

Agile and the Seven Deadly Sins of Project Management

Mike Cohn

June 11, 2008

1

A cornucopia of agile processes

Agile Processes

- Extreme Programming (XP)
- Scrum
- Crystal
- DSDM
- Lean software development
- Unbranded “agile”

Semi-Agile Processes

- Feature-Driven Development (FDD)
- Unified Process
- OpenUP



2

A closer look at Scrum

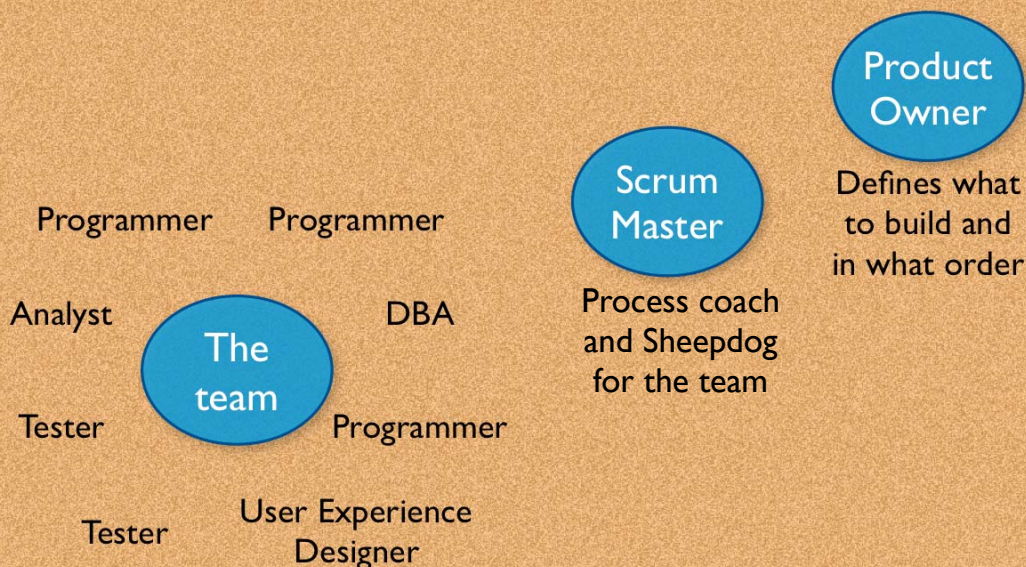
- No specific engineering practices prescribed
 - But many Scrum teams are adopting much of XP
- 2- to 4-week iterations called “sprints”
- Self-organizing, cross-functional teams
- Uses generative rules to create an agile environment



© 2003–2008 Mountain Goat Software®

3

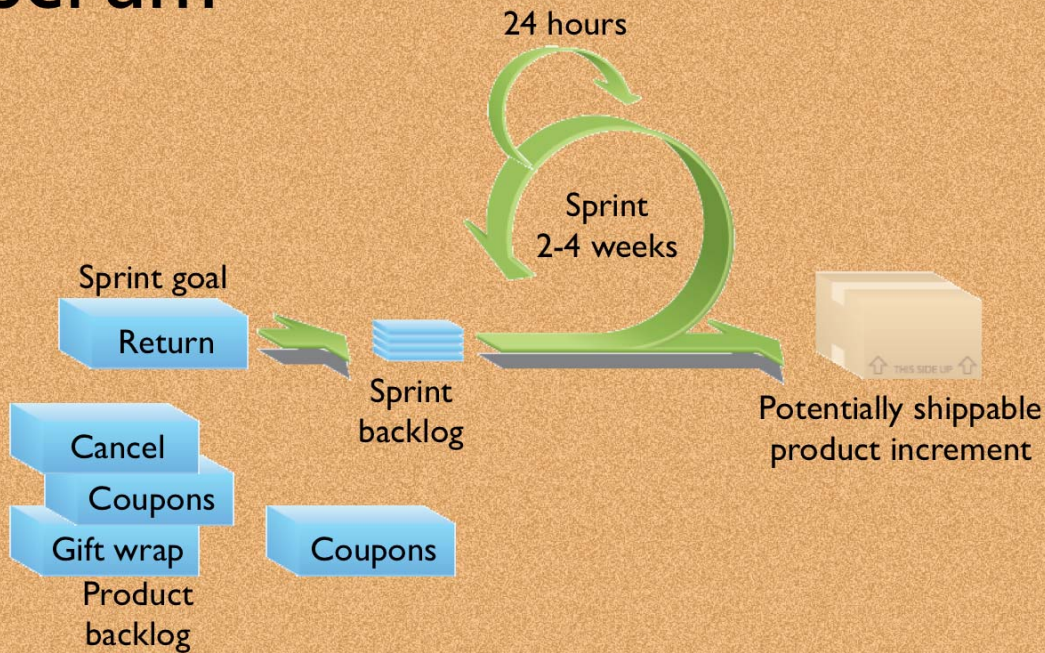
The Scrum project community



© 2003–2008 Mountain Goat Software®

4

Scrum



Seven Sins of Project Management

- 1 Gluttony
- 2 Lust
- 3 Sloth
- 4 Opaqueness
- 5 Pride
- 6 Wastefulness
- 7 Myopia



Sin #1: Gluttony

- Definition
 - Fixing all dimensions (scope, schedule, resources, and quality) of a project
 - A project management sin of excess
- Experienced as
 - Impossible schedules
 - Death marches
- Leads to
 - Trying to do too much for the resources (time, people) available
 - Cutting quality to meet other goals



© 2003–2008 Mountain Goat Software®

7

The iron triangle



Which dimension is best to adjust to achieve project success?

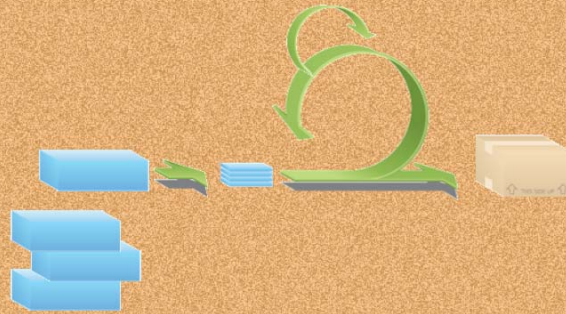


© 2003–2008 Mountain Goat Software®

8

Timeboxes help avoid gluttony

- Agile iterations are timeboxed
 - So the schedule of an iteration is fixed
 - But the number of iterations is variable
 - Focus is always on “what can we accomplish next?”

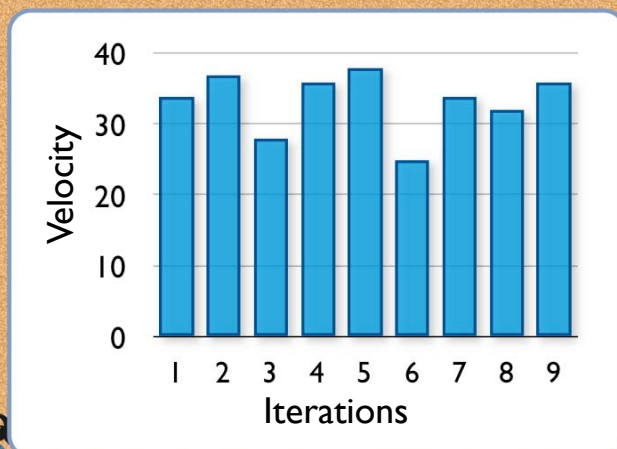


© 2003–2008 Mountain Goat Software®

9

Timeboxed iterations increase predictability

- Over time a team learns how much it can complete per iteration (its “velocity”)



- Knowing this prevents the temptation to overcommit



© 2003–2008 Mountain Goat Software®

10

What expectation of future velocity should this team set?

Feature A	5
Feature B	3
Feature C	5
Feature F	3
Feature D	5
Feature E	5
Feature G	3
Feature I	3
Feature H	5
Feature J	2
Feature K	5
Feature L	3

✓	Feature A	5
✓	Feature B	3
✓	Feature C	5
	Feature F	3
	Feature D	5
	Feature E	5
	Feature G	3
	Feature I	3
	Feature H	5
	Feature J	2
	Feature K	5
	Feature L	3



© 2003–2008 Mountain Goat Software®

11

Sin #2: Lust

- Definition
 - Intense or unrestrained craving for features
- Experienced as
 - Trying to put too many features into a product during the time allowed
 - Treating all features as “critical”
- Leads to
 - Overtime, reduced quality, surprises



© 2003–2008 Mountain Goat Software®

12

Three ways agile deals with lust

1

Developing features in priority order

2

Incremental gratification



© 2003–2008 Mountain Goat Software®

13

3

Working at a sustainable pace

“Overtime is a symptom of a serious problem on the project. The XP rule is simple—you can’t work a second week of overtime. For one week, fine, crank and put in some extra hours. If you come in on Monday and say ‘To meet our goals, we’ll have to work late again,’ then you already have a problem that can’t be solved by working more hours.”

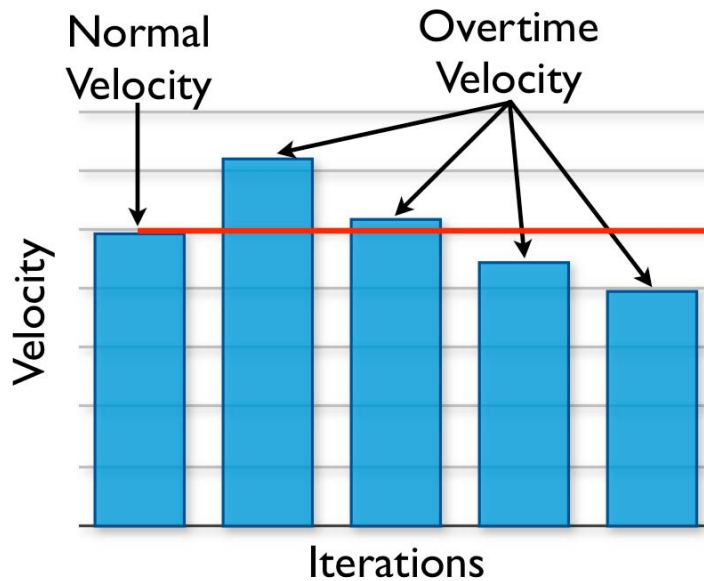
~Kent Beck



© 2003–2008 Mountain Goat Software®

14

Old habits die hard



© 2003–2008 Mountain Goat Software®

15

Sin #3: Sloth

- Definition
 - Failing to do high quality work at all times
- Experienced as
 - Testing quality in at the end
 - Instability during development
- Leads to
 - Big delays
 - Unpredictable schedules



© 2003–2008 Mountain Goat Software®

16

Agile practices related to quality

Simple design

Automated testing

Test-driven development

Continuous integration

Pair programming

Refactoring



© 2003–2008 Mountain Goat Software®

17

Case Study I

Cosmodemonic Biotech

	Waterfall	Scrum
Use Case pages	3,000	
User Stories		1,400
Calendar months	9	12
Person months	540	54
Lines of Java code	58,000	51,000
Lines of Java code per person-month	120	840



© 2003–2008 Mountain Goat Software®

18

ePS

Productivity (NCSS / month)

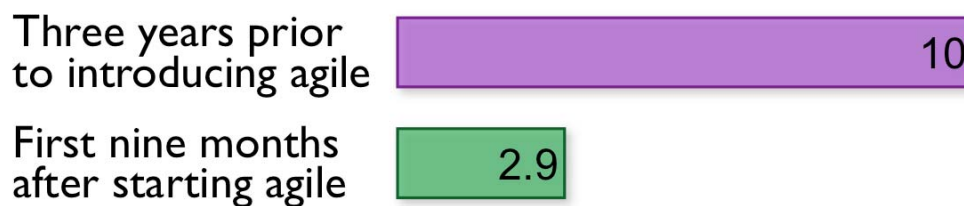


NCSS=Non-Comment Source Statment (Java)



ePS

Defects per KNCSS



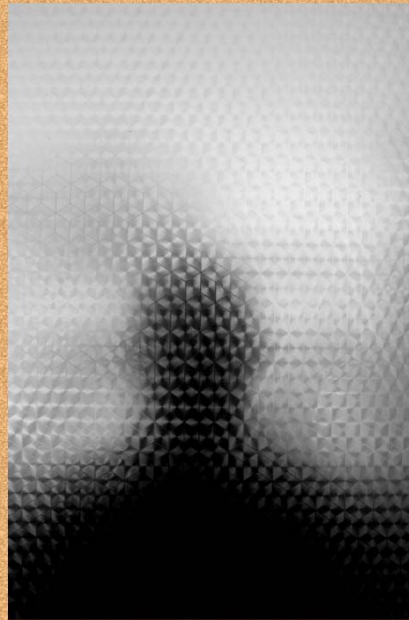
But wait, there's more...

- Results achieved without any targeted rewrite of existing (buggy) code
- Many of the post-agile defects continued to be in the old code
- True results would be even better (if we had measured them)



Sin #4: Opaqueness

- Definition
 - Obscuring the progress, quality or other attribute of a project
- Experienced as
 - Not knowing the true state of the project
- Leads to
 - Surprises
 - Poor decisions



© 2003–2008 Mountain Goat Software®

21

Three types of opaqueness

1

How agile addresses Quality opaqueness

- Don't let bugs accumulate
- Continuous integration
- Features are either “Done” or “Not Done”
 - Avoids the 90% Syndrome



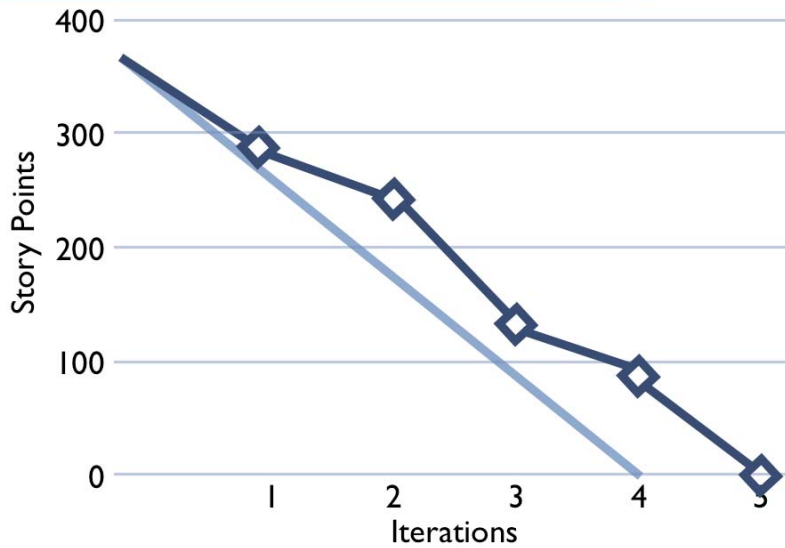
© 2003–2008 Mountain Goat Software®

22

2

How agile addresses Schedule opaqueness

A release burndown chart

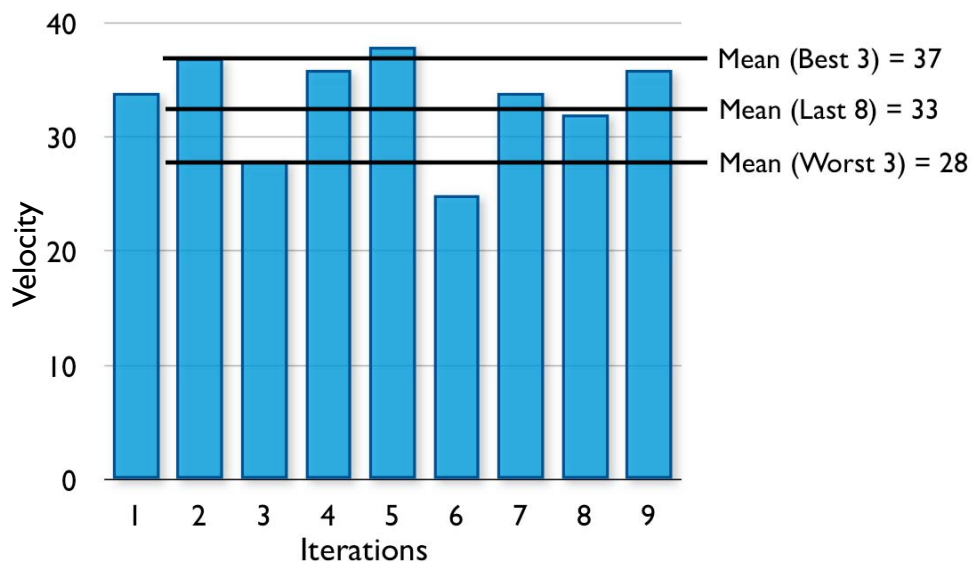


© 2003–2008 Mountain Goat Software®

23

3

How agile addresses Scope opaqueness

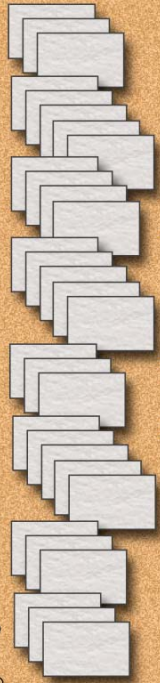


© 2003–2008 Mountain Goat Software®

24

Predicting what will be in

Assume:
There are five sprints left.



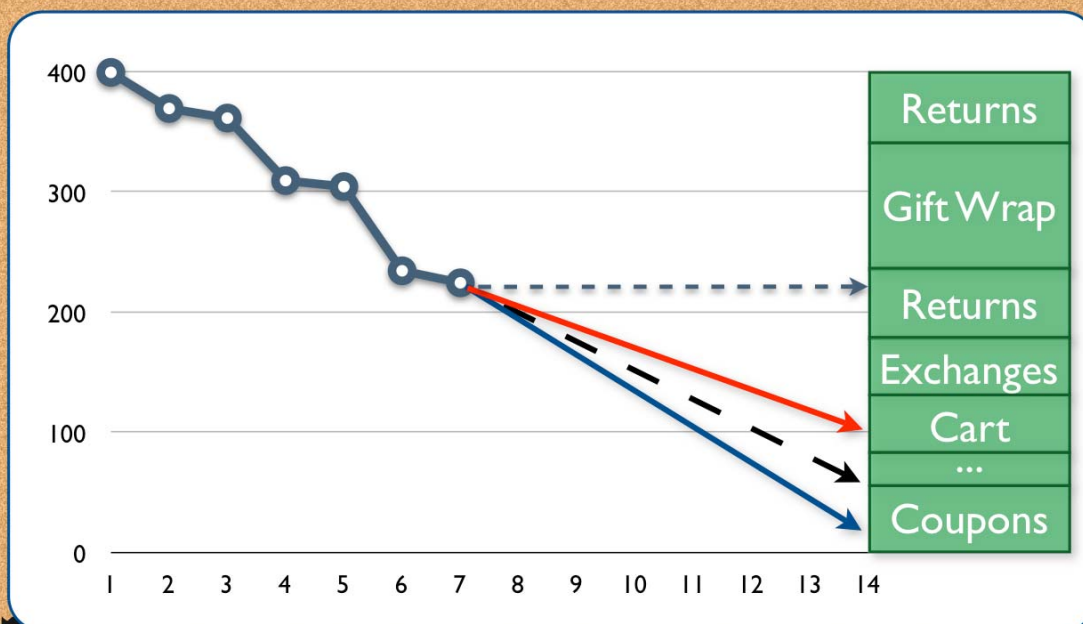
← At our slowest velocity we'll finish here (5×28)

← At our long-term average we'll finish here (5×33)

← At our best velocity we'll finish here (5×37)

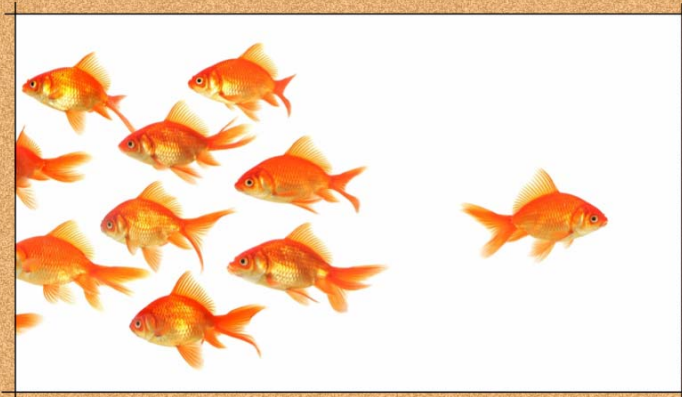


A Backlog Burndown Chart



Sin #5: Pride

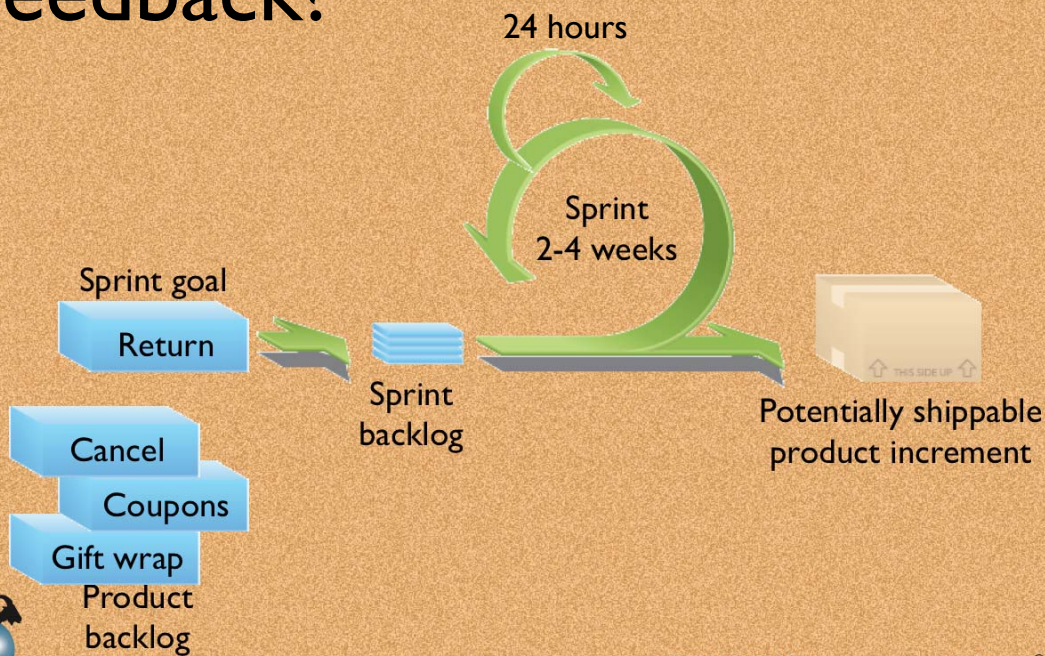
- Definition
 - Believing that we know everything to build the product
- Experienced as
 - A lack of stakeholder and user involvement
- Leads to
 - Failure to solicit feedback
 - Failure to learn



© 2003–2008 Mountain Goat Software®

27

Where are opportunities for feedback?



© 2003–2008 Mountain Goat Software®

28

Agile requirements

“User stories” facilitate working closely with users & customers

As a user, I want to reserve a hotel room.

As a vacation traveler, I want to see photos of the hotels so that I can choose the best one for me.

As a user, I want the site to be available 99.999% of the time I try to access it.



29

Progressive refinement

As a VP Marketing, I want to review the performance of historical promotional campaigns so that I can identify and repeat profitable ones.

As a VP Marketing, I want to select the timeframe to use when reviewing the performance of past promotional campaigns, so that I can identify and repeat profitable ones.

As a VP Marketing, I can select which type of campaigns (direct mail, TV, email, radio, etc.) to include when reviewing the performance of historical promotional campaigns.



30

An example

As a VP Marketing, I want to see information on **direct mailings** when reviewing historical campaigns.

As a VP Marketing, I want to see information on **television** advertising when reviewing historical campaigns.

As a VP Marketing, I want to see information on **email** advertising when reviewing historical campaigns.



© 2003–2008 Mountain Goat Software®

31

Sin #6: Wastefulness

- Definition
 - Misuse of critical resources
- Experienced as
 - Losses of creativity, motivation, and time
- Leads to
 - Project malaise
 - Delays
 - Doing it the same way (again)



© 2003–2008 Mountain Goat Software®

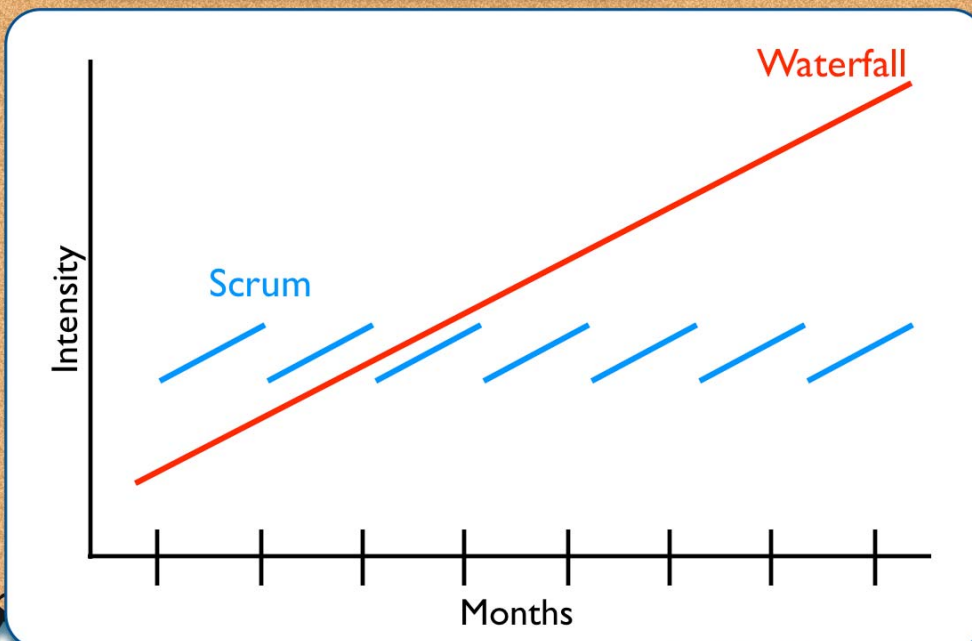
32

How agile handles waste

- 1 Timeboxing
- 2 Daily standups
- 3 Iteration retrospectives
- 4 Self-organizing teams



- 5 Spreading intensity evenly



Salesforce.com

Improvement in mean time to release for major releases

+61%

Increase in features delivered in major releases

+94%

Increase in features delivered per developer

+38%

Increase in major release cumulative value

+568%



Source: Scrum Gathering, April 16, 2008

© 2003–2008 Mountain Goat Software®

35

Sin #7: Myopia (Shortsightedness)

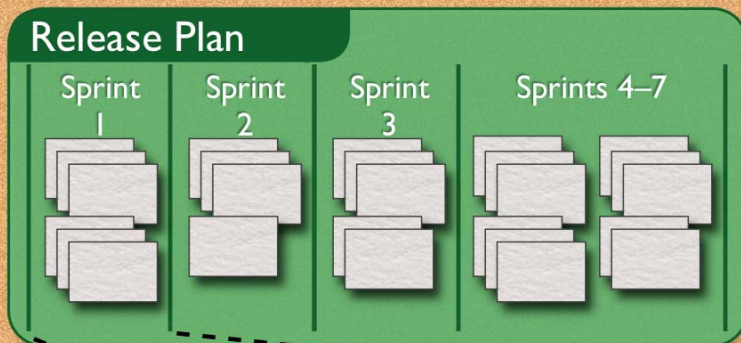
- Definition
 - Not seeing beyond your own work
- Experienced as
 - Teams who don't see the big picture
 - Individuals who work only within their roles
- Leads to
 - Unsuccessful products
 - Delays



© 2003–2008 Mountain Goat Software®

36

Seeing the forest and the trees at the same time



Task A	8 hours
Task B	16 hours
Task C	5 hours
Task D	8 hours

Iteration Plan



© 2003–2008 Mountain Goat Software®

37

Cross-functional team

- All disciplines necessary to go from idea to implementation
- Improves creativity and ownership
- Whole team commitment

But does that make everyone a generalist?



© 2003–2008 Mountain Goat Software®

38

Is everyone a generalist?

Meconi's

How that
here's
unch?

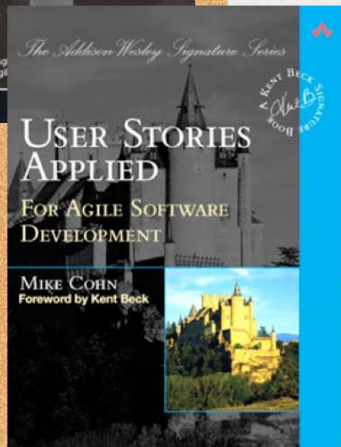
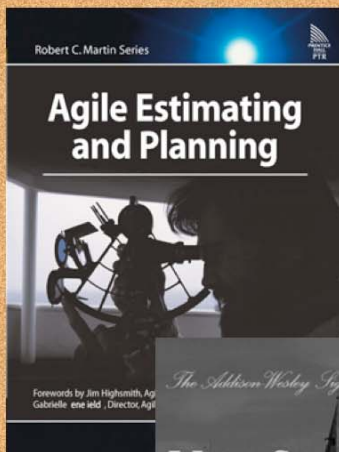
Key

- Order-taker
- Floater
- Sandwich-maker

Bruno's

Ferentino's

© 2003–2008 Mountain Goat Software®



Mike Cohn
 mike@mountaingoatsoftware.com
 www.mountaingoatsoftware.com
 (720) 890-6110 (office)
 (303) 810-2190 (mobile)

MOUNTAIN GOAT SOFTWARE

© 2003–2008 Mountain Goat Software®