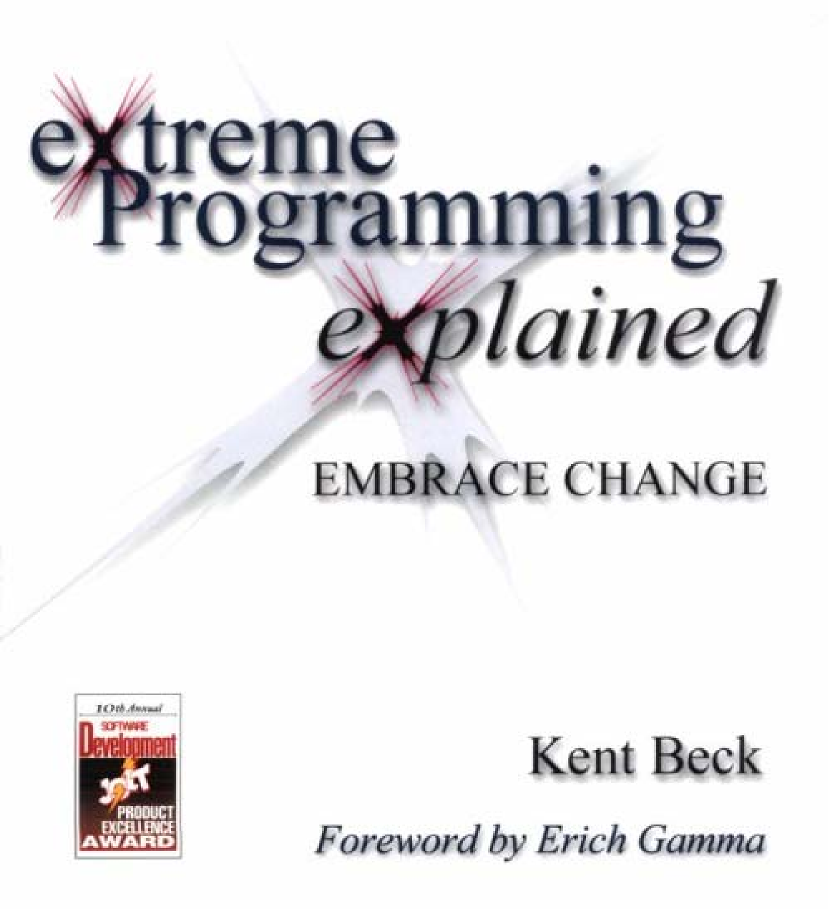


Agile Software Development with Scrum

red
yellow
green
blue
red
blue
yellow
green
blue

Color Test

Ken Schwaber ■■■■ Mike Beedle



eXtreme Programming eXplained

EMBRACE CHANGE

10th Annual Software Development Jolt Product Excellence Award

Kent Beck

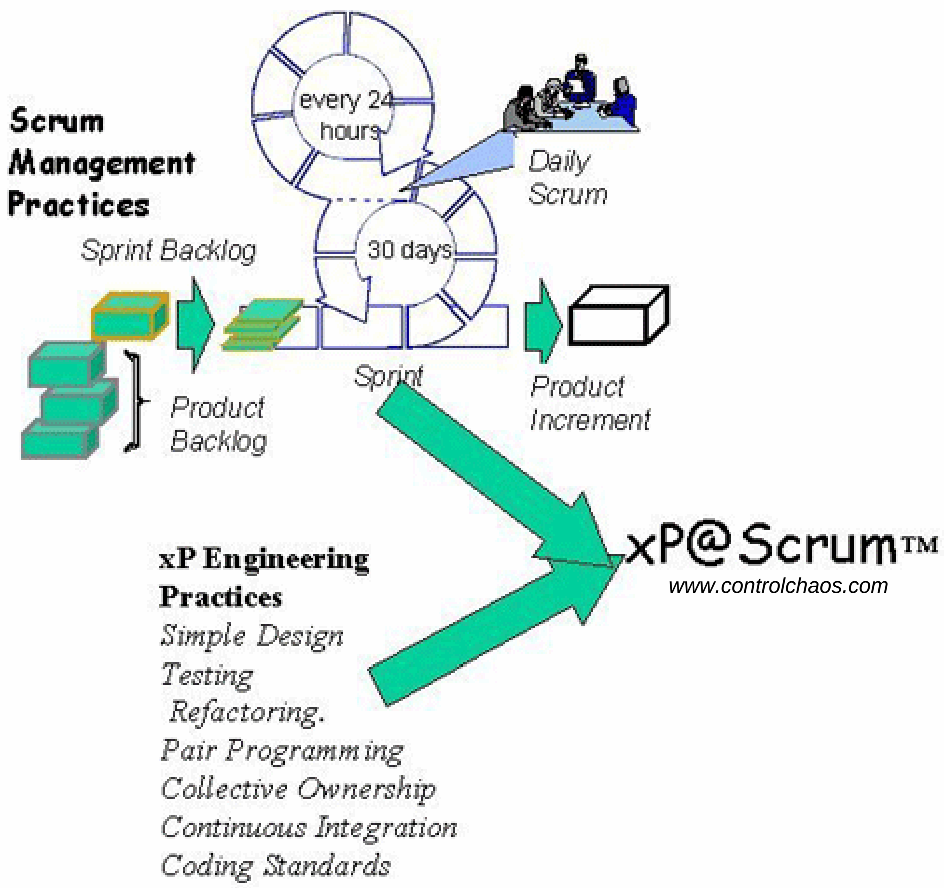
Foreword by Erich Gamma

Deep Agile
Software Development

xP@Scrum

- Scrum as a management wrapper for XP engineering practices.
- Scrum provides the agile management mechanisms
- Extreme Programming provides the integrated engineering practices.

Sliwa, Carol. XP, Scrum Join Forces. ComputerWorld, 18 Mar 2002



Benefits of Scrum and XP together

- Management and control mechanisms of Scrum are applicable for any type of project
  - multiple, simultaneous software development initiatives
  - business development, re-engineering, marketing, support, and implementation projects
  - teams are iteration (or Sprint) goal directed, rather than story directed.
- XP projects wrapped by Scrum become scalable and can be run simultaneously by non-colocated teams.
- Scrum implements in a day; XP can be gradually implemented within the Scrum framework.
- Scrum with XP has demonstrated linear scalability on distributed, outsourced projects and CMMI Level 5 projects
- The most productive large project (>1,000,000 lines of Java code) ever documented is a Scrum and XP project

Critical practices for Scrum productivity gains

- Early testing
- Continuous integrating
- Constant refactoring
- Simple design
- Some pair programming – the most experienced XP companies do pair programming about 50% of the time
- Evolving the code base

The first Scrum at Easel Corporation used all XP practices

- Continuous builds were there before Scrum began
- If no code, generate some code in a week, and iterate on that code
- Pair programming with a mentor would often eliminate 75% of the code base in a morning session and result in radically new design
- One week of nothing but refactoring for entire team in every iteration
- Testing (and documentation) happened the first day and every day of each iteration
- Common ownership of code with coding standards
- Some practices went way beyond XP and conventional Scrum
  - Test engineers built probes for component frameworks – similar to testing chips – test first and ensure reusable components
  - Set Based Concurrent Engineering

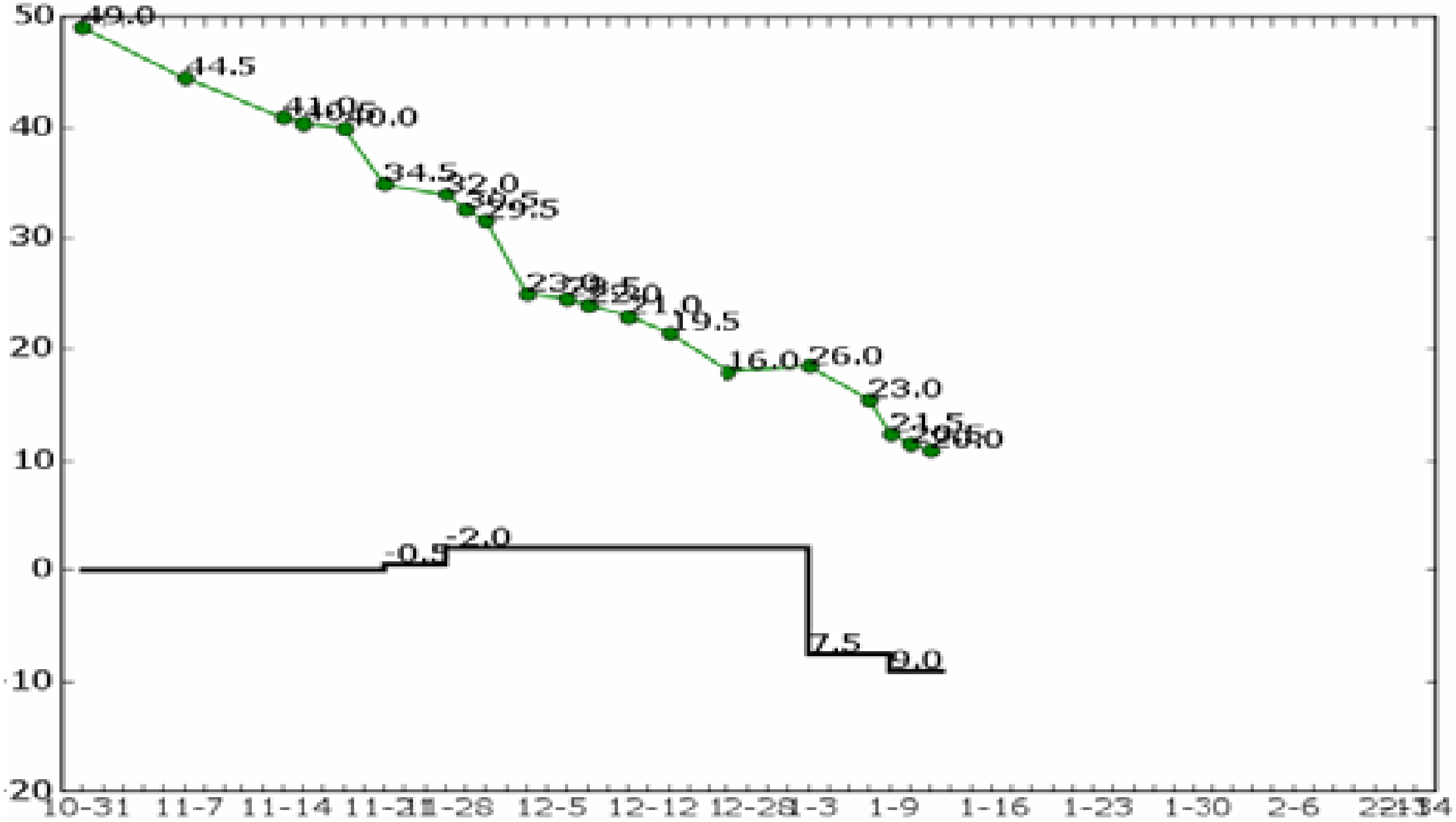
Hyperproductivity in the First Scrum

- Productivity 5-10 times industry average has been observed in many Scrums since 1993.
- Factors accelerating the first Scrum
  - Scrum organizational pattern
  - XP engineering practices
  - Stimulating software evolution
    - Emergent architecture
    - Set-Based Concurrent Engineering (SBCE)

First Scrum prioritized the Sprint backlog by business value

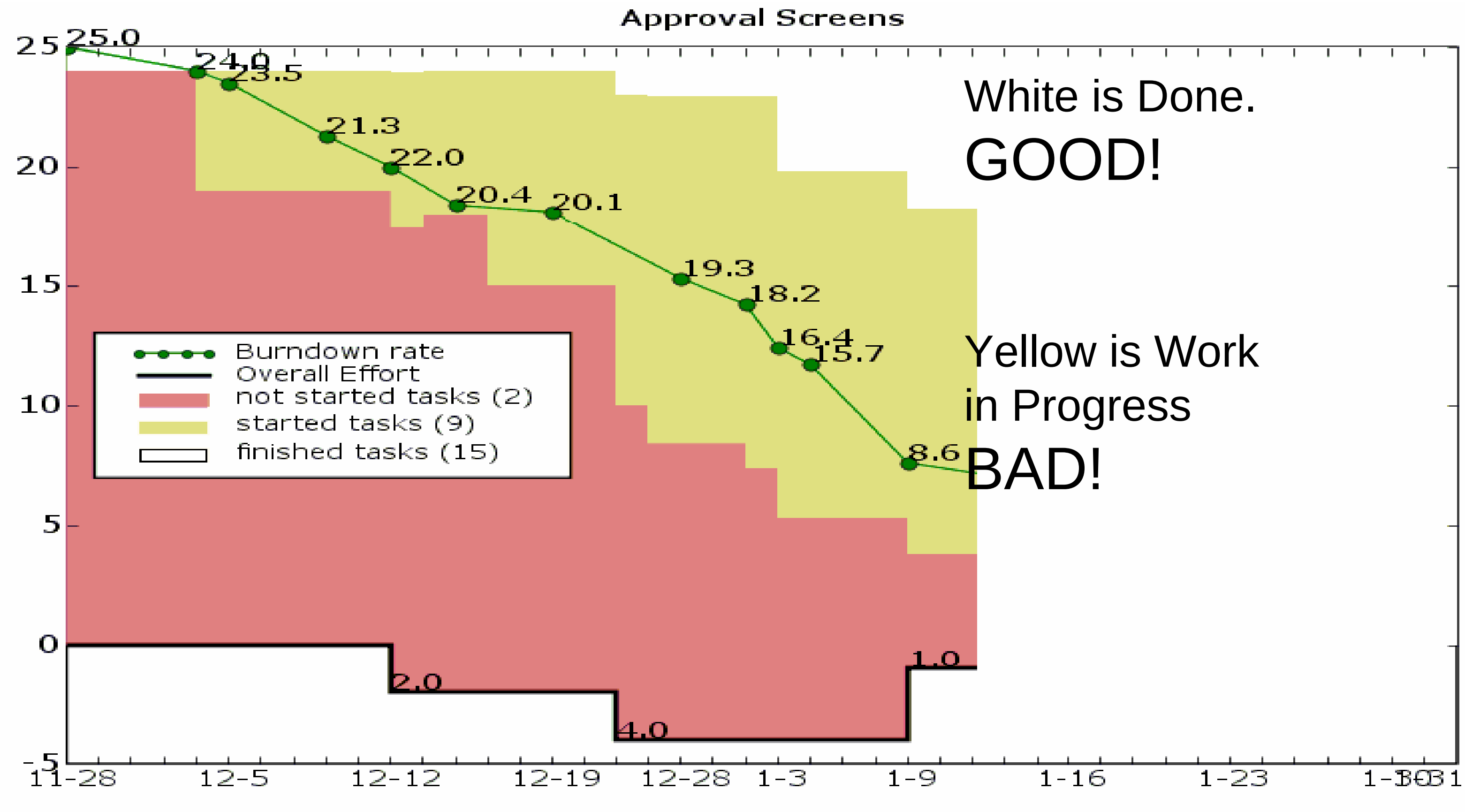
- The team asks every day before any developer started a new task:
  - What task will maximize the speed of appearance of a new feature?
  - Will it maximize the speed of appearance of a new feature by implementing it in a new evolving component worked on by anyone on the team?
  - Will it maximize the speed of appearance of a new feature if it is done in a completely new way not previously thought of?

Google Release Burndown Chart



© Jeff Sutherland 1993-2007

Google reinvents Lean:
Showing “work in progress”



White is Done.

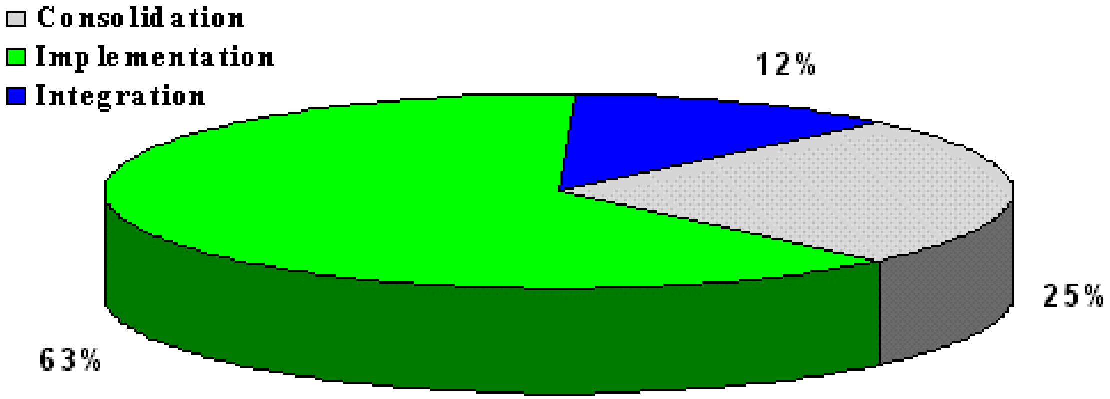
GOOD!

Yellow is Work in Progress

BAD!

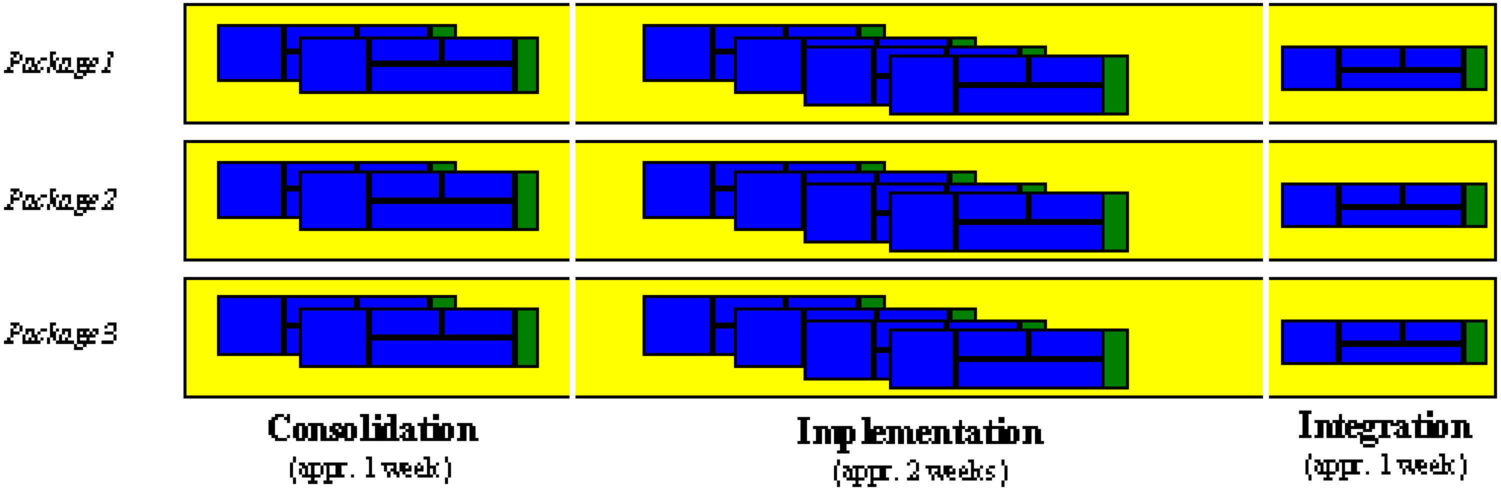
Ssh! We are adding a process… Agile 2006, Minneapolis

Sprint Monthly Cycle: First Scrum



- Three phases
  - Consolidation
  - Implementation
  - Integration.
- Output is next iteration of a production prototype

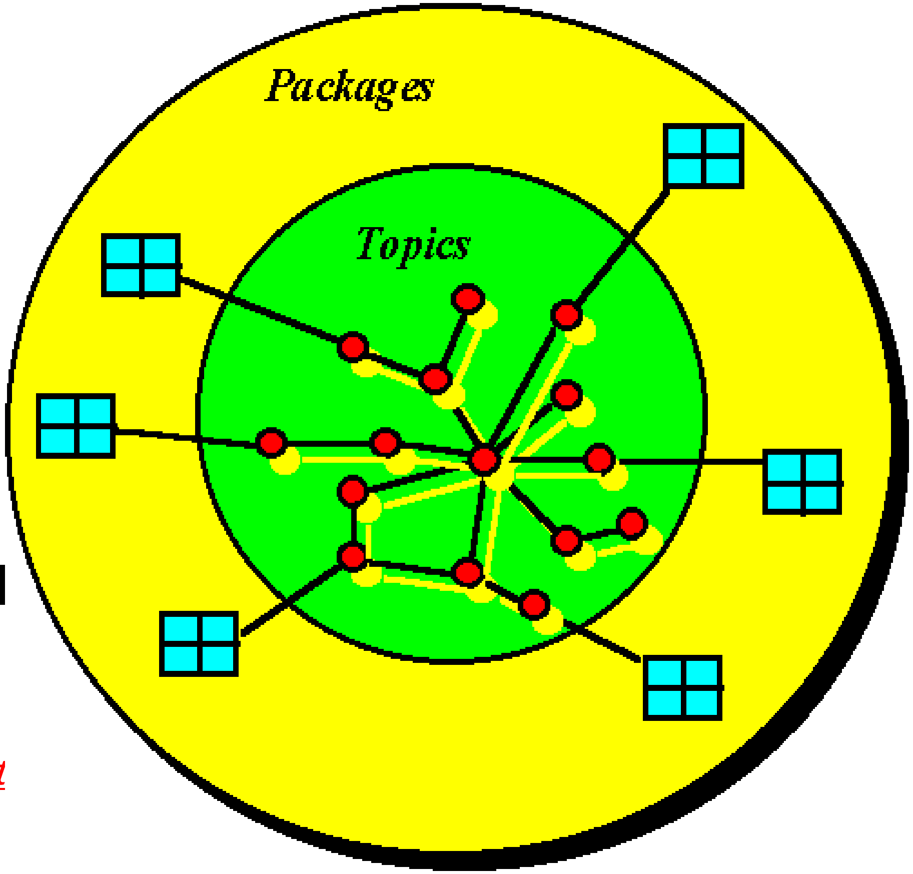
Overlapping Development Phases



- **Incremental production prototypes (monthly)**
- **Incremental package delivery (dynamic)**

Project Domain

- **A project domain is a set of packages that will form a release.**
- **Packages evolve out of work on topic areas.**
- **Rapid evolution can produce "punctuated equilibrium" effect yielding dramatic resul in unexpected time frames.**



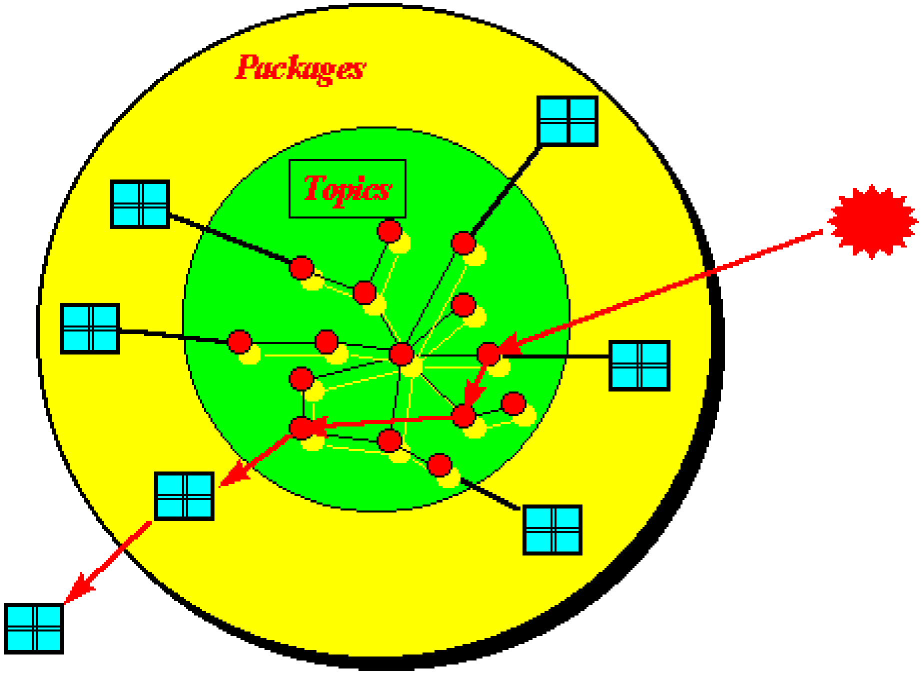
Levy, Steven. Artificial Life: A Report from t Frontier Where Computers Meet Biology. Vintage Books, 1993. (See notes on the simulation by Daniel Hillis of punctuated equilibrium on a Thinking Machine highly parallel computer.)

SynchStep Firing

- SynchSteps are individual tasks in a topic areas.
- Work on topic areas results in a package ready to be put in a software release.

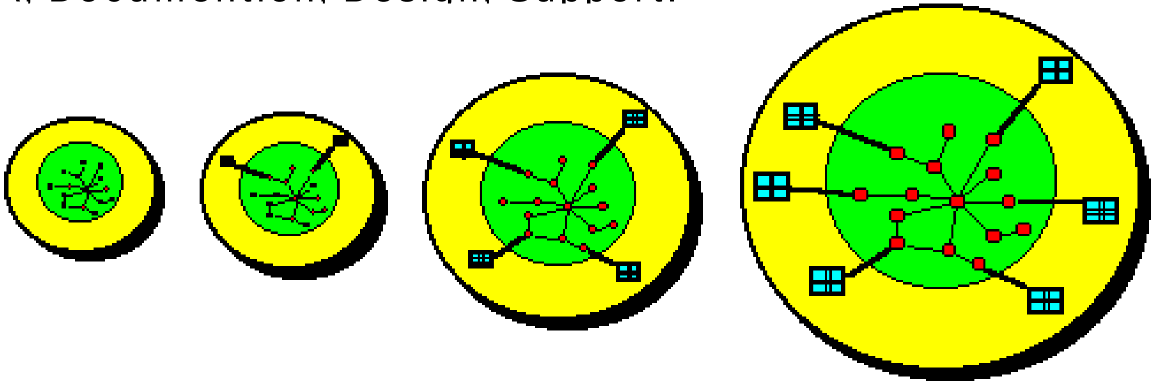
Example packages for visual tool

- Component Builder
  - Scenario editor
  - Event editor
  - Ensemble diagrammer
- Workgroup Support
  - Persistent object repository
  - Version control
  - Multi-user access



SynchStep Delivery Strategy

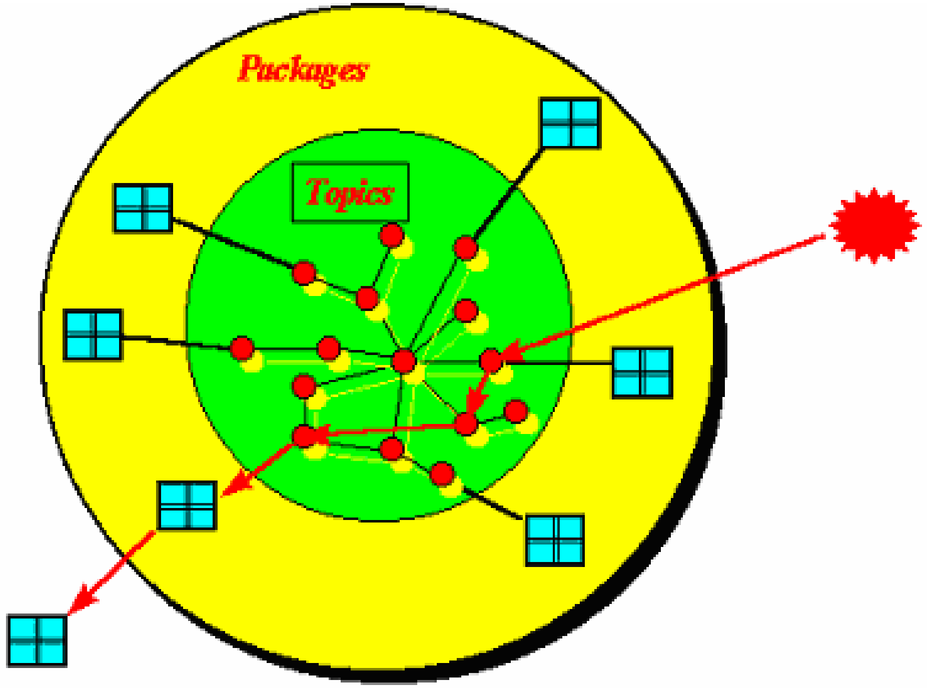
High performance production team where engineers had to be outnumbered 2-1 by QA, Documention, Design, Support.



- Example: two packages per month for team of 12 (4 engineers maximum)
- AlphaBeta delivery
  - First iteration of a package is alpha
  - Next iteration package must be beta quality
  - Third iteration package is production quality (frozen)
- Release is announced by Product Owner
  - Production packages always ship at end of every iteration.
  - When enough production packages are shipped, call it a release.

Emergent Architecture – SBCE

(Set Based Concurrent Engineering)

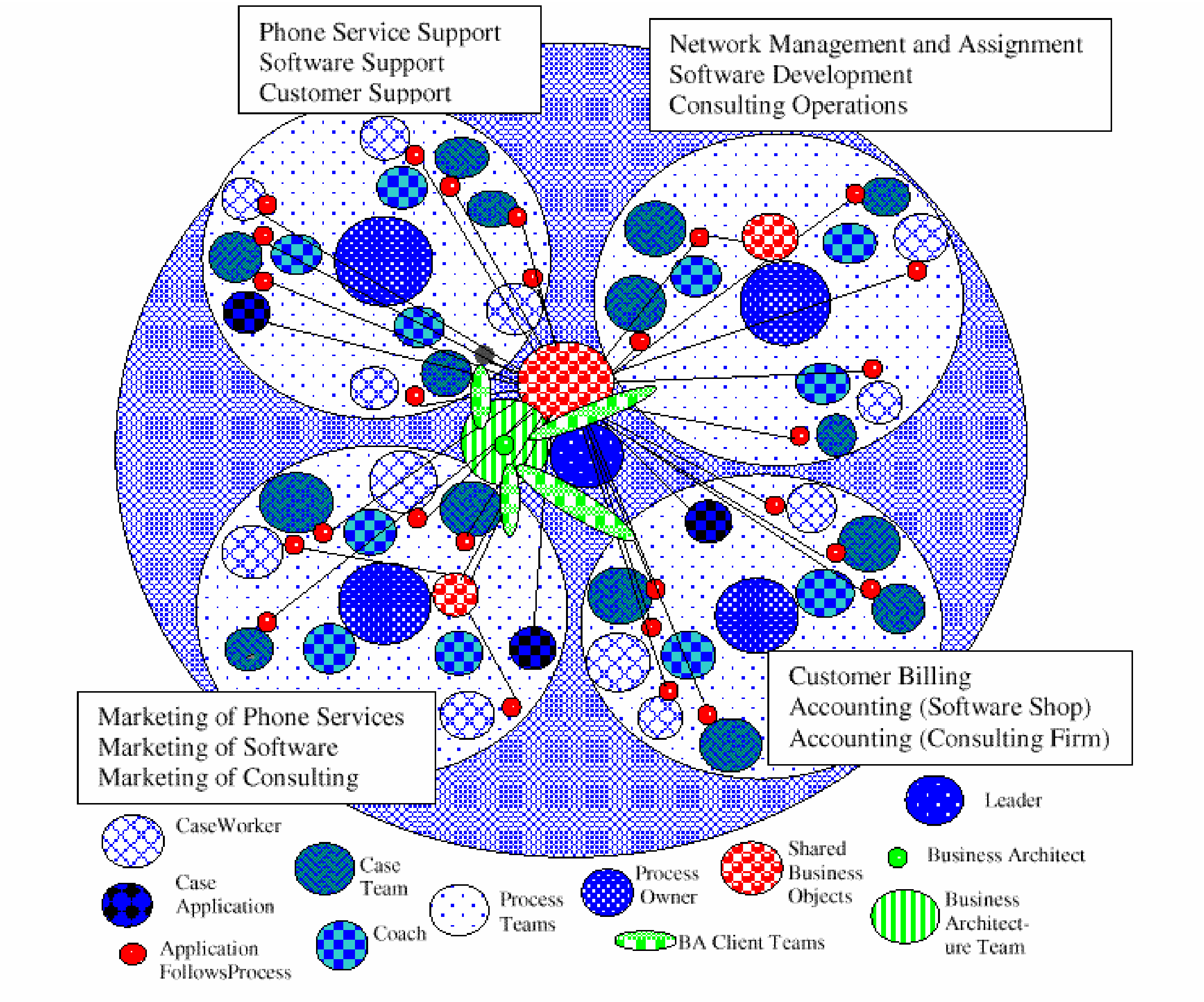


SyncStep
(Sprint task)

Agile Enterprise (formerly Xbreed)

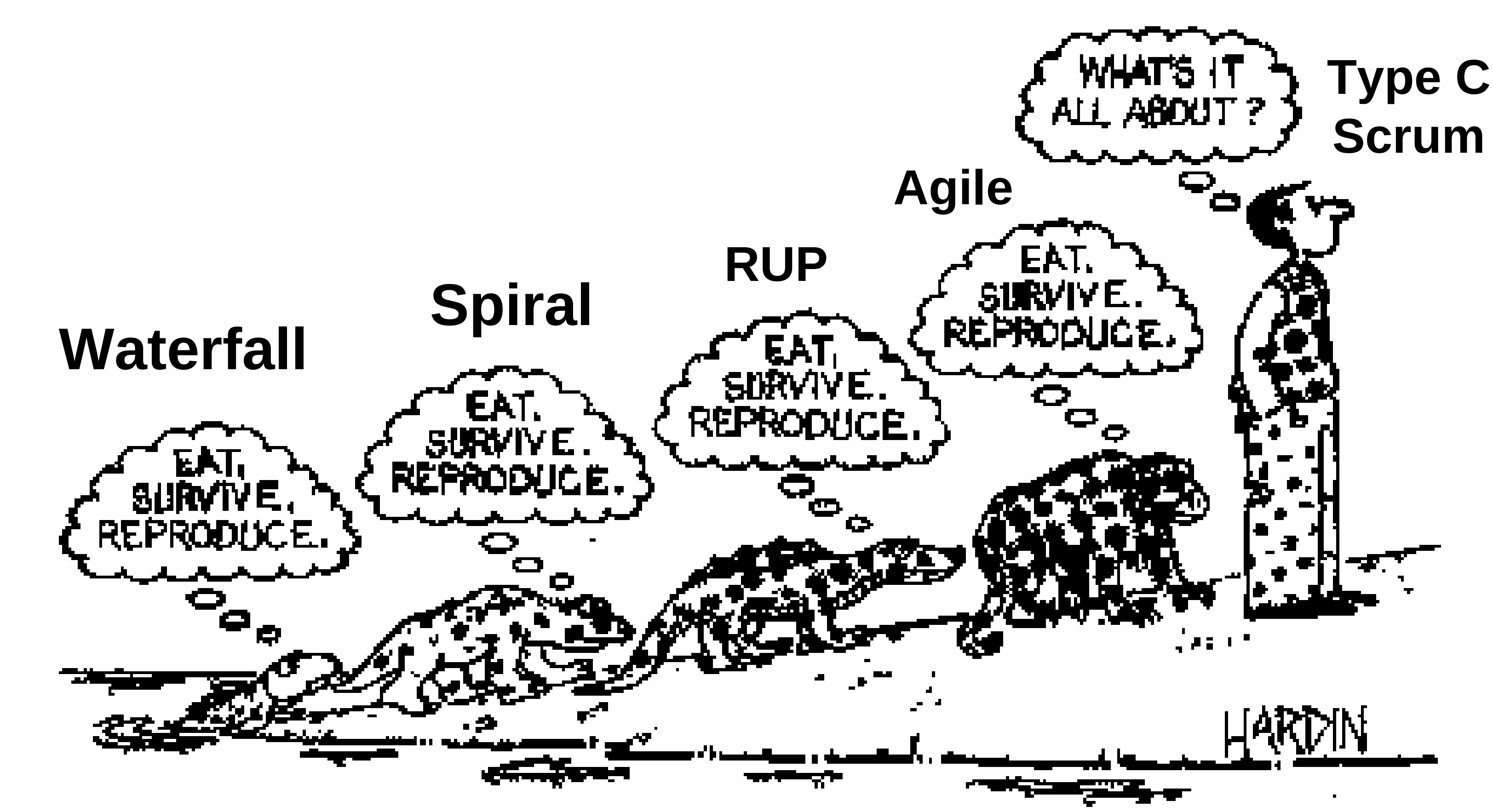
- A business builds and deploys ***new business processes*** to shorten work-cycles, lower operational costs, satisfy regulatory constraints, etc.
- Multiple business processes must be inspected, deployed, re-architected and monitored all-at-once, (Sashimi style), all of them which may encompass one or more enabling applications.
- Scrum – as a process that drives ***prioritized change***, at the business level (including all enabling applications) is the foundation for business improvement.
- Scrum at the business level, allows for the deployment of new business processes that deploy business goals, regulatory requirements, mission and vision objectives, and/or keep process initiatives – ***with or without enabling applications.***

Take Scrum beyond Scrum and XP to the Business



© Jeff Sutherland 1993-2007

Climbing out of the tar pit …



Waterfall

Spiral

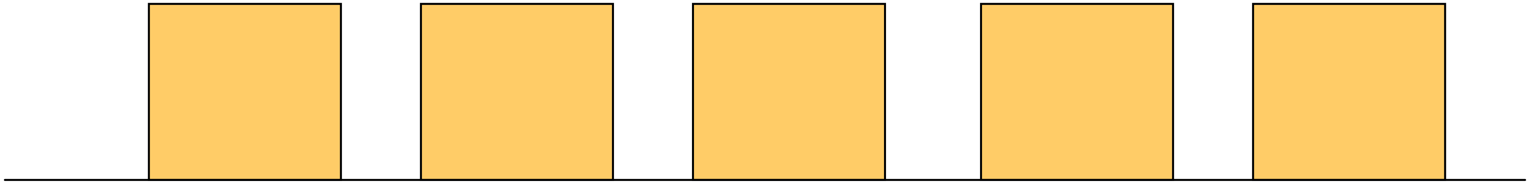
RUP

Agile

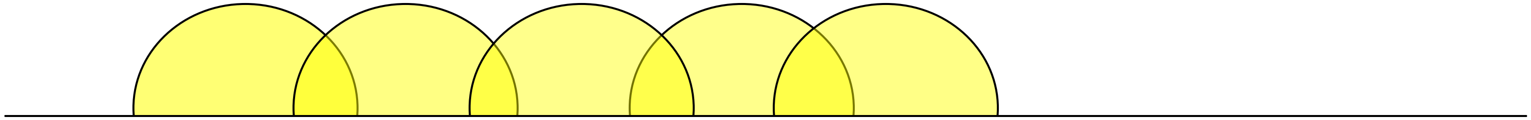
Type C
Scrum

Theory: Scrum Evolution

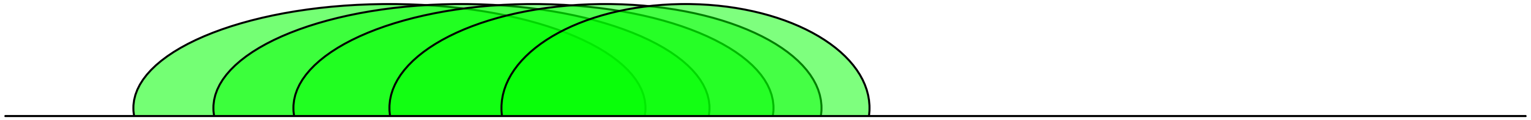
Type A, B, C Sprints



Type A – Isolated cycles of work



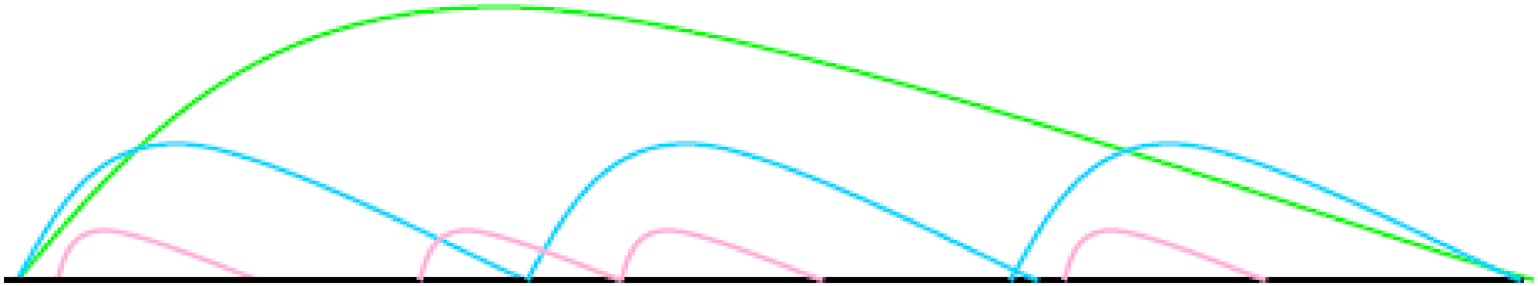
Type B – Overlapping iterations



Type C – All at once

Sutherland, J. (2005). Future of Scrum: Parallel Pipelining of Sprints in Complex Projects. AGILE 2005 Conference, Denver, CO, IEEE.

Simultaneous Overlapping Sprints



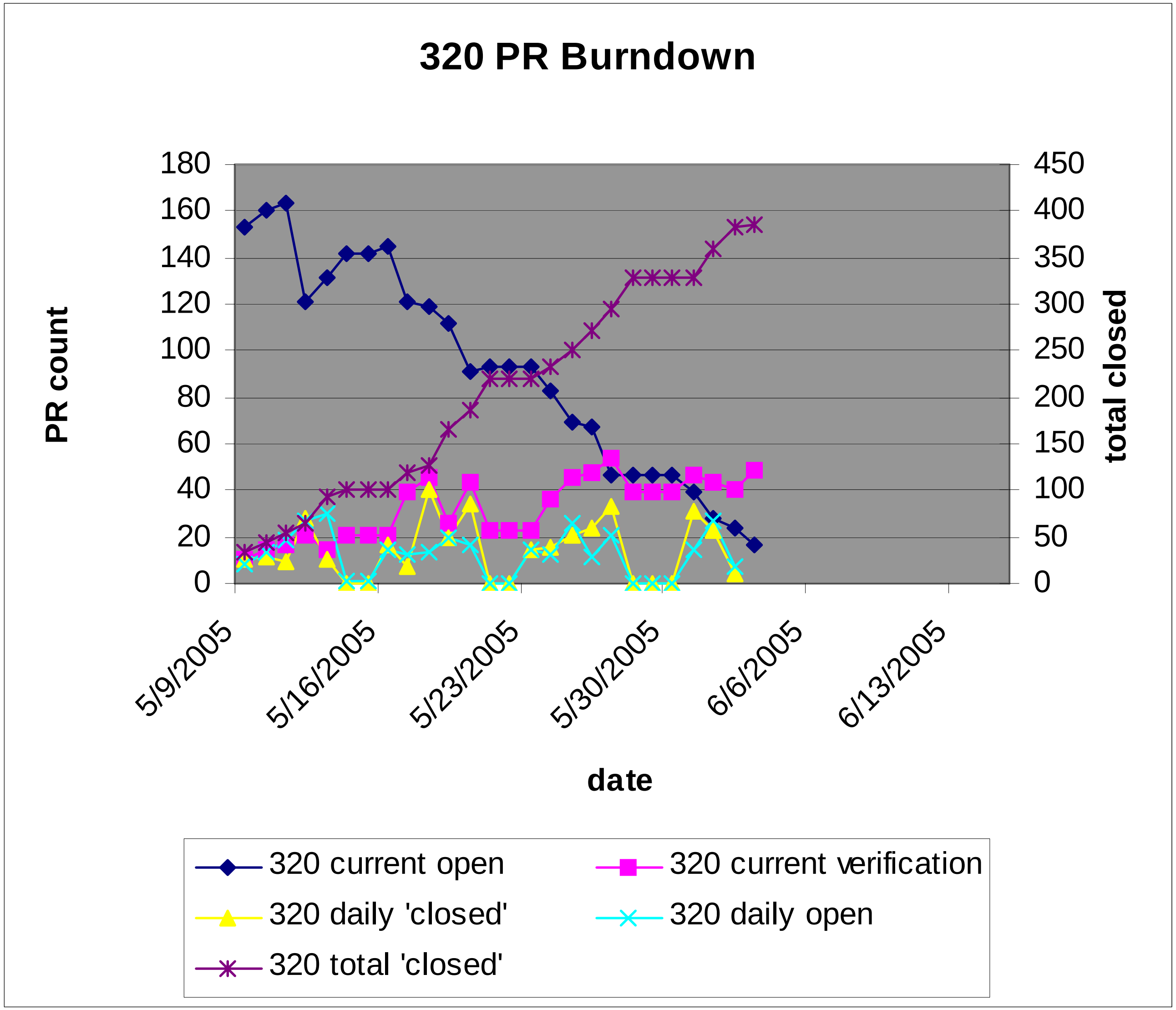
Red - weekly

Blue - monthly

Green - quarterly

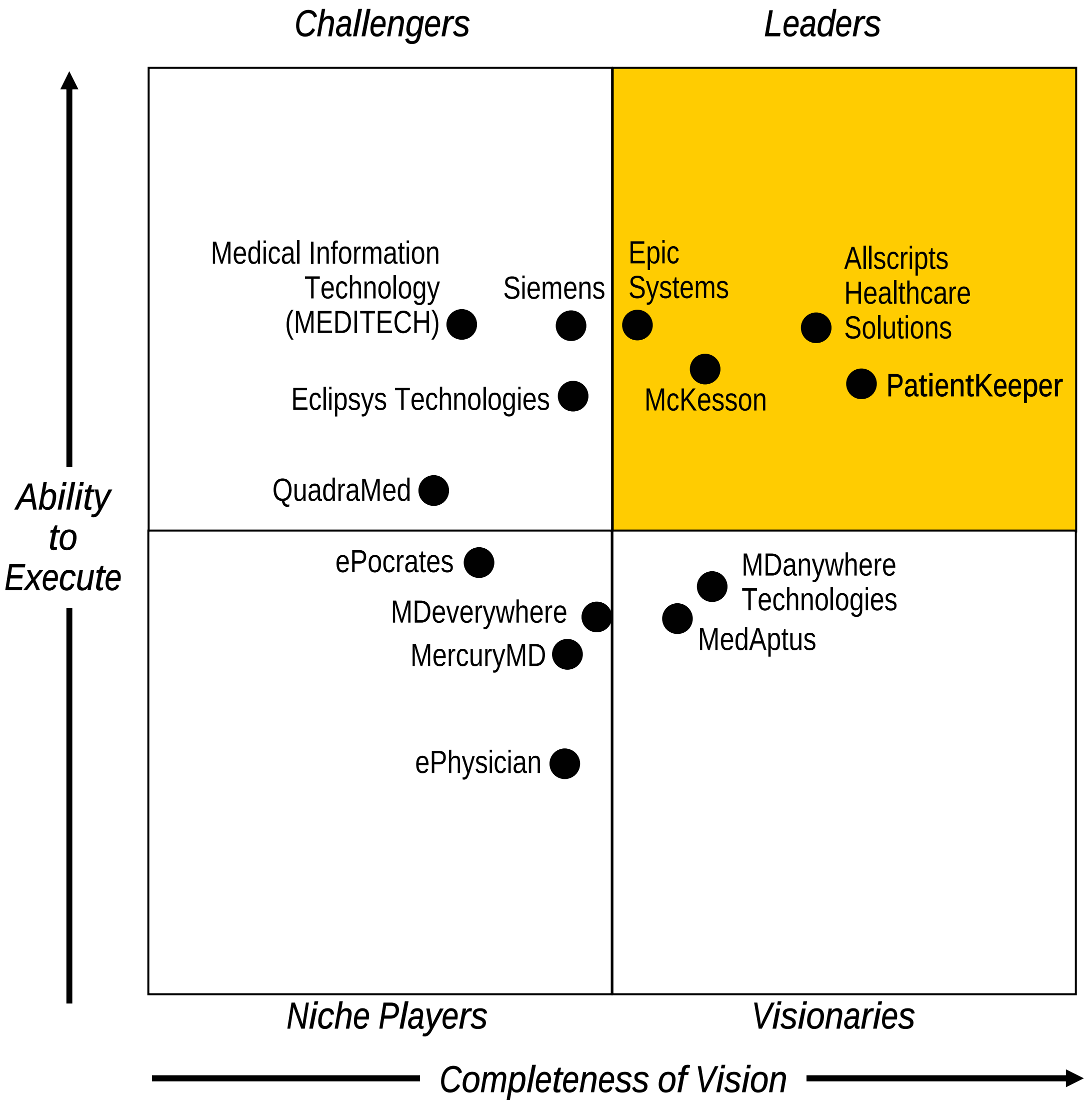
PatientKeeper delivered 45 production releases of quality code to large healthcare systems in 2004.

Project Reporting



© Jeff Sutherland 1993-2007

Practice: Agility
Using Scrum Type C to Capture Industry Leadership



Gartner Magic Quadrant

"PatientKeeper is one of the best-funded and strongest vendors in the mobile/wireless healthcare market. It is one of the few to market itself as providing a mobile computing infrastructure and development environment for which it, and other vendors, system integrators and users, can develop their own mobile applications. It supports both the Palm and Pocket PC platforms."

— *Ken Kleinberg, Gartner Research*

MOORE
© 2002 Universal Press Syndicate
4-12
www.ucomics.com
SCRUM!!
GRRRR!
GRRRRR!
GRRRR!!

Questions?

Bibliography

- Cohn, M. (2004). User Stories Applied : For Agile Software Development, Addison-Wesley.
- Cohn, M. (2005). Agile Estimation and Planning, Addison-Wesley.
- Poppendieck, M. and T. Poppendieck (2006). Lean Software Development: An Implementation Guide, Addison-Wesley.
- Kniberg, Henrik. Scrum and XP from the Trenches: How We Do Scrum. Version 2.1, Crisp, 5 Apr 2007.
- Sutherland, J. (2005). Future of Scrum: Parallel Pipelining of Sprints in Complex Projects. AGILE 2005 Conference, Denver, CO, IEEE.
- Sutherland, J., C. Jacobson, et al. (2007). Scrum and CMMI Level 5: A Magic Potion for Code Warriors! Agile 2007, Washington, D.C., IEEE.
- Sutherland, J. and K. Schwaber (2007). The Scrum Papers: Nuts, Bolts, and Origins of an Agile Method. Boston, Scrum, Inc.
- Takeuchi, H. and I. Nonaka (1986). "The New New Product Development Game." Harvard Business Review(January-February).
- Takeuchi, H. and I. Nonaka (2004). Hitotsubashi on Knowledge Management. Singapore, John Wiley & Sons (Asia).