

# End-to-End SDLC with Agile Methodologies

Baiju Joseph, Senior QE Manager ,Yahoo! &  
Srinivasan Desikan, Master System Technologist , HP

17<sup>th</sup> Dec 2009 - CSI, Chennai

0

## Agenda

- Introduction to Agile Methodologies
- Extreme Programming (XP) basics
- Scrum basics
- Scrum roles and terminologies
- Scrum Ceremonies and artifacts
- Test driven development, Automation
- Implementing Agile with Quality Center

1

# End-to-End SDLC with Agile Methodologies

Introduction to Agile Methodologies

2

## Monumental Methodologies

© Original Artist  
Reproduction rights obtainable from  
[www.CartoonStock.com](http://www.CartoonStock.com)



"Somehow, it's not the same as when we were here on our honeymoon."

3

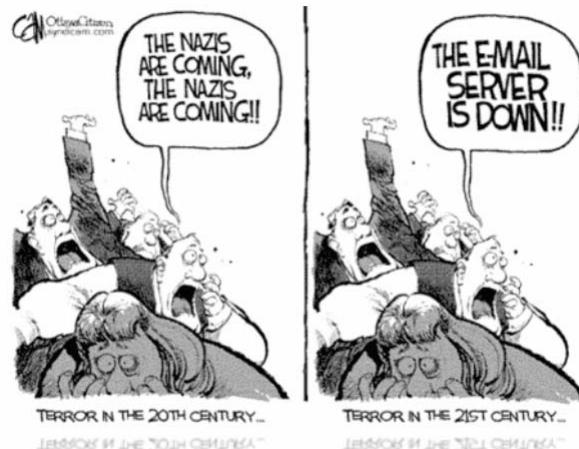
## Monumental Methodologies

# Different reality

Different business needs and dynamics

4

## Monumental Methodologies



5

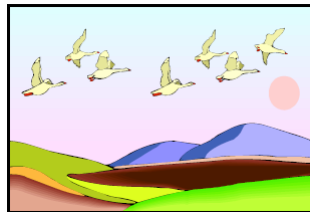
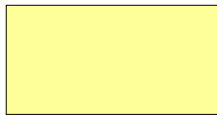
## Agile Methodologies

# Continuous Improvement

There is nothing new here

6

## From Nothing, to Heavy, to Light



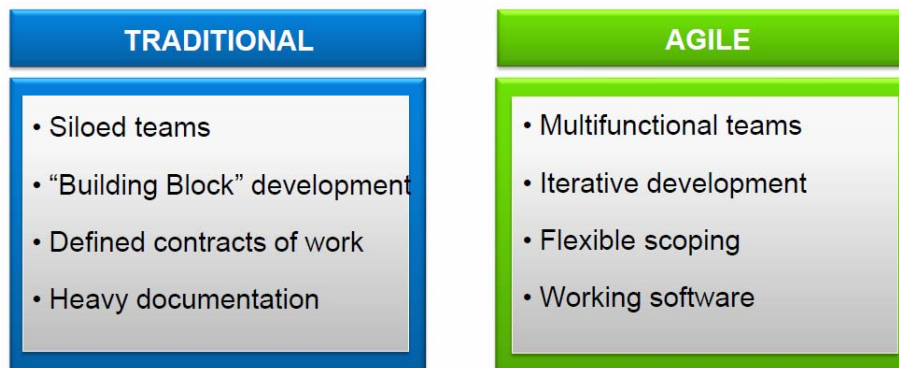
7

## Agile Methodologies

- Reaction to the bureaucracy of the engineering methodologies.
- Compromise between no process and too much process
- *Adaptive* rather than predictive
- *People-oriented* rather than process-oriented.

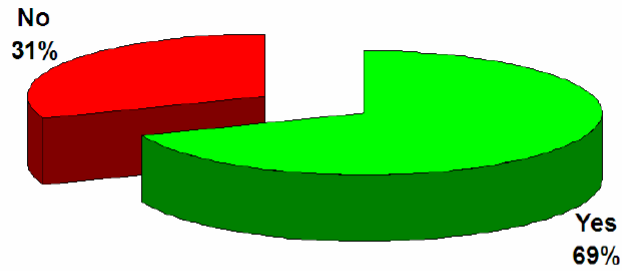
8

## Agile Methodologies



9

## Agile Methodologies



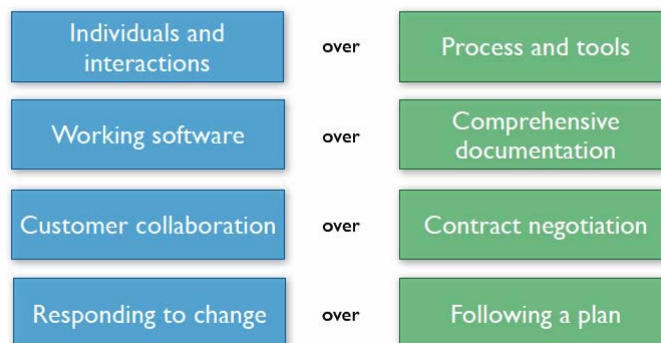
781 respondents:  
52% were developers, 22% were in management  
40% had 10-20 years IT experience, 33% had 20+ years  
33% worked in orgs of 1000+ people  
85% worked in commercial firms

Source: [www.ambysoft.com/surveys/](http://www.ambysoft.com/surveys/)

10

## Agile Methodologies

### Agile Manifesto



Source: [www.agilemanifesto.org](http://www.agilemanifesto.org)

11

## Agile Methodologies

### Methodologies

Methodologies share common principles,  
but differ in practices

- eXtreme Programming (XP)
- Scrum
- Evolutionary Project Management (Evo)
- Unified Process (UP)
- Crystal
- Lean Development (LD)
- Adaptive Software Development (ASD)
- Dynamic System Development Method (DSDM)
- Feature Driven Development (FDD)

12

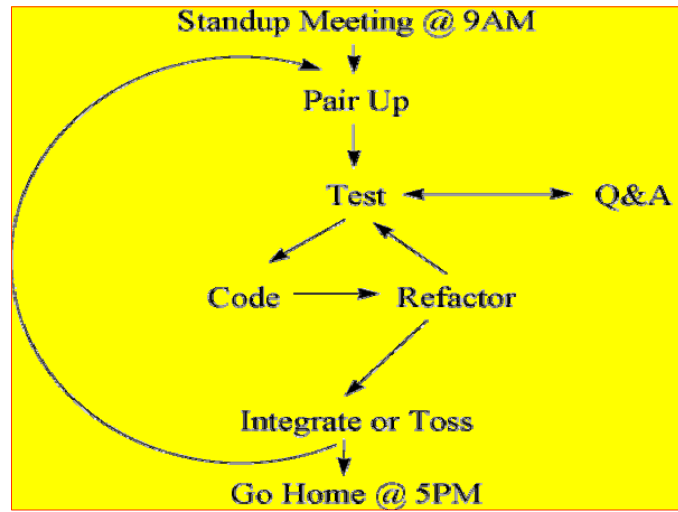
## Extreme Programming (XP) basics

- Review code all the time...
- Everybody test all the time...cross boundaries
- Integrate and test several times a day...
- Short iterations...customer involvement...
- Continues design refactoring...
- Everybody define and refine architecture



13

## XP Day



14

## End-to-End SDLC with Agile Methodologies

Scrum Basics

15

## Scrum...



- A *scrum* is an important way to restart play during a rugby match. To form a scrum, the forwards of each team pack together in a tight formation. The *scrum half* of the team in possession then rolls the ball between the two front rows of forwards, and each team tries to hook the ball backward with the feet. When the ball is released by the successful team's last forward, the scrum half takes the ball and runs, passes, or kicks, as play continues.
- Check out: <http://www.youtube.com/watch?v=gjzhGqVvoo4&NR=1> for a short video of a scrum.

16

## Scrum

- Scrum excels on urgent projects that are critical to the organization. Scrum excels when requirements are unknown or changing. Scrum helps by helping teams excel.

17

## Scrum origins

- Jeff Sutherland
  - Initial Scrums at Easel Corp in 1993
  - Initial definitions of Scrum at OOPSLA 96 with Ken Schwaber
- Ken Schwaber
  - *Agile Software Development with Scrum* book in 2001 along with Mike Beedle
  - *Agile Project Management with Scrum* book in 2004

18

## Scrum has been used in...

- Independent Software Vendors (ISVs)
- Fortune 100 companies
- Small startups
- Internal development
- Contract development

19

## Scrum has been used by...

- ▶ Microsoft
- ▶ Yahoo!
- ▶ Google
- ▶ Electronic Arts
- ▶ Lockheed Martin
- ▶ Philips
- ▶ Siemens
- ▶ Nokia
- ▶ BBC
- ▶ Nielsen Media
- ▶ SalesForce.com
- ▶ Oracle
- ▶ Time Warner
- ▶ Turner Broadcasting
- ▶ Toyota
- ▶ BMC Software
- ▶ CapitalOne
- ▶ hp

20

## Scrum has been used for...

- FDA-approved, life-critical software for x-rays and MRIs
- Enterprise workflow systems
- Financial payment applications
- Biotech
- Call center systems
- Tunable laser subsystems for fiber optic networks
- Application development environments
- 24x7 with 99.99999% uptime requirements
- Multi-terabyte database applications
- Media-neutral magazine products

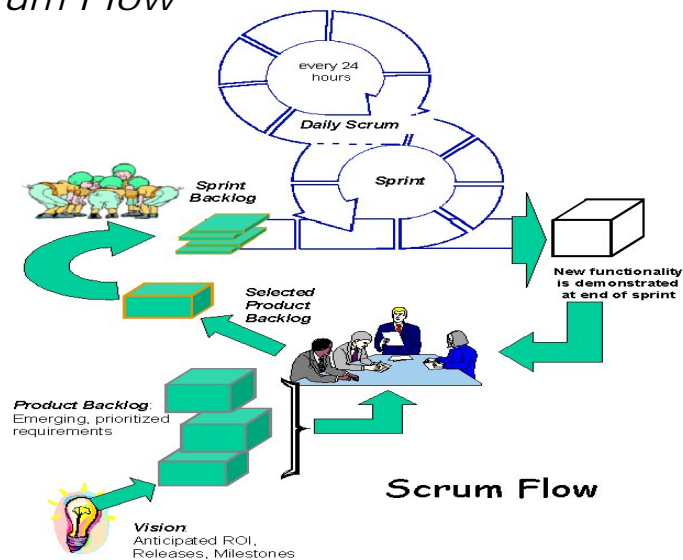
21

## Scrum Characteristics

- One of the agile methodologies
- Self-organizing teams
- Product progresses in a series of month-long “sprints”
- Requirements are captured as items in a list of “product backlog”
- No specific engineering practices prescribed
- Wraps existing engineering practices, including Extreme Programming
- Delivers business functionality in 30 days

22

## Scrum Flow



23

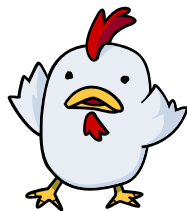
**Scrum-It's About Common Sense**

## Scrum roles and terminologies

24

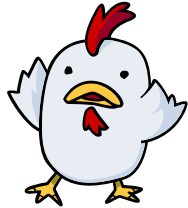
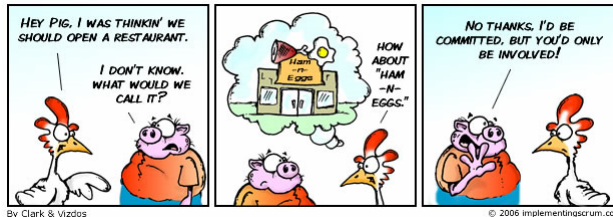
### Scrum's Roles

- The Product Owner
- The Scrum Master
- The Team
- Everyone else is not part of Scrum



25

## Scrum's Roles



26

## The Product Owner



- Creates initial product vision - overall requirements
- Manage and prioritize the Product Backlog
- Decide on release dates and its contents
- Responsible for the ROI, audience, etc.
- Interact with stakeholders and customers to define the Product backlog
- Adjust features and prioritize every sprint, as needed!
- Accept or reject work results during the Sprint Review

27

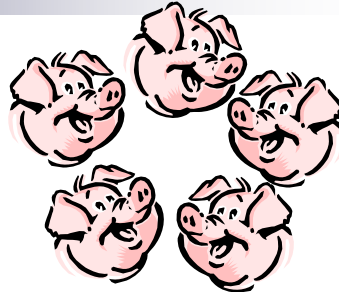
## The Scrum Master



- Protect values and principles
- Removes impediments
- Keep the team fully functional and productive
- Enable cooperation
- Shield the team from “dark forces”
- Facilitates integration
- Does **NOT** allocate tasks

28

## The Team

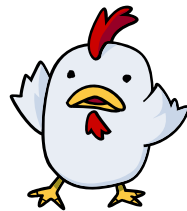
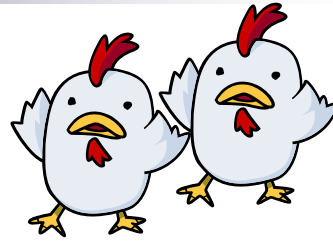


- Typically 7 plus or minus 2
- Self organizing
- Self managing
- Full time allocated
- Cross functional (Developers , QA ,Doc ,UED, PM etc...)
- Figure out how to turn Scrum backlog into an increment of functionality.
- Responsible for the quality
- Estimates the complexity
- Committed to developing functionality

29

## Chickens

- No formal responsibilities
- No accountabilities
- Not a team member
- Someone who is interested in the project



30

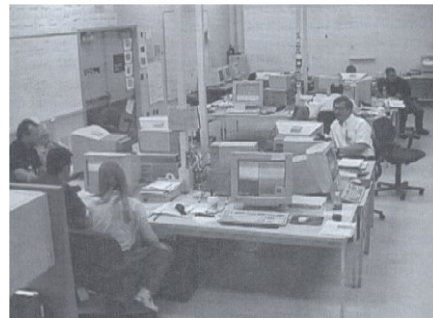
## Facilities Strategy

- Open space.
- Tables in the middle of the space.
- Cubbies (Private) around the outside of the space.

The DaimlerChrysler  
C3 work area



**From: eXtreme Programming  
Explained 2000**



31

## Scrum Practices

- The Sprint Planning Meeting
- The Sprint
- The Daily Scrum
- The Sprint Review Meeting
- The Sprint Retrospective
- All other practices are not part of Scrum



32

## Scrum's Artifacts

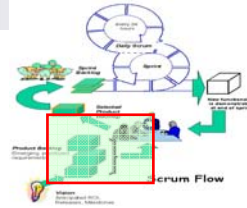
- The Product Backlog
- The Product Burndown Chart
- The Sprint Goal
- The Sprint Backlog
- The Sprint Burndown Chart
- The Potentially Shippable Product Increment



33

## Product Backlog

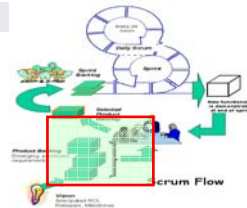
- The product requirements
- Everything (ideas, features, epics)
- Expressed in User Stories
- PO keeps it organized
- Always prioritized and estimated
- Maintained and visible
- Evolves as business need, technology changes
- Reviewed in every Sprint



34

## Product Backlog

- One list for multiple teams
- Product Owner responsible for priority – agile business project manager
- Anyone can contribute
- Evolves as business need, technology changes
- Reviewed in every Sprint

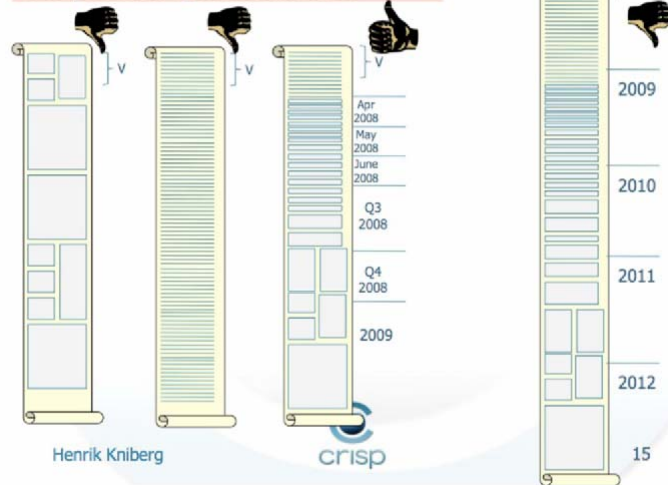


35



## The Product Backlog

10 ways to screw up with Scrum and XP



38

## The User Story

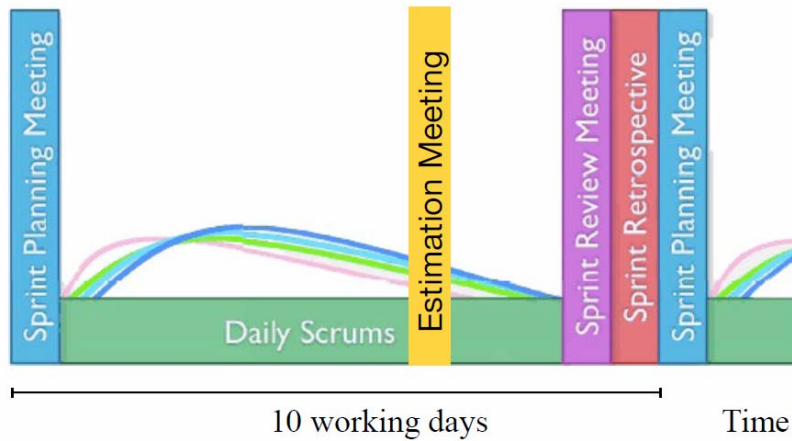
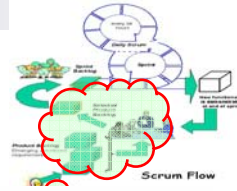
I as a <client>  
want <feature>  
because <ROI>

Acceptance Criteria

It does not say how to do it

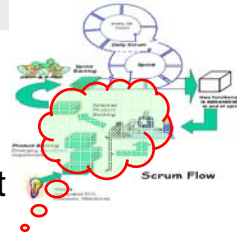
39

## Meetings



40

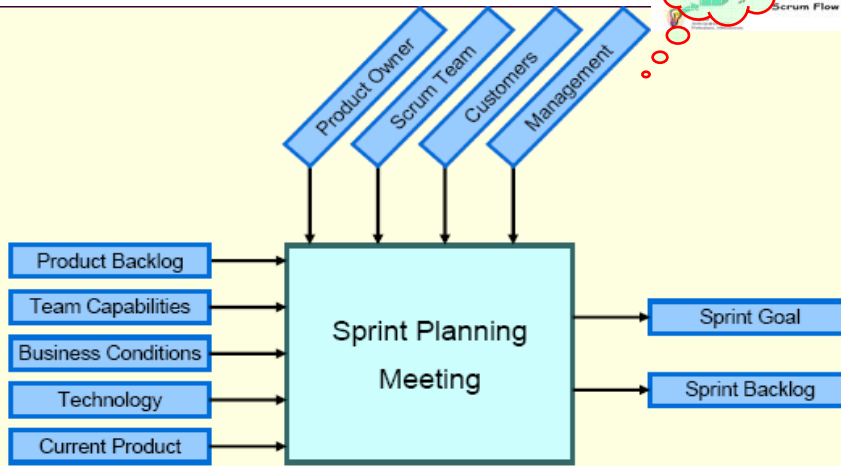
## Sprint Planning



- What will be done for the next Sprint
- Sprint Planning Phase 1 – 4 hrs
  - Product owner present highest priority product backlog
  - Review, consider, organize
  - Team select as much product backlog it believe it can develop in the next sprint
  - Team Commitment
  - Product Owner, Team, customers and management

41

## Sprint Planning



42

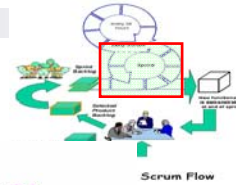
## Sprint Planning...

- Sprint Planning Phase 2 – 4 hrs
  - Team responsible for managing its own work
  - Team plans out Sprint – tentative plan
  - Tasks entered in Sprint backlog based on the plan
  - Tasks in sprint backlog emerges as Sprint progress
  - Chickens not allowed

43



## Sprint



# No changes during the Sprint



46

## Sprint



- Time box of 30 sequential calendar days
- Team works to turn the sprint backlog into an “increment of potentially shippable functionality”
- “increment of potentially shippable functionality” that meets the sprint goal.
- Self organization of the team
- Activities are visible through the Sprint Backlog and Sprint Burndown Charts
- Tasks not completed in current Sprint → moved to the next Sprint backlog

47

## Sprint – Weekly cadence

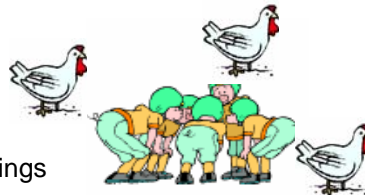


- Build
- Develop
- Weekly Demo
- Group Test (Friday Group Test)
- Checkin

48

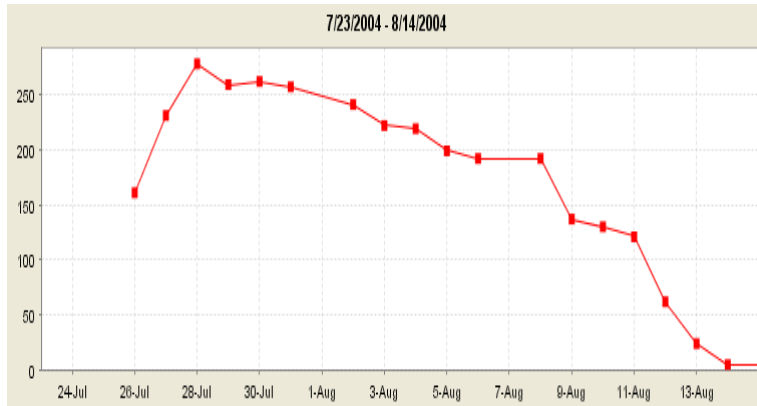
## Daily Scrum

- Socialization and Synchronization
- Time boxed to 15 minutes
- Daily Stand up meeting
- Not for problem solving. Resolutions are offline.
- Three questions:
  - 1. What did you do yesterday
  - 2. What will you do today?
  - 3. What obstacles are in your way?
- Chickens and pigs are invited
  - Help avoid other unnecessary meetings
- Only pigs can talk



49

## Sprint Burndown Chart

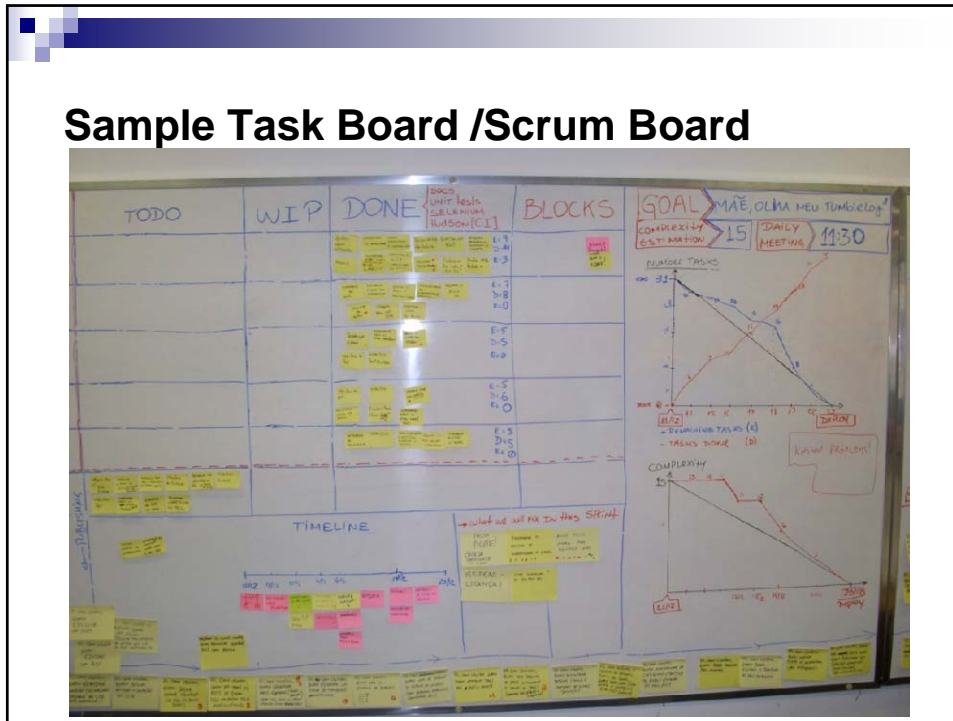


50

## Sample Task Board /Scrum Board



## Sample Task Board /Scrum Board



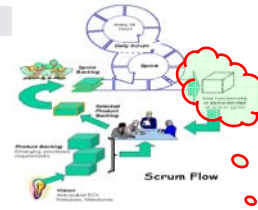
## Sprint Review Meeting

- End of Sprint
- Time boxed to 4 hrs
- Informal
- Team presents (Demo)
  - What was accomplished during the current sprint
  - To product owner, the team and any chickens who are interested in the sprint outcome
  - Input to the next sprint planning



## Sprint Retrospective Meeting

- End of Sprint...
- After sprint review and before next sprint planning
- Time boxed to 3 hrs
- Facilitated by Scrum master
- Team reviews the just concluded sprint and determine
  - What went well and What are the improvement areas
  - What need to be changed
  - So that next sprint more enjoyable and productive
- Team devises solution to most vexing problems



54

## Meeting suggestion

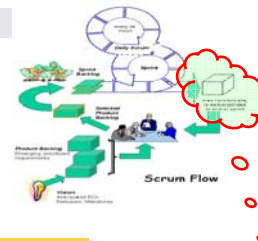
Sprint Review (2h max)

Sprint Retrospective (2h max)

Lunch

Sprint Planning 1 (2h max)

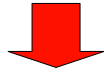
Sprint Planning 2 (2h max)



55

## Defining “Done”

- Potentially shippable product increment
  - At the end of every Sprint
  - Customer should be able to use the increment

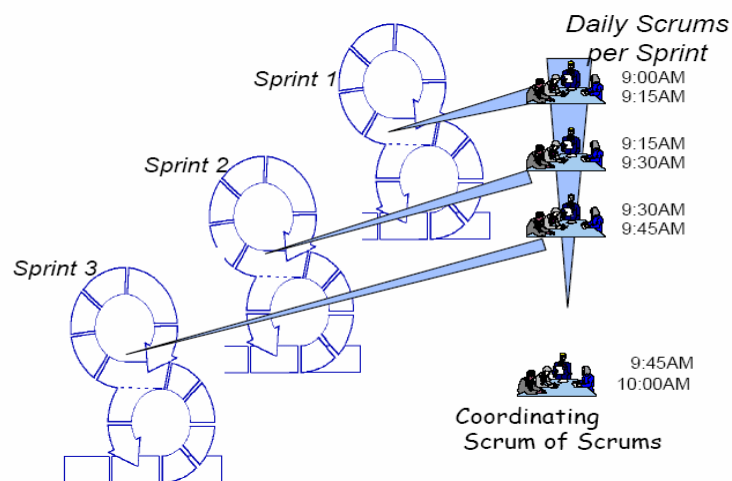


- **Thoroughly tested**
- Checked in
- Built into an executable → implementable
- Fully documented
- Sashimi- thin slice of a product which contains all aspects of the final product




56



## Scrum of Scrums

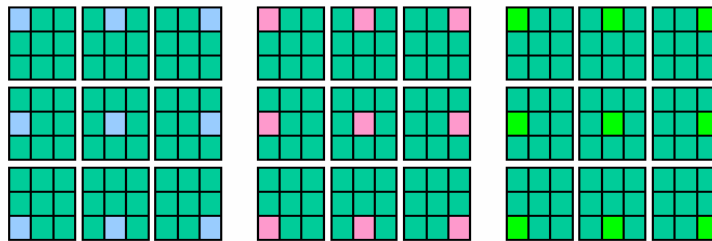


57

## Scrum of Scrums

Coordinating  
Scrum 

 Scrum of  
Scrums 



Daily Scrums

58

## Agile Testing

The main reason that testing at the end of a development cycle finds problems is not that problems were put in near the end, it is that testing was put off until then.  
- Ron Jeffries

59

## QE in a Scrum Team

- Testing in Scrum



60

## QE and Sprint

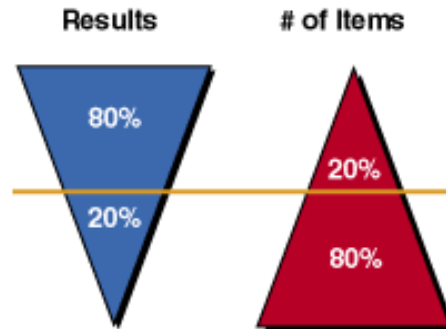
- Test Planning & estimates - QE
- Don't try to do too much documentation
- Test Plan reviewed by the Scrum Team
- Test case design – QE
- Test cases reviewed by Scrum Team
- Try to complete this in the initial 2 weeks
- Adhoc testing on the developer build
- More focused testing during the final 2 weeks



61

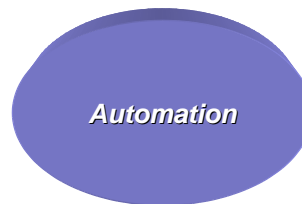
## 80/20 Rule

Test top priority test cases first....



62

## Automation



- ▶ Automation of acceptance tests of current Sprint
- ▶ Prioritize
- ▶ Early automation of new acceptance tests
  - ▶ allow defects to be uncovered and corrected on an on-going basis.
- ▶ Automate any test when it makes sense, where it is possible

63

## Automation

- Identify the test cases to be automated
- Factors to be Considered
  - ROI
  - Time to automate vs Time to Test(manual)
  - No of times the test will be executed
  - Importance of the Test case (Highly used scenario??)

64

## Test Reporting

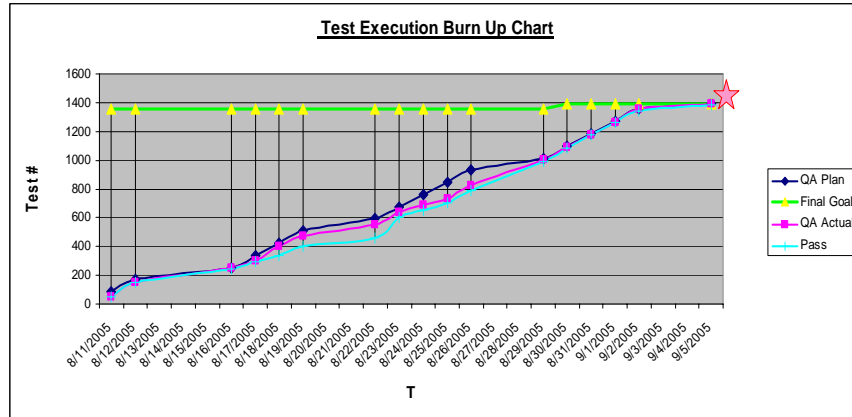
- Weekly Test Reports
- Sprint Test Reports
- Completion and Pass %
- Defect Data
- Risks ,Top Issues & Dependencies



*Test Reports*

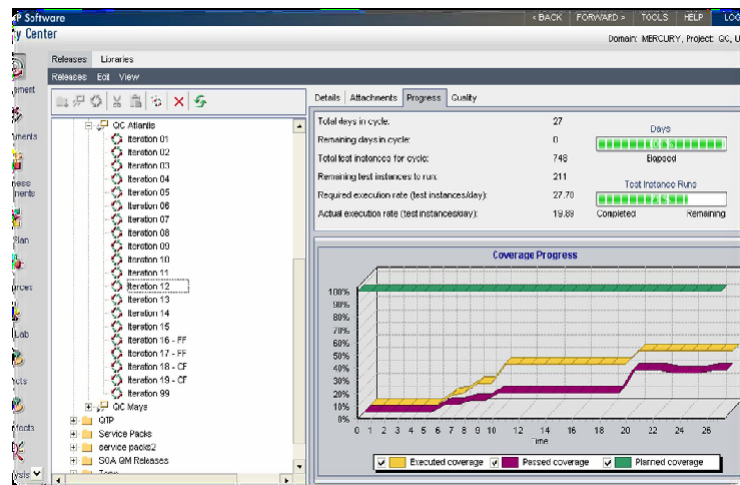
65

# Test Execution Burn up Chart



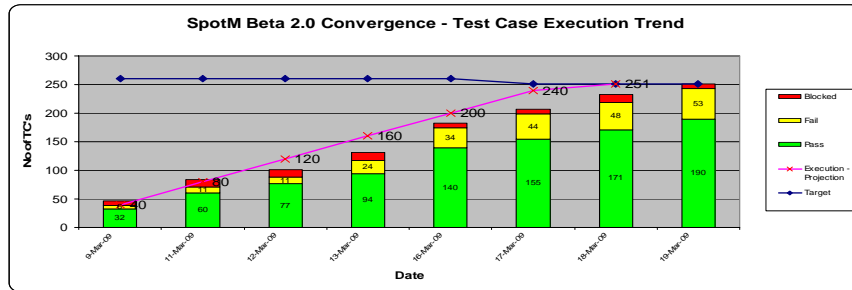
66

# Test Execution Burn up Chart



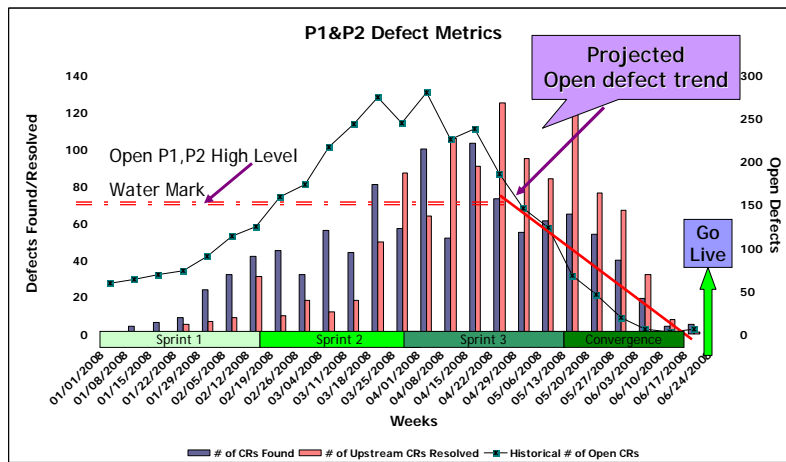
67

# Test Execution Burn up chart



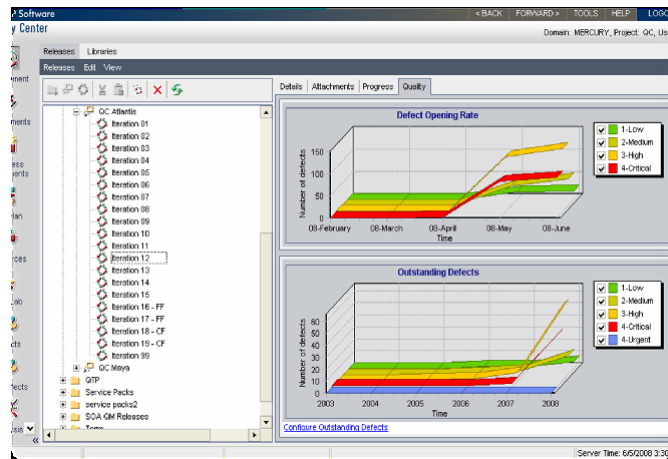
68

# Defect Metrics



69

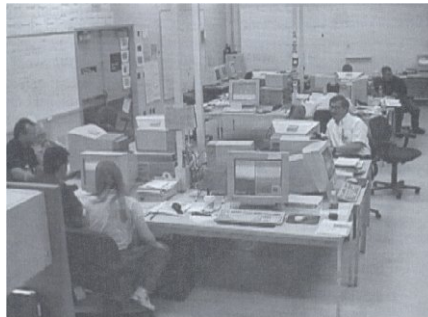
## Defect Metrics



70

## Agile Testing – Best Practices

- Pair testing
- Refactoring
- Onsite customer



71

## **Scrum-It's About Common Sense**

### Building an effective Agile Team

72

## **Phased Approach**

- Educate
  - External Consultant – Certified Scrum Master
- Evangelize
  - Simplicity
  - Benefits
- Finding a “Good” Scrum Master who can drive Scrum in your team
  - Leader and Facilitator
- Fine tuning processes and practices of the team

73

## Let's Scrum

- Set up the Scrum Team
- Scrum master as a mentor and Change agent
- Support from Top Management
- Build an open culture
- Cross functional teams
- Self Managed and self organized...
- Sub-teams (including Testing) to work tightly as a single team

74

## Let's Scrum

- Environment
  - Preferably every one in same location
  - Maximize Communications
  - White Board / Scrum Board
  - Big Visible charts
  - Light Weight tools

75

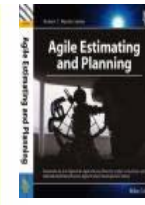
## References

### ■ URLs...

- [www.controlchaos.com](http://www.controlchaos.com)
- <http://jeffsutherland.com/scrum/>
- [www.mountangoatsoftware.com/scrum](http://www.mountangoatsoftware.com/scrum)
- [scrumdevelopment@yahoo.com](mailto:scrumdevelopment@yahoo.com)

### ■ Books

- *Agile Software Development with Scrum*
  - Ken Schwaber and Mike Beedle
- *Agile Project Management with Scrum*
  - Ken Schwaber and Mike Beedle
- *Agile Estimating and Planning*
  - Mike Cohn
- *Agile Testing: A Practical Guide for Testers and Agile Teams*
  - Lisa Crispin



76

## References- More books

- *Agile Estimating and Planning* by Mike Cohn
- *Lean software development* by Mary Poppendieck
- *Agile and Iterative Development: A Manager's Guide* by Craig Larman
- *Agile Retrospectives* by Esther Derby and Diana Larsen
- *Scrum and The Enterprise* by Ken Schwaber
- *Practices of an Agile Developer* by Venkat Subramaniam and Andy Hunt
- *Test Driven Development: By Example* by Kent Back Getting Real by 37Signals
- *Agile Testing: A Practical Guide for Testers and Agile Teams* by Lisa Crispin
- *Software Testing- Principles and Practice* by Srinivasan Desikan and Gopaldaswamy Ramesh

77

