Agile and the Seven Deadly Sins of Project Management

Mike Cohn

February 15, 2011

Mike Cohn - background

The Addison-Wesley Signature Series
User Stories Applied
For Agil
Developm
Mike Cohn
Foreword by Ken
Robert C. Martin Series
Agile Estimating and Planning
Forewords by Jim Highsmith, A
Gabrielle ene ield , Director, Ag
The Addison-Wesley Signature Series
Succeeding with Agile
Software Development Using Scrum
Mike Cohn
Foreword by Tim Lister

Agile coach and trainer

- Founding member and director of Agile Alliance and Scrum Alliance
- Founder of Mountain Goat Software
- Ran my first Scrum project in 1995
- Former VPE in four companies

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

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

The Scrum project community

Product Owner

Defines what to build and in what order

Scrum Master

Process coach and Sheepdog for the team

The team

Programmer

Programmer

Analyst

DBA

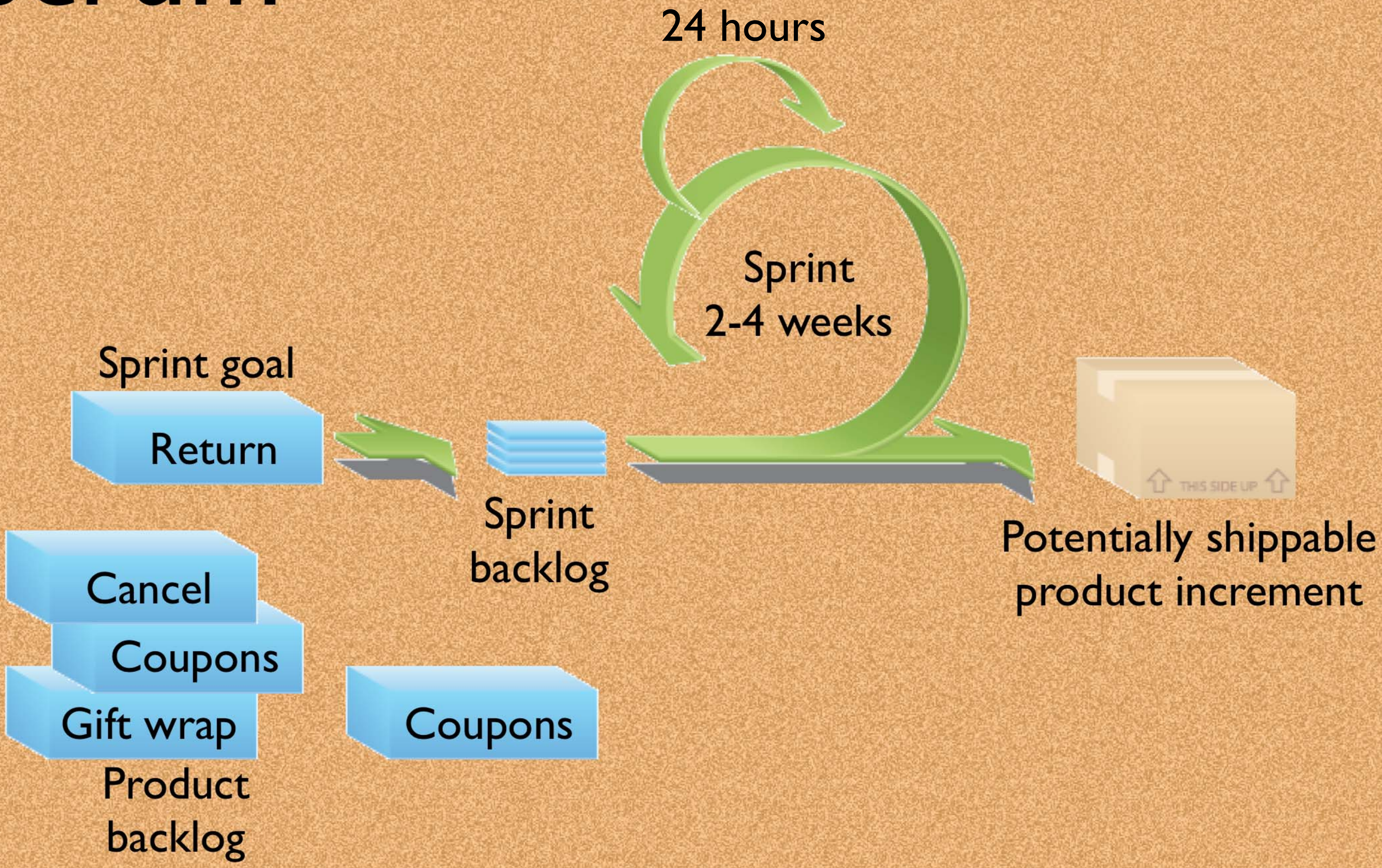
Tester

Programmer

Tester

User Experience Designer

Scrum


24 hours
Sprint
2-4 weeks
Sprint goal
Return
Sprint
backlog
Potentially shippable
product increment
THIS SIDE UP
Cancel
Coupons
Gift wrap
Coupons
Product
backlog

Seven Sins of Project Management

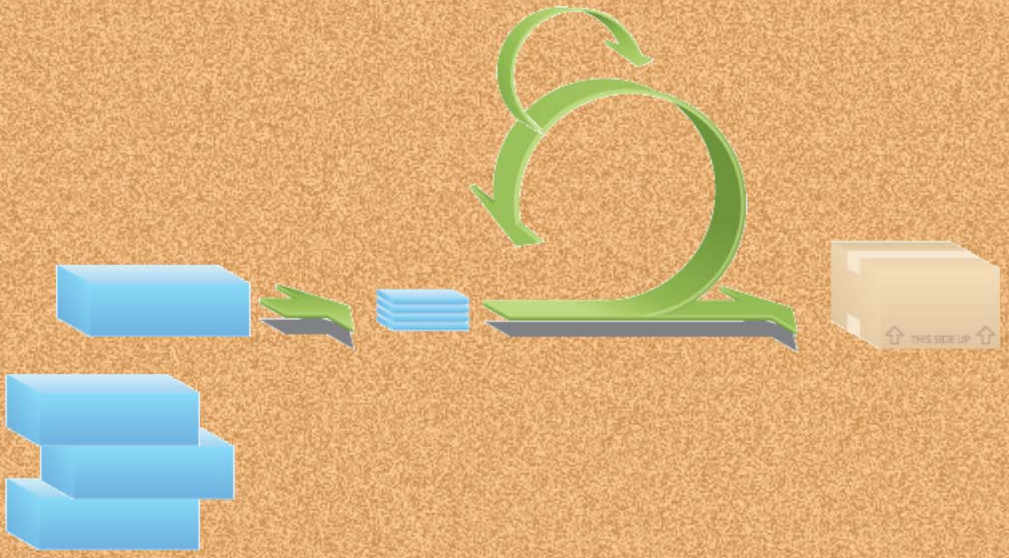
1. Gluttony
2. Lust
3. Sloth
4. Opaqueness
5. Pride
6. Wastefulness
7. Myopia

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

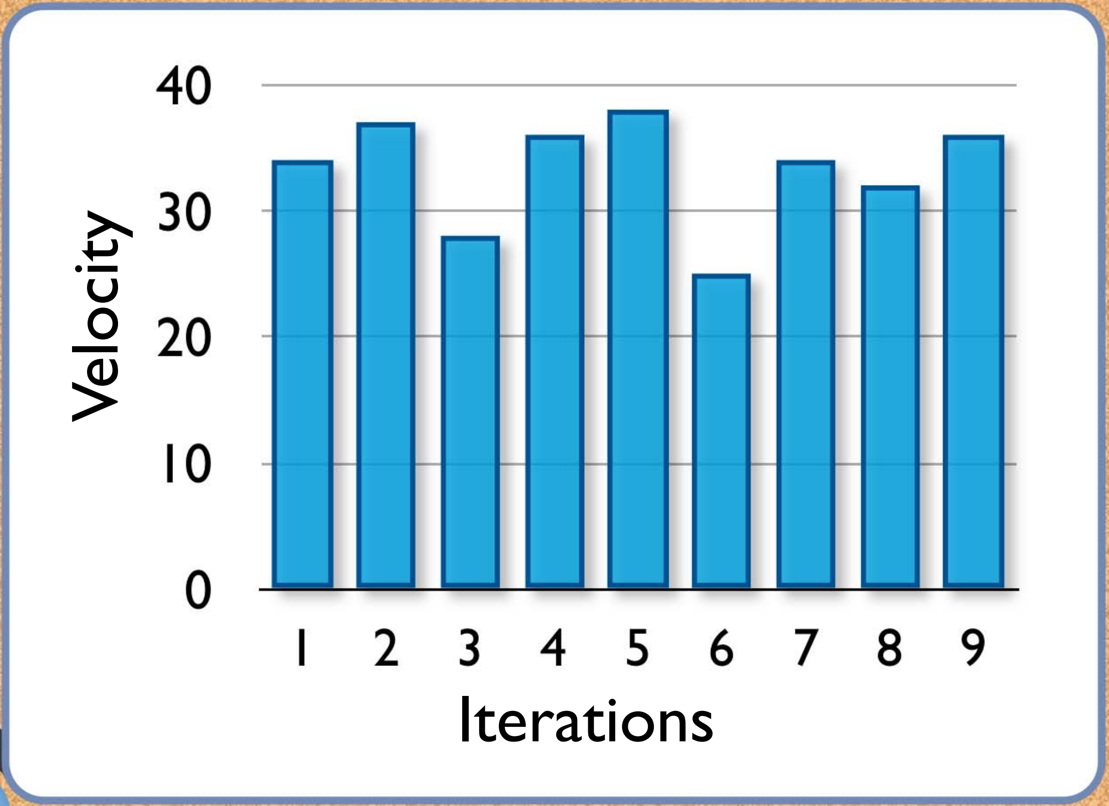
Timeboxes help avoid gluttony

- 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?”



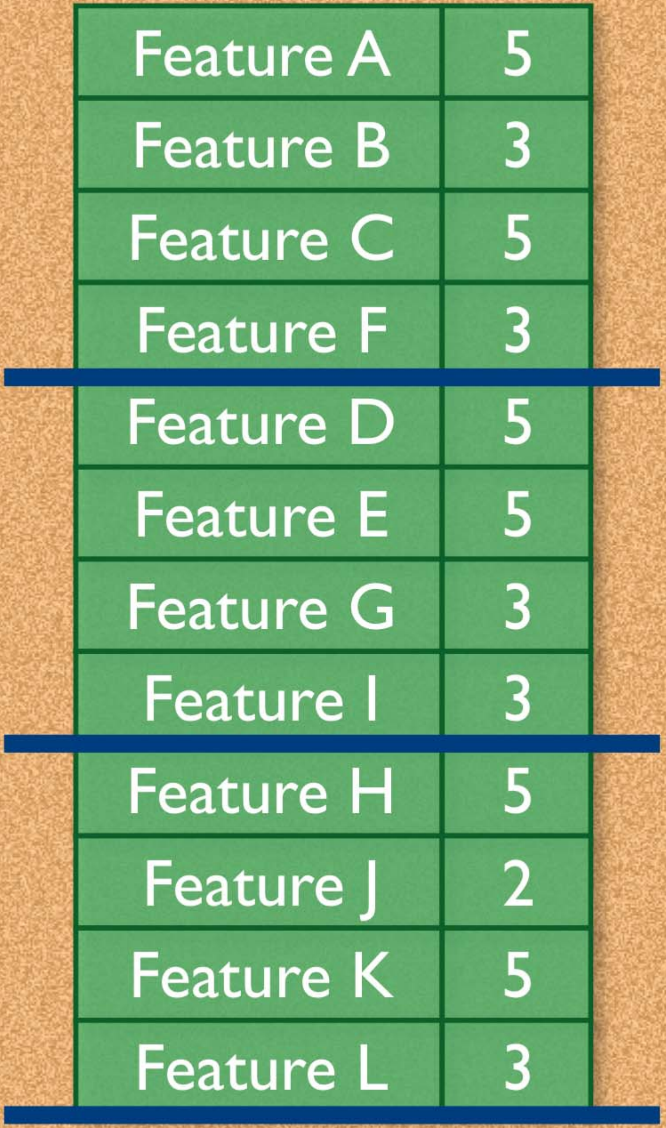
Timeboxed iterations increase predictability

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

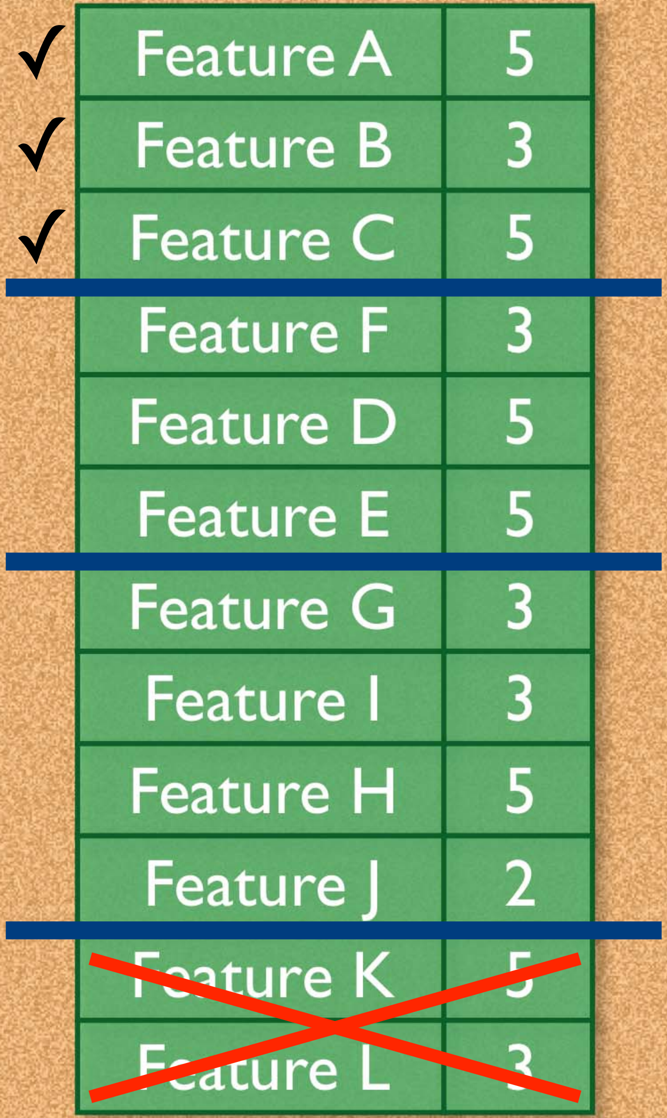


- Knowing this prevents the temptation to overcommit

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~~

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

Three ways agile deals with lust

1. Developing features in priority order

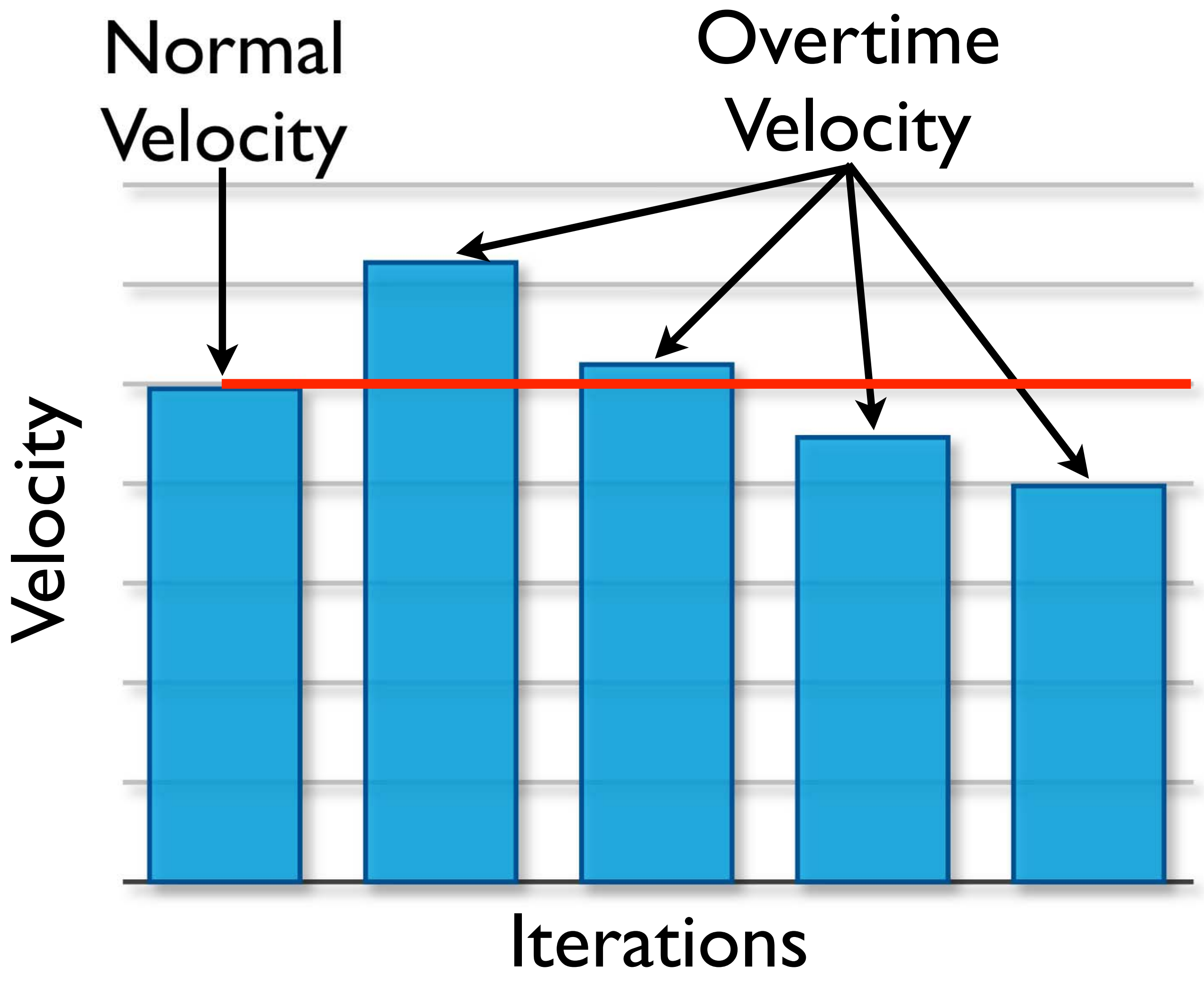
2. Incremental gratification

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

Old habits die hard



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

Agile practices related to quality

Simple design

Automated testing

Test-driven development

Continuous integration

Pair programming

Refactoring

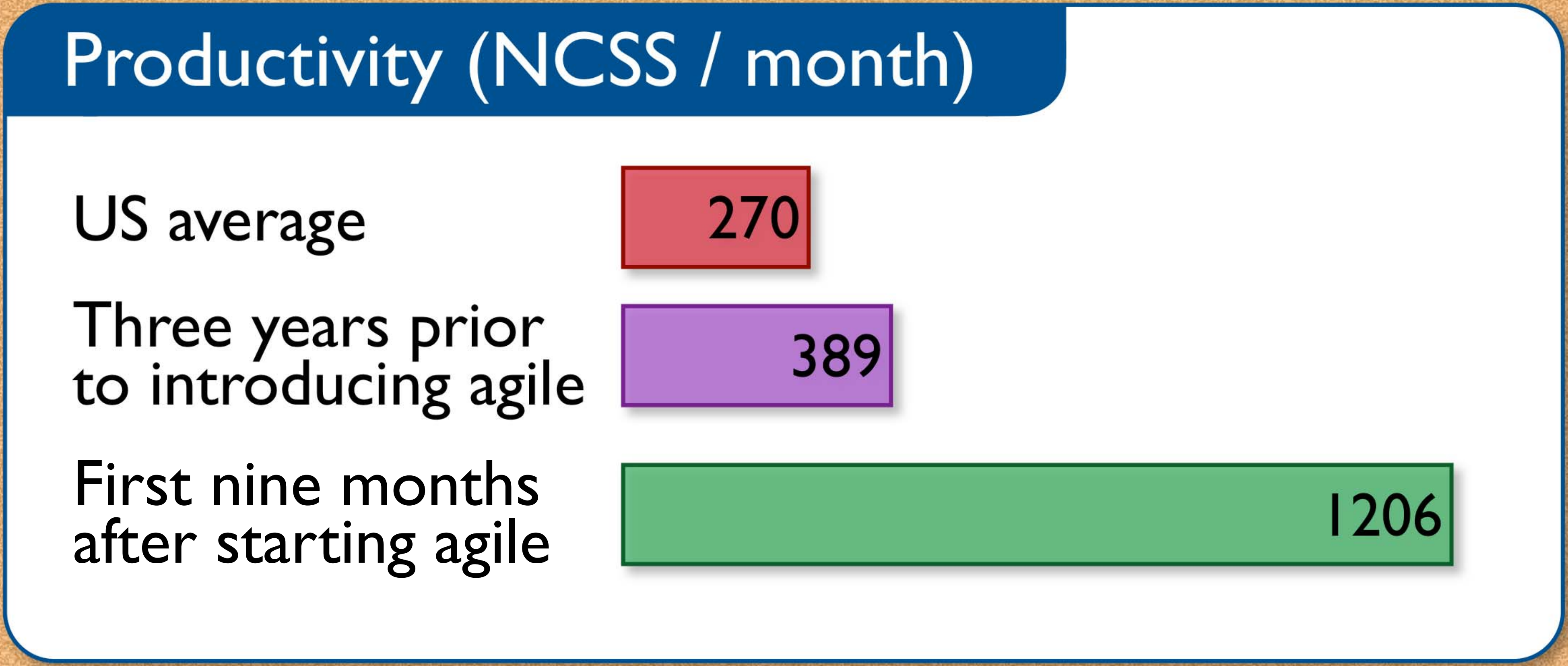
Case Study 1

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

Case Study 2

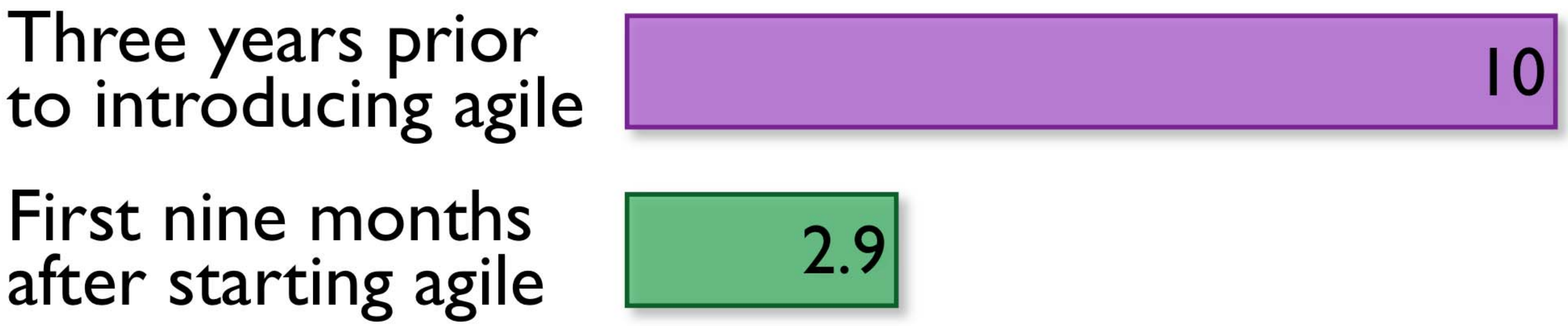
ePS



NCSS=Non-Comment Source Statment (Java)

ePS

Defects per KNCSS

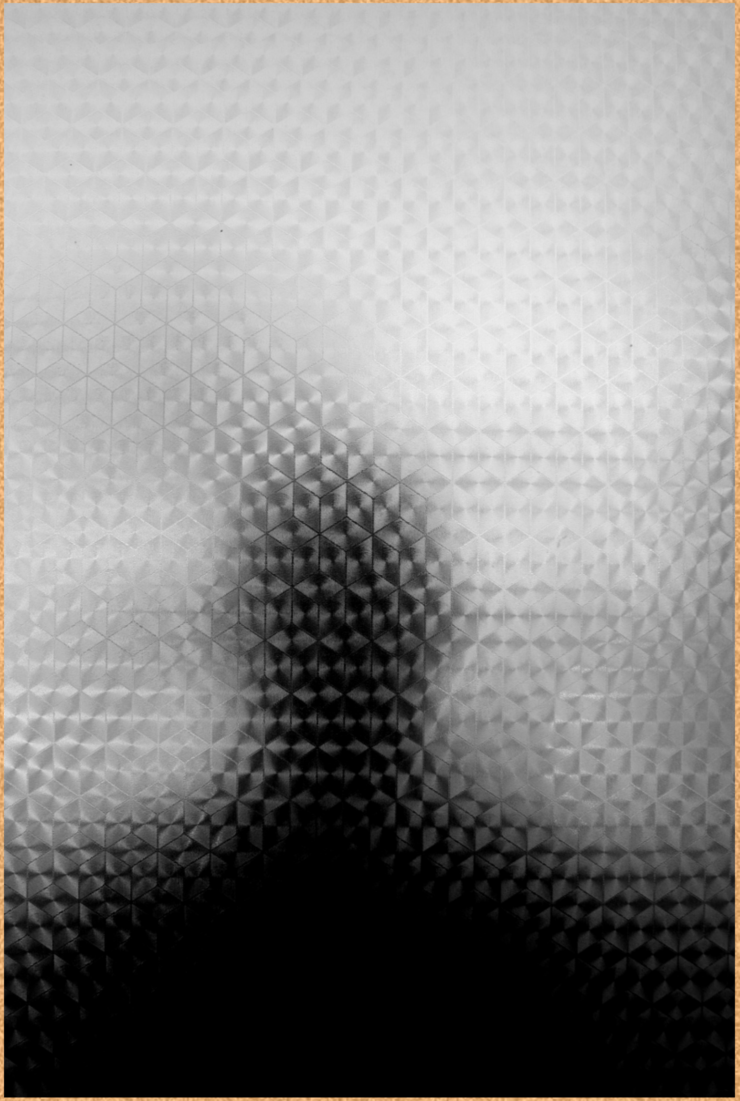


But wait, there's more...

- 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



Three types of opaqueness

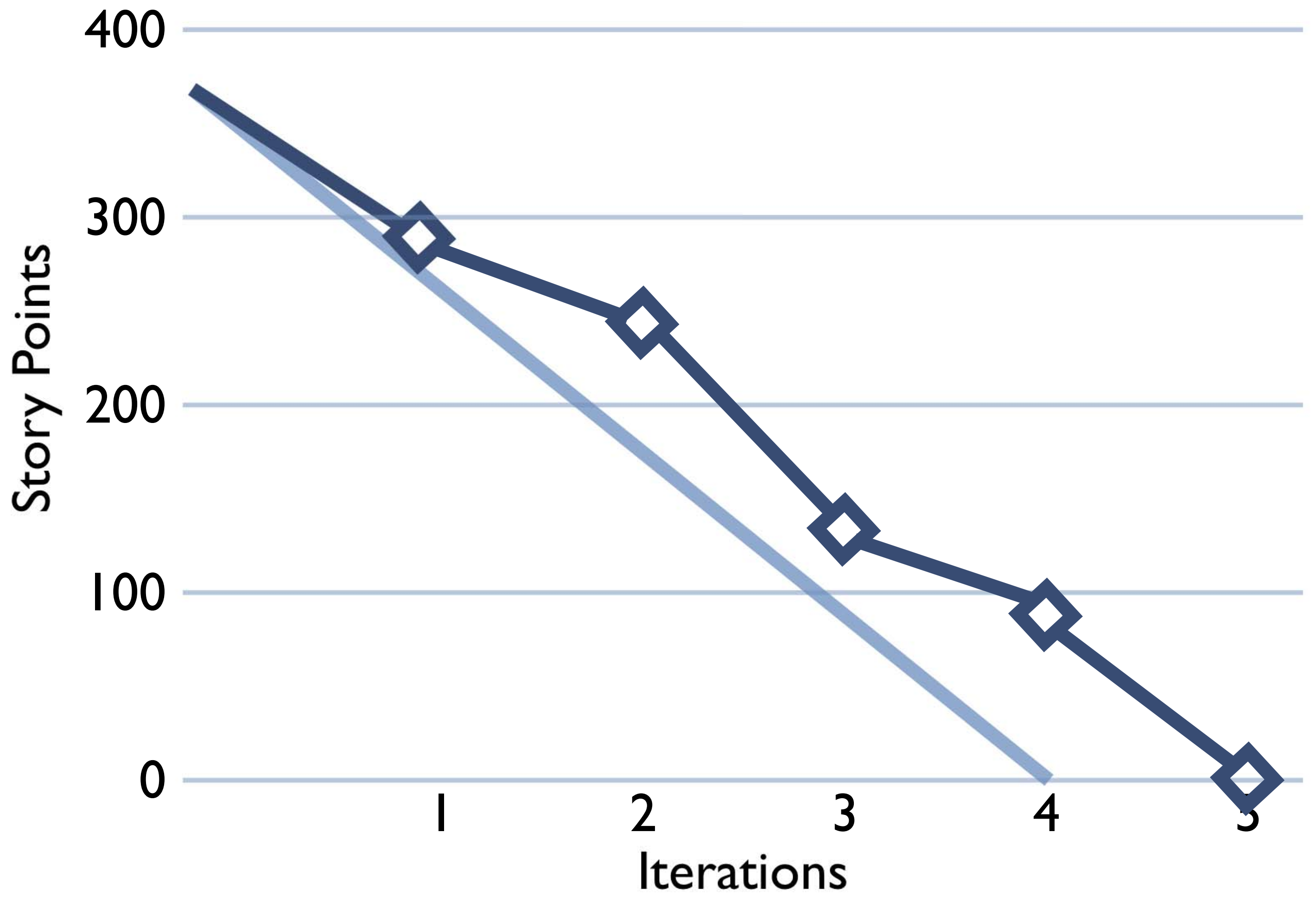
1 How agile addresses Quality opaqueness

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

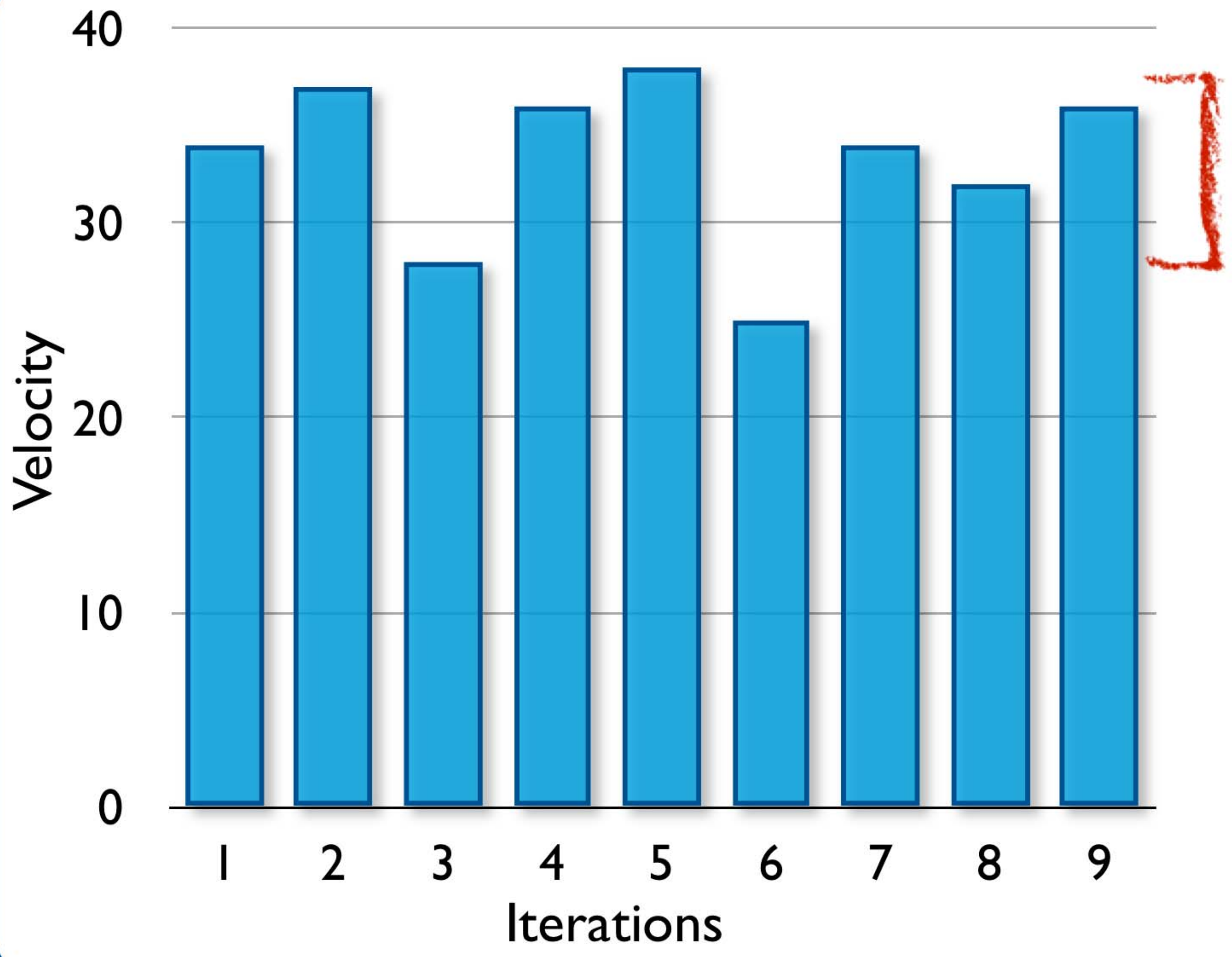
2

How agile addresses Schedule opaqueness

A release burndown chart

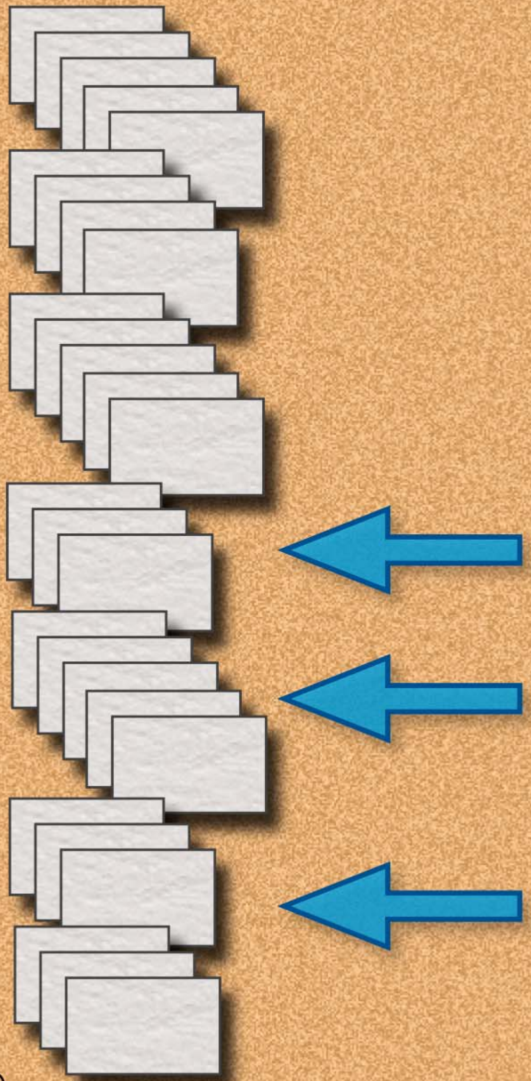


3 How agile addresses Scope opaqueness



A confidence interval (28–37 here)

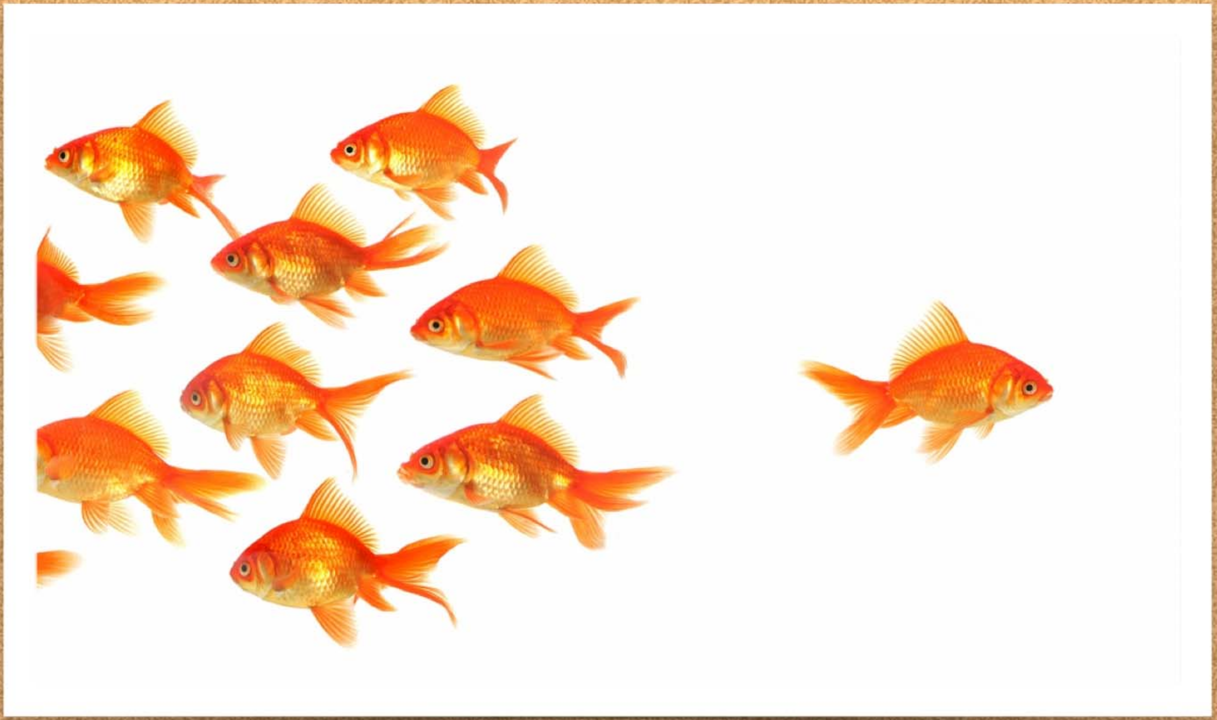
Using a confidence interval to predict what will be delivered


We'll almost certainly get here (5×28)
Based on our median, we'll finish here (5×33)
This is the most we can realistically expect (5×37)

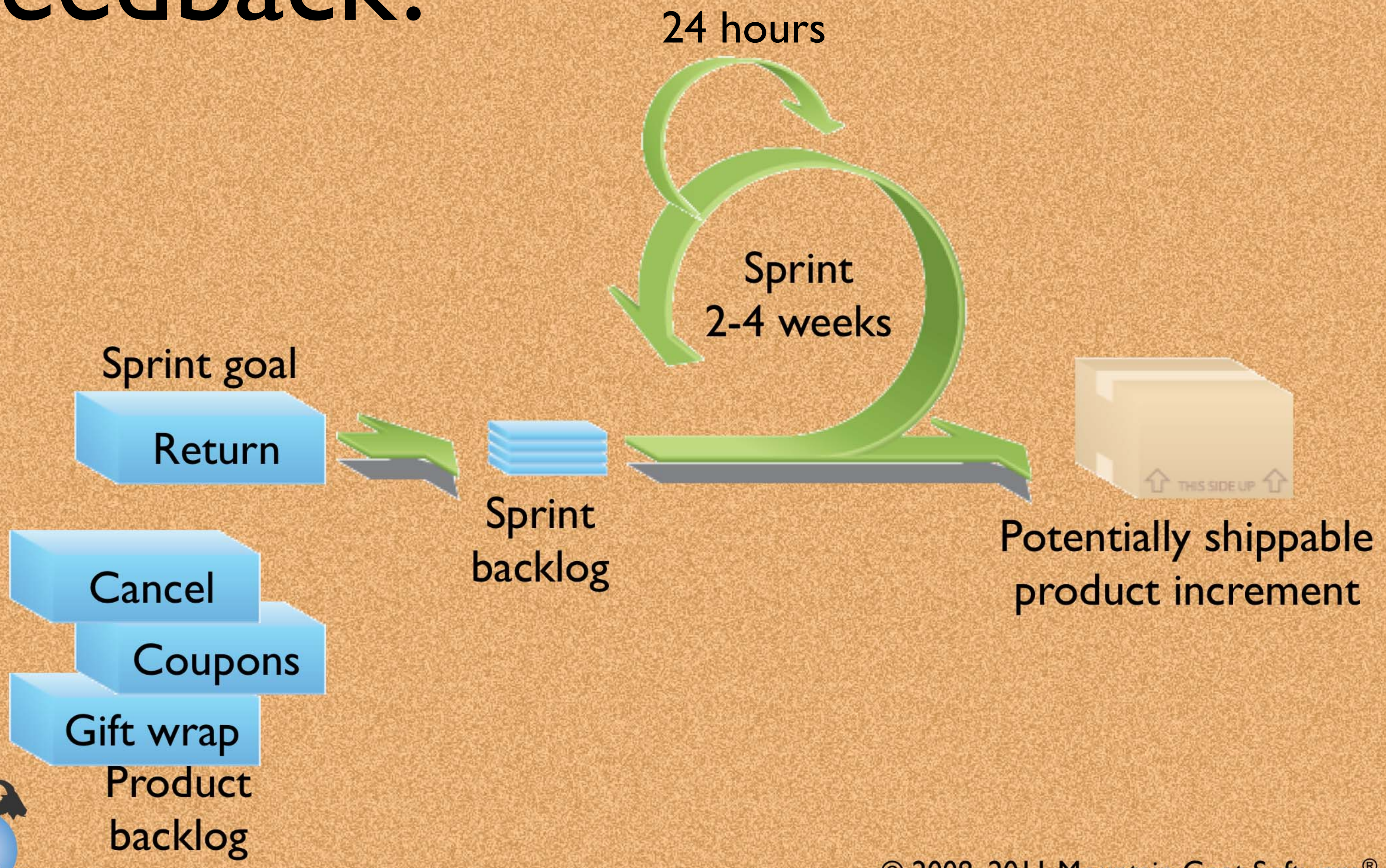
Assume:
There are five sprints left.

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



Where are opportunities for feedback?



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.

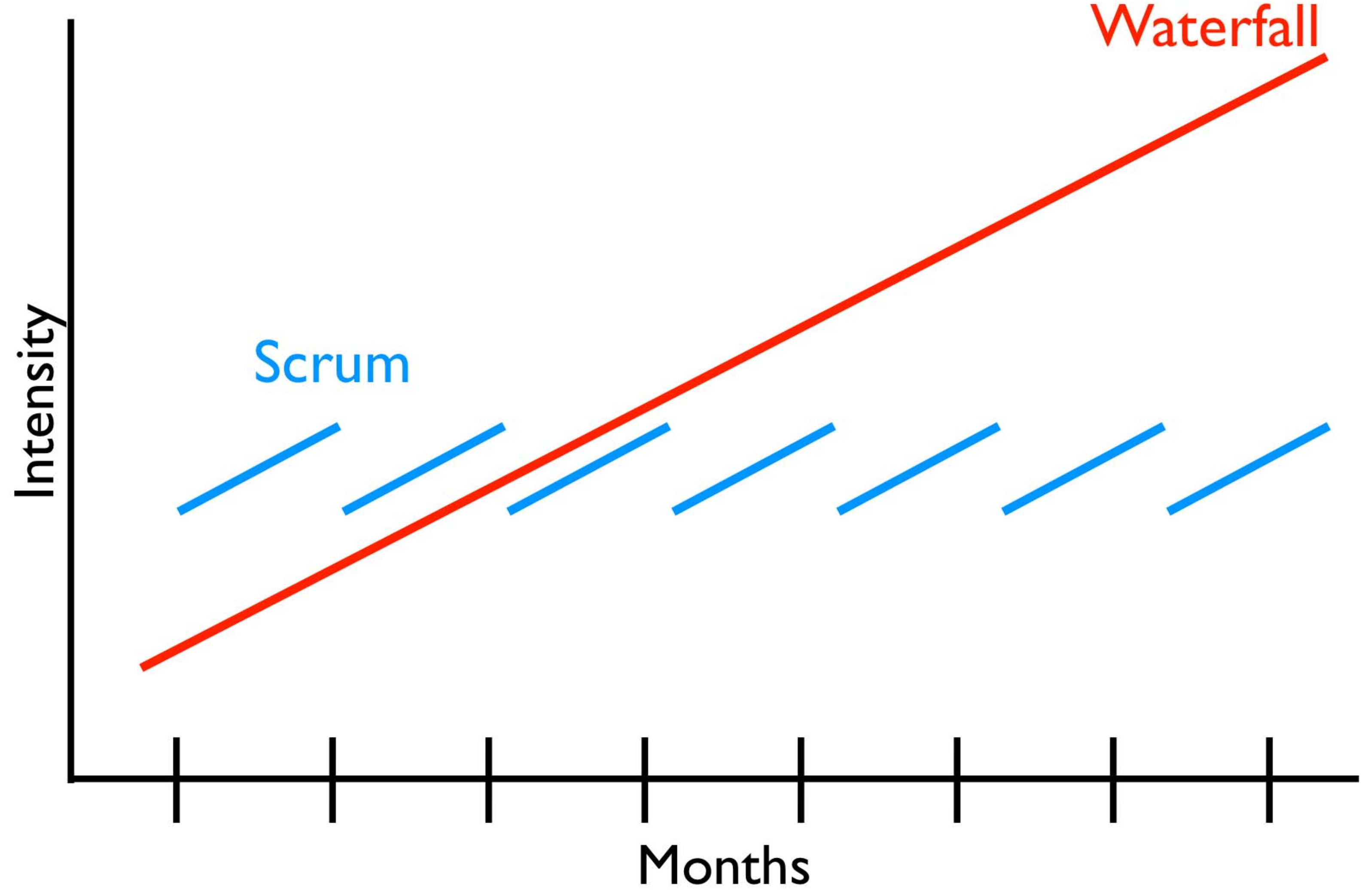
Sin #6: Wastefulness

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

How agile handles waste

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

5 Spreading intensity evenly



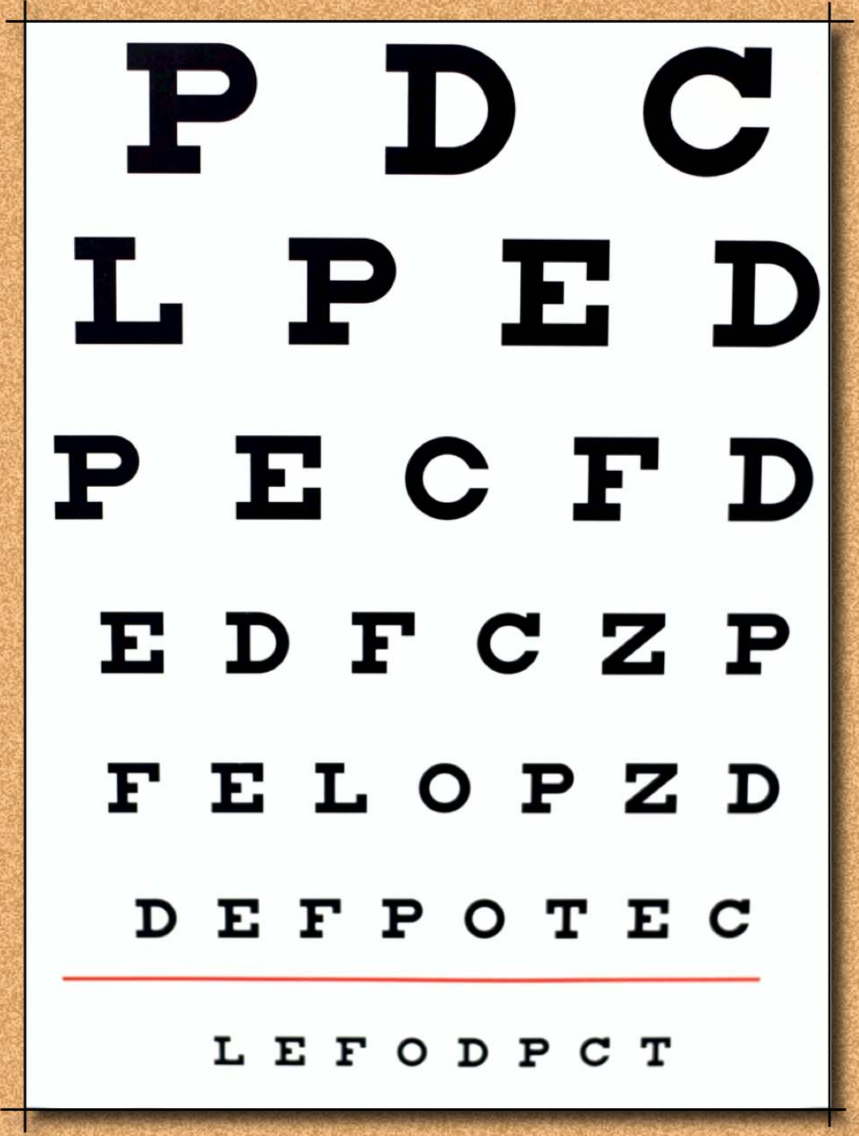
Salesforce.com

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%

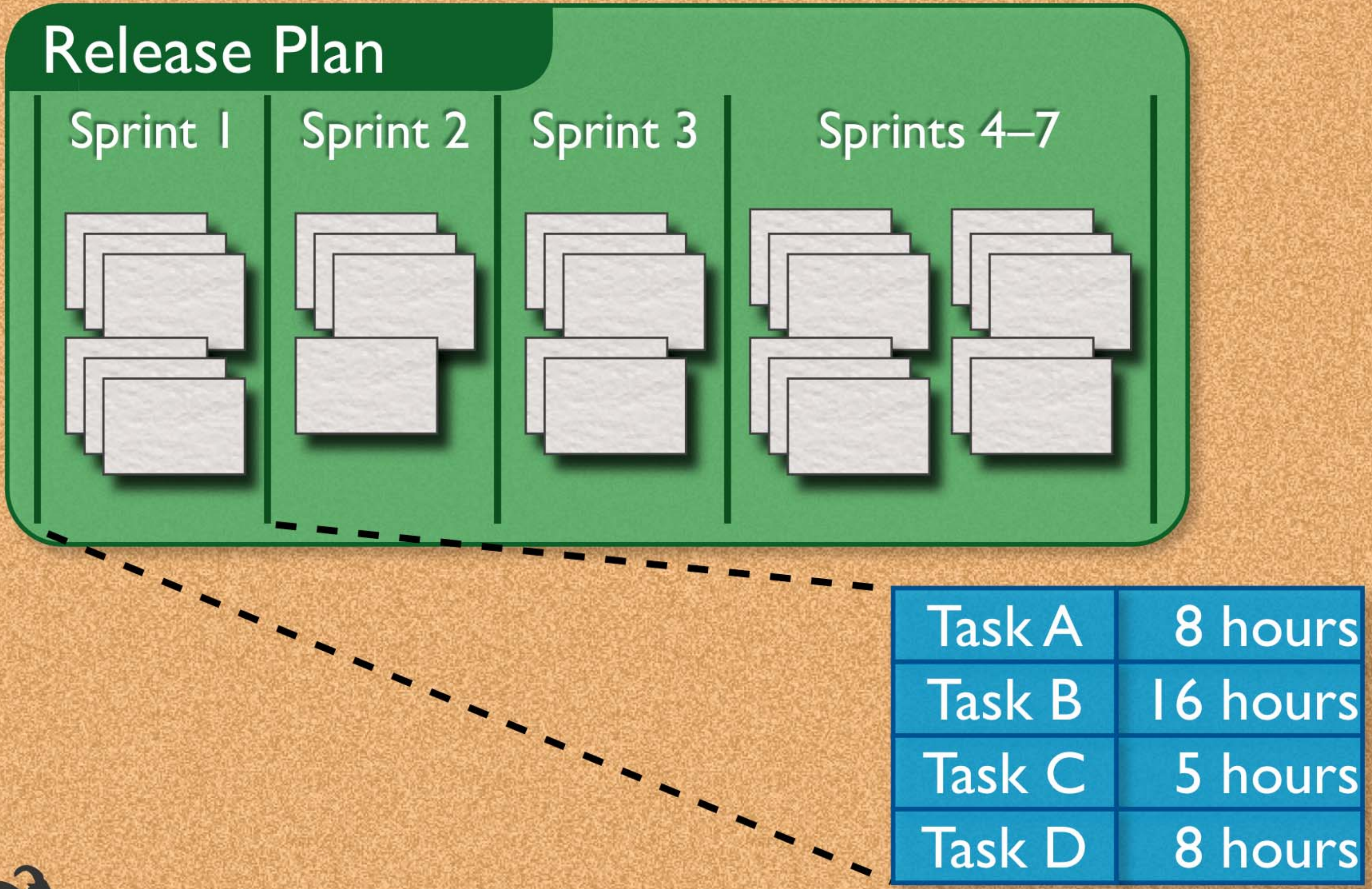
Source: Scrum Gathering, April 16, 2008

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



Seeing the forest and the trees at the same time



Iteration Plan

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

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?

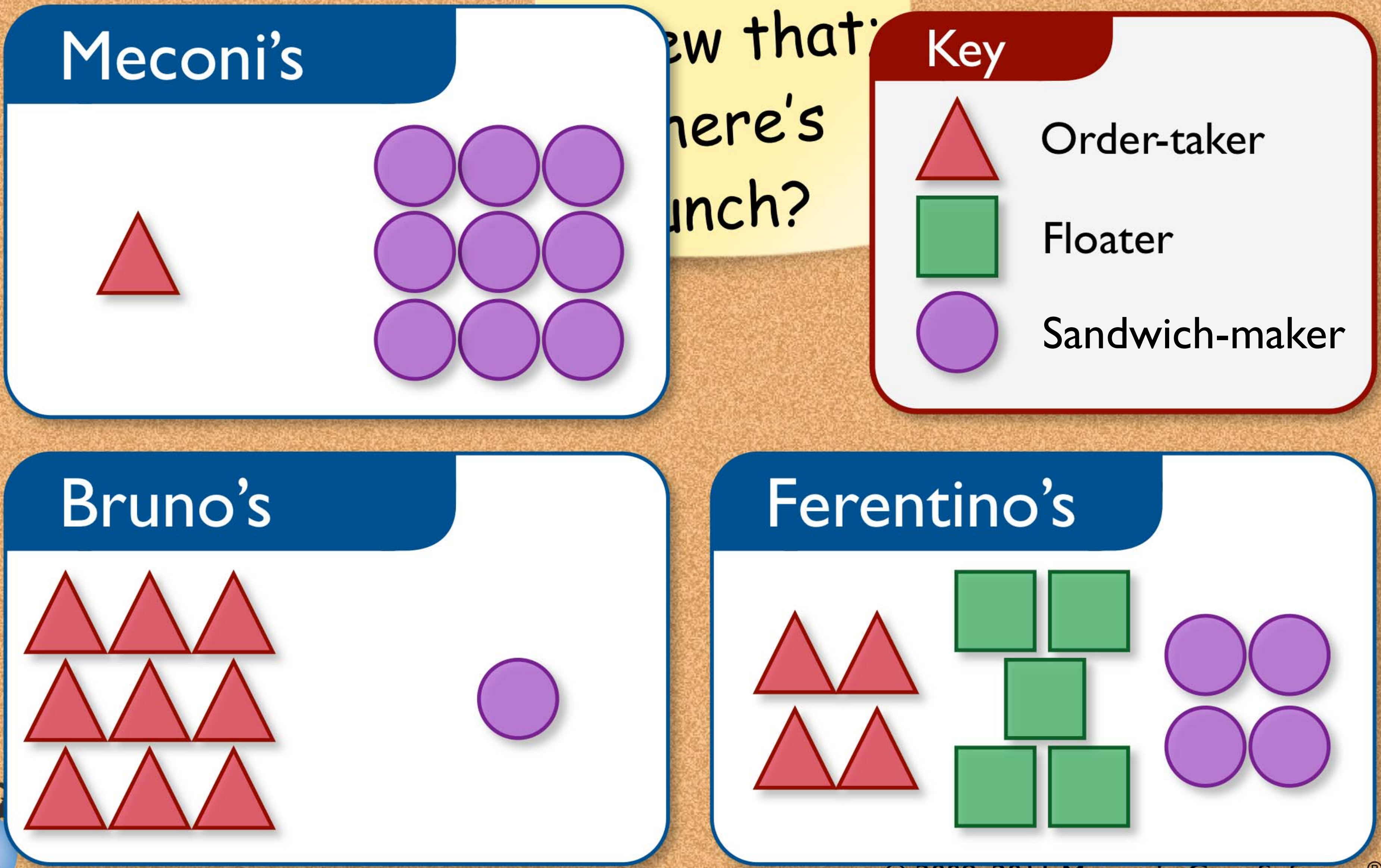
Is everyone a generalist?

~~Is everyone a generalist?~~

Screw that; where's lunch?

~~Is everyone a generalist?~~

ew that
ere's
nch?



Additional resources

Scrum Information

- www.MountainGoatSoftware.com/scrum
- www.MountainGoatSoftware.com/presentations
- www.ScrumAlliance.org

Training

- www.MountainGoatSoftware.com/training

Books

- *Agile Software Development with Scrum*, Ken Schwaber
- *Succeeding with Agile*, Mike Cohn

The Addison-Wesley Signature Series
A Kent Beck Signature Book
User Stories Applied
For Agile Software Development
Mike Cohn
Foreword by Kent Beck
Robert C. Martin Series
Agile Estimating and Planning
M
The Addison-Wesley Signature Series
A Mike Cohn Signature Book
Succeeding with Agile
Software Development Using Scrum
Mike Cohn
Foreword by Tim Lister

Mike Cohn

mike@mountaingoatsoftware.com

www.mountaingoatsoftware.com

twitter: mikewcohn

(888) 61−AGILE

MOUNTAIN GOAT
SOFTWARE