

# An Introduction to Scrum

Presented by

**Massimo Mecella**

MSE-CS - Advanced Programming



Mountain Goat Software, LLC



# We're losing the relay race

“The... ‘relay race’ approach to product development...may conflict with the goals of maximum speed and flexibility. Instead a holistic or ‘rugby’ approach—where a team tries to go the distance as a unit, passing the ball back and forth—may better serve today’s competitive requirements.”

Hirotaka Takeuchi and Ikujiro Nonaka, “The New New Product Development Game”, *Harvard Business Review*, January 1986.



# Scrum in 100 words

- Scrum is an agile process that allows us to focus on delivering the highest business value in the shortest time.
- It allows us to rapidly and repeatedly inspect actual working software (every two weeks to one month).
- The business sets the priorities. Teams self-organize to determine the best way to deliver the highest priority features.
- Every two weeks to a month anyone can see real working software and decide to release it as is or continue to enhance it for another sprint.



# Scrum origins

- Jeff Sutherland
  - Initial scrums at Easel Corp in 1993
  - IDX and 500+ people doing Scrum
- Ken Schwaber
  - ADM
  - Scrum presented at OOPSLA 96 with Sutherland
  - Author of three books on Scrum
- Mike Beedle
  - Scrum patterns in PLOPD4
- Ken Schwaber and Mike Cohn
  - Co-founded Scrum Alliance in 2002, initially within the Agile Alliance



# Scrum has been used by:

- Microsoft
- Yahoo
- Google
- Electronic Arts
- IBM
- Lockheed Martin
- Philips
- Siemens
- Nokia
- Capital One
- BBC
- Intuit
- Nielsen Media
- First American Real Estate
- BMC Software
- Ipswitch
- John Deere
- Lexis Nexis
- Sabre
- Salesforce.com
- Time Warner
- Turner Broadcasting
- Oce



# Scrum has been used for:

- Commercial software
- In-house development
- Contract development
- Fixed-price projects
- Financial applications
- ISO 9001-certified applications
- Embedded systems
- 24x7 systems with 99.999% uptime requirements
- the Joint Strike Fighter
- Video game development
- FDA-approved, life-critical systems
- Satellite-control software
- Websites
- Handheld software
- Mobile phones
- Network switching applications
- ISV applications
- Some of the largest applications in use



# Characteristics

- Self-organizing teams
- Product progresses in a series of 2-4 week “sprints”
- Requirements are captured as items in a list of “product backlog”
- No specific engineering practices prescribed
- Uses generative rules to create an agile environment for delivering projects
- One of the “agile processes”



# The Agile Manifesto—a statement of values

Individuals and interactions

over

Process and tools

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

Mountain Goat Software, LLC



# The Agile Manifesto—a statement of values

Individuals and interactions

Working software

over

over

Process and tools

Comprehensive documentation

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



Mountain Goat Software, LLC



# The Agile Manifesto—a statement of values

Individuals and interactions

Working software

Customer collaboration

over

over

over

Process and tools

Comprehensive documentation

Contract negotiation

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



Mountain Goat Software, LLC



# The Agile Manifesto—a statement of values

Individuals and interactions

Working software

Customer collaboration

Responding to change

over

over

over

over

Process and tools

Comprehensive documentation

Contract negotiation

Following a plan

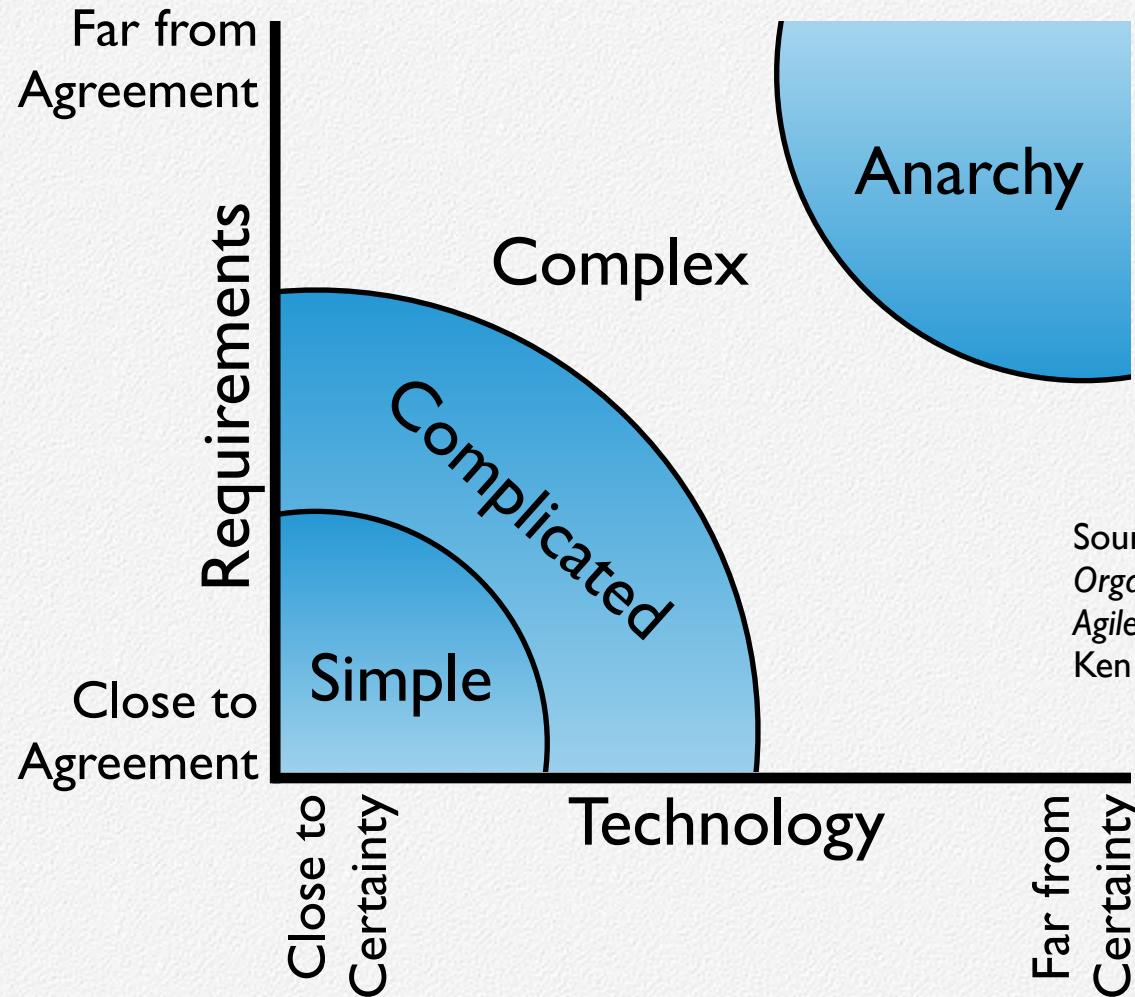
Source: [www.agilemanifesto.org](http://www.agilemanifesto.org)



Mountain Goat Software, LLC

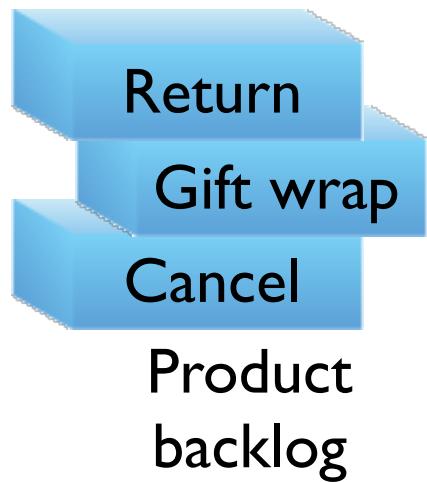


# Project noise level

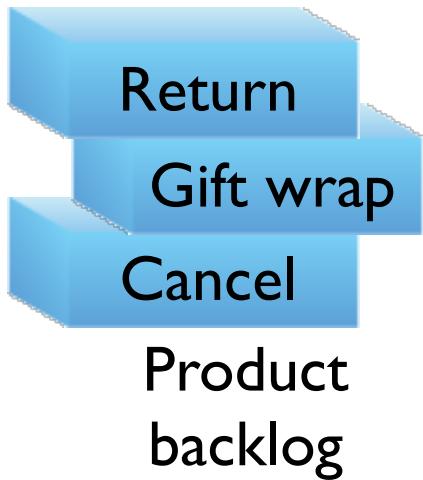


Source: *Strategic Management and Organizational Dynamics* by Ralph Stacey in *Agile Software Development with Scrum* by Ken Schwaber and Mike Beedle.

# Scrum



# Scrum



Product  
backlog



Mountain Goat Software, LLC



# Scrum

Sprint goal

Feature I

Gift wrap

Cancel

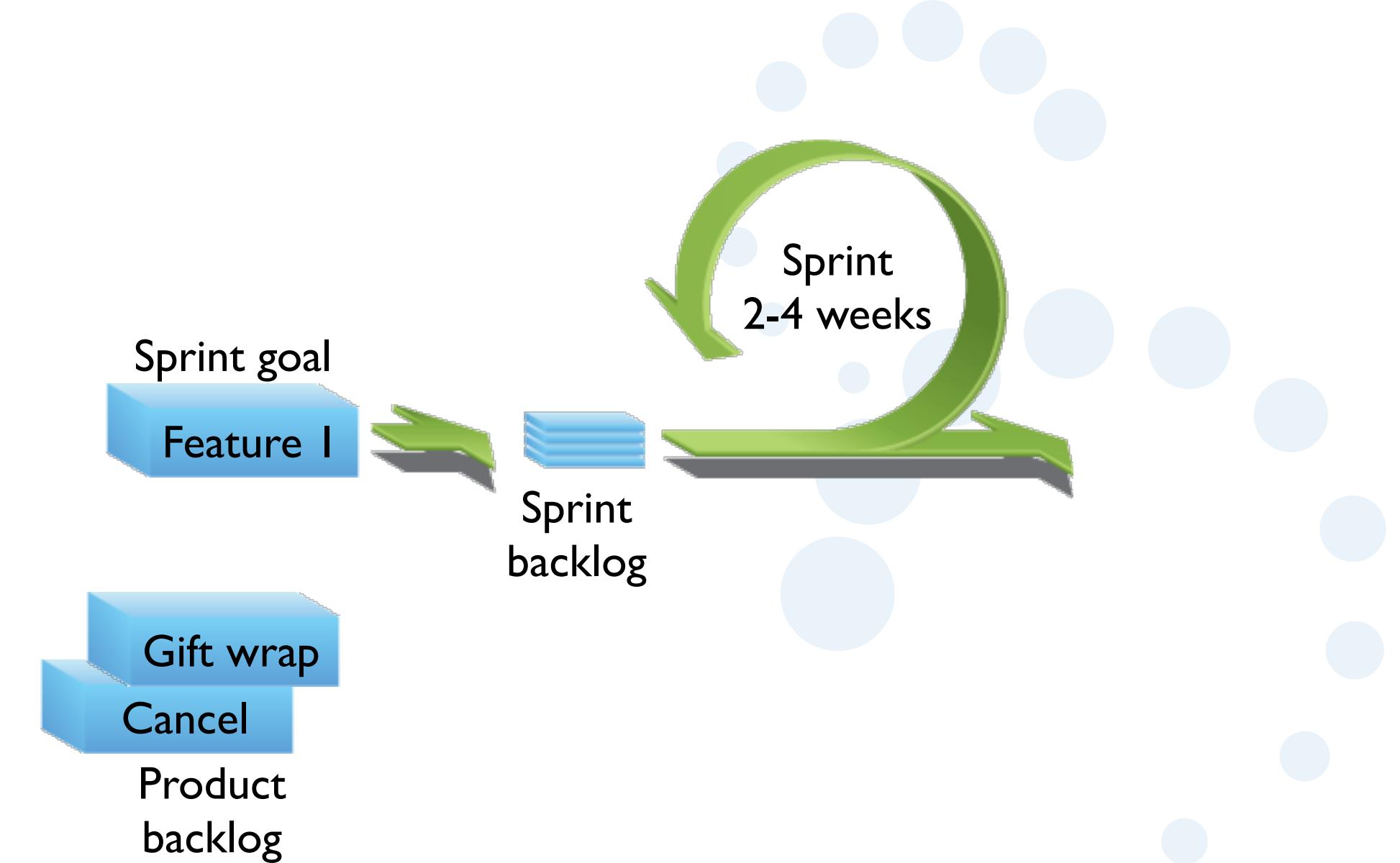
Product  
backlog



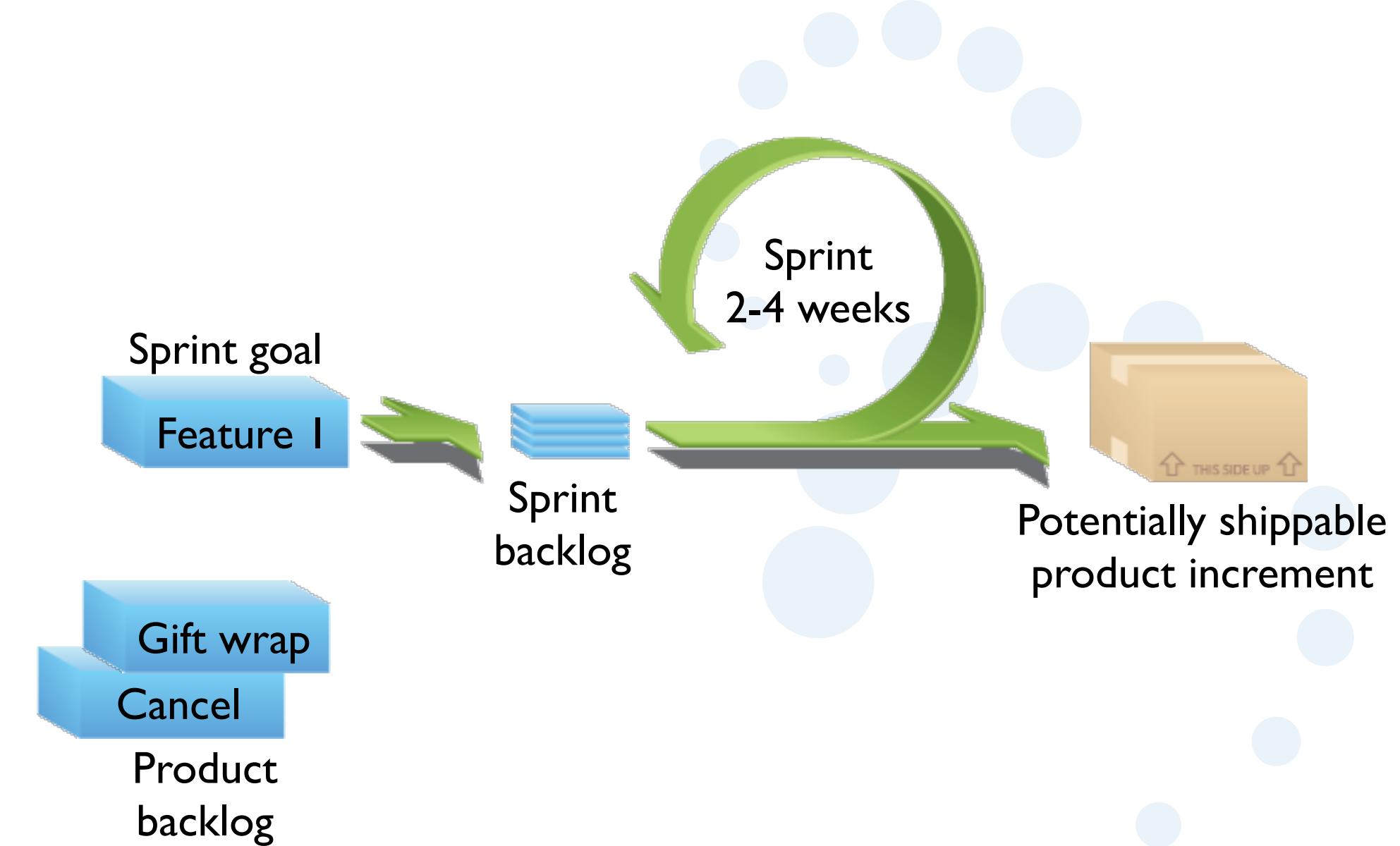
Mountain Goat Software, LLC



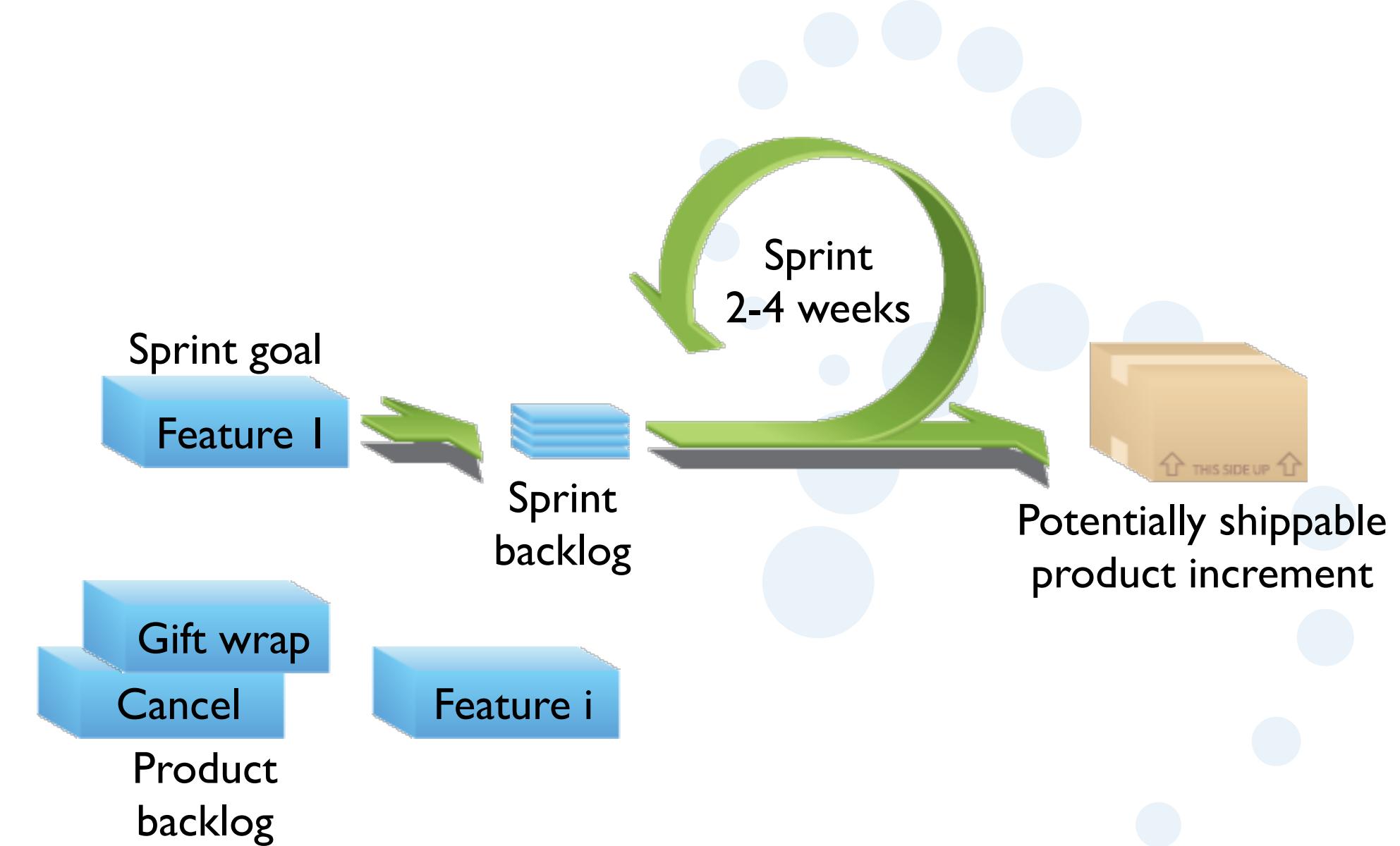
# Scrum



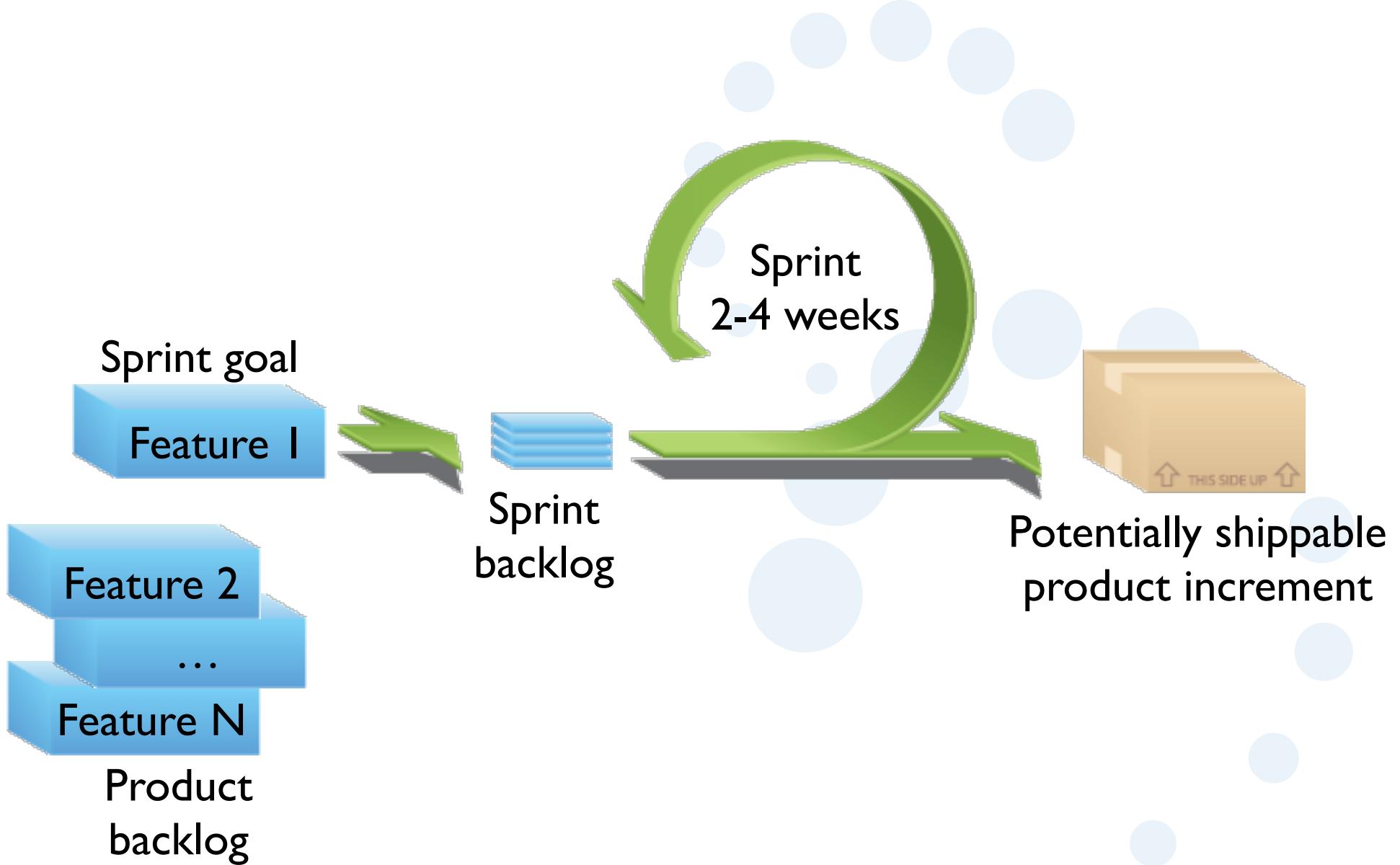
# Scrum



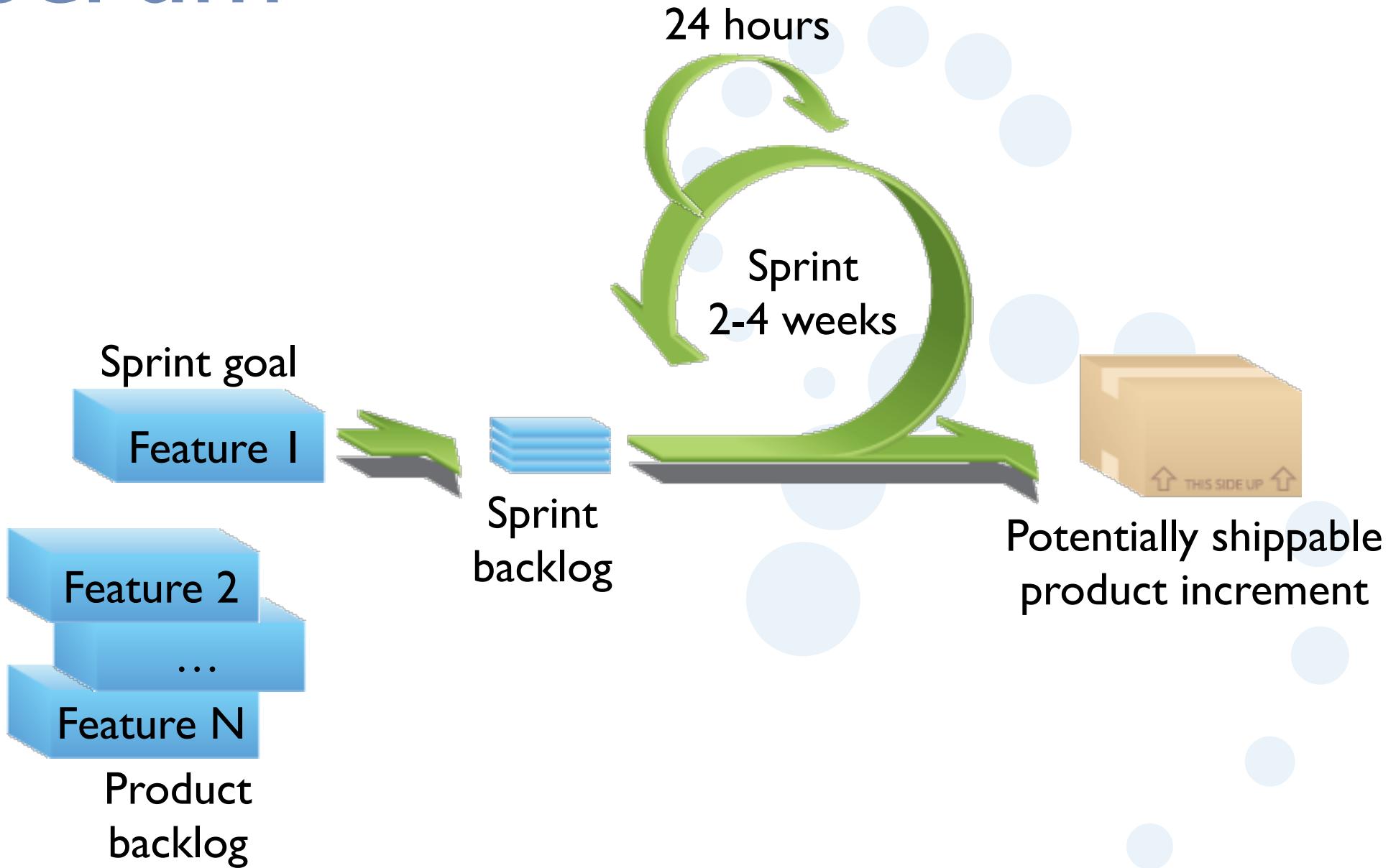
# Scrum



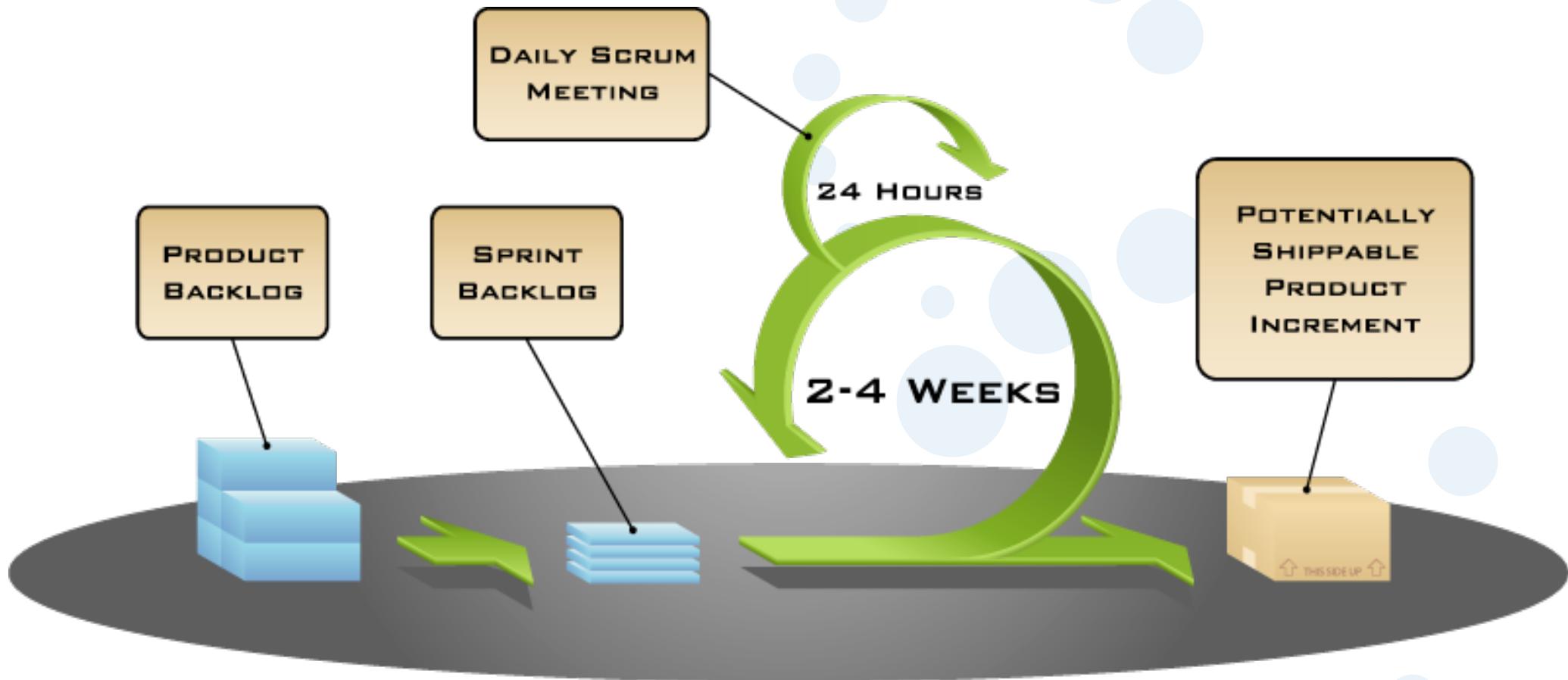
# Scrum



# Scrum



# Putting it all together



COPYRIGHT © 2005, MOUNTAIN GOAT SOFTWARE

Image available at  
[www.mountaingoatsoftware.com/scrum](http://www.mountaingoatsoftware.com/scrum)



Mountain Goat Software, LLC



# Sprints

- Scrum projects make progress in a series of “sprints”
  - Analogous to Extreme Programming iterations
  - Typical duration is 2–4 weeks or a calendar month at most
  - A constant duration leads to a better rhythm
  - Product is designed, coded, and tested during the sprint



# Sequential vs. overlapping development

Requirements

Design

Code

Test

Rather than doing all of  
one thing at a time...

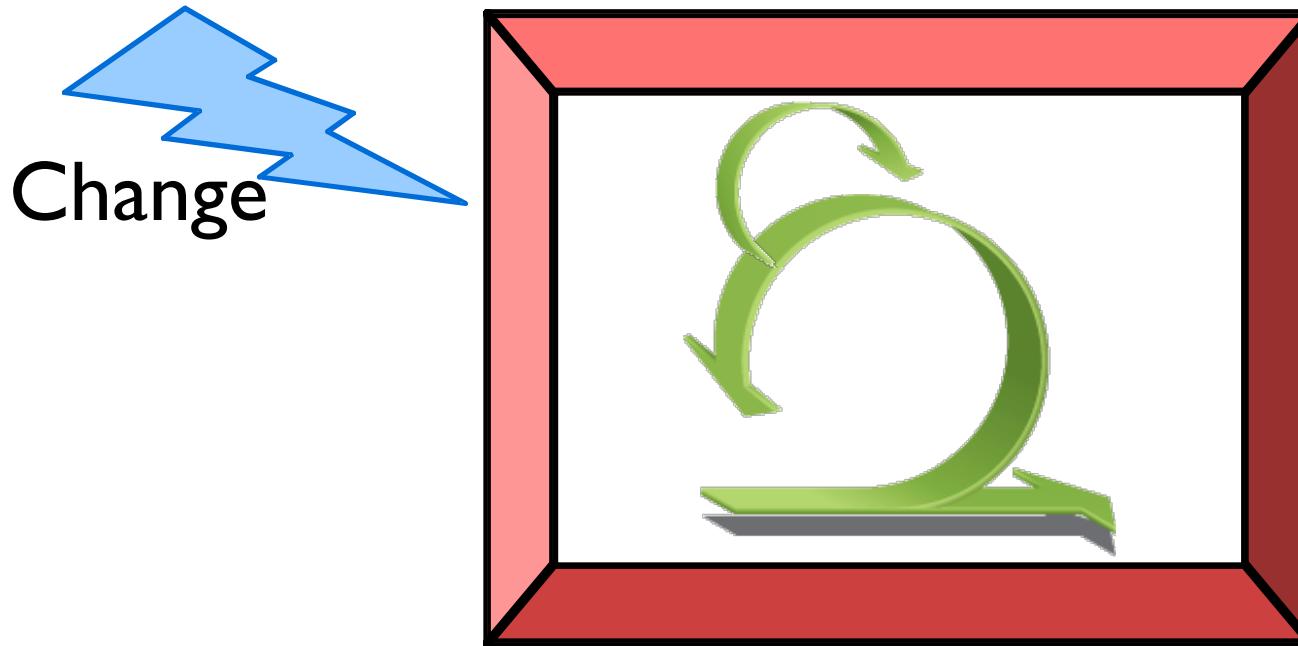
...Scrum teams do a little  
of everything all the time

Source: "The New New Product Development Game" by Takeuchi and Nonaka. *Harvard Business Review*, January 1986.

Mountain Goat Software, LLC



# No changes during a sprint



- Plan sprint durations around how long you can commit to keeping change out of the sprint



# Scrum framework

## Roles

- Product owner
- ScrumMaster
- Team

## Ceremonies

- Sprint planning
- Sprint review
- Sprint retrospective
- Daily scrum meeting

## Artifacts

- Product backlog
- Sprint backlog
- Burndown charts



# Scrum framework

## Roles

- Product owner
- ScrumMaster
- Team

## Ceremonies

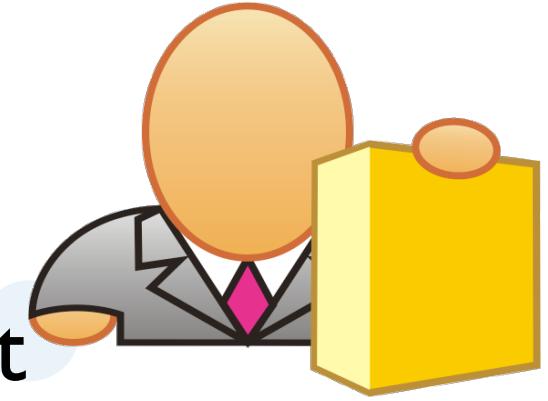
- Sprint planning
- Sprint review
- Sprint retrospective
- Daily scrum meeting

## Artifacts

- Product backlog
- Sprint backlog
- Burndown charts



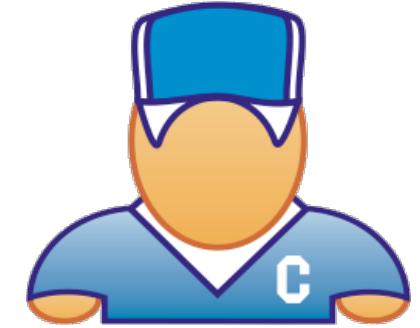
# Product owner



- Define the features of the product
- Decide on release date and content
- Be responsible for the profitability of the product (ROI)
- Prioritize features according to market value
- Adjust features and priority every iteration, as needed
- Accept or reject work results



# The ScrumMaster

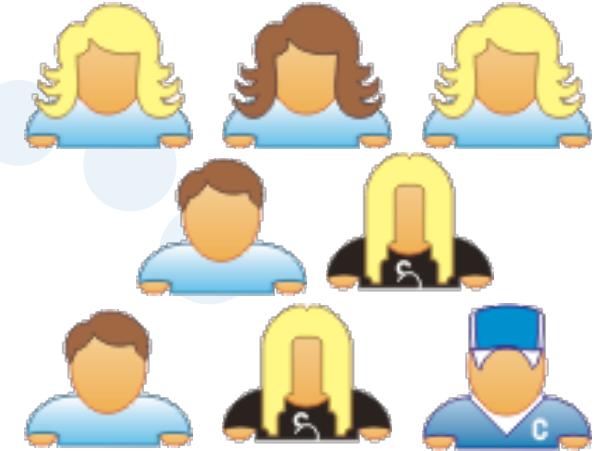


- Represents management to the project
- Responsible for enacting Scrum values and practices
- Removes impediments
- Ensure that the team is fully functional and productive
- Enable close cooperation across all roles and functions
- Shield the team from external interferences



# The team

- Typically 5-9 people
- Cross-functional:
  - Programmers, testers, user experience designers, etc.
- Members should be full-time
  - May be exceptions (e.g., database administrator)
- Teams are self-organizing
  - Ideally, no titles but rarely a possibility
- Membership should change only between sprints



# Scrum framework

## Roles

- Product owner
- ScrumMaster
- Team

## Ceremonies

- Sprint planning
- Sprint review
- Sprint retrospective
- Daily scrum meeting

## Artifacts

- Product backlog
- Sprint backlog
- Burndown charts



# Sprint planning meeting

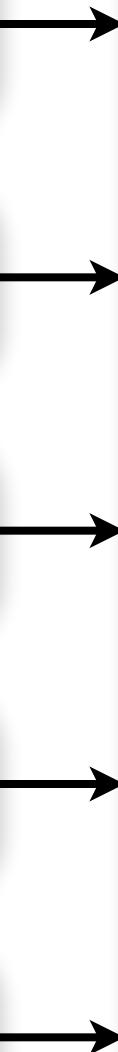
Team capacity

Product backlog

Business conditions

Current product

Technology



Team capacity

Product backlog

Business conditions

Current product

Technology

## Sprint planning meeting

### Sprint prioritization

- Analyze and evaluate product backlog
- Select sprint goal



# Sprint planning meeting

## Sprint prioritization

- Analyze and evaluate product backlog
- Select sprint goal

Team capacity

Product backlog

Business conditions

Current product

Technology

Sprint goal



Team capacity

Product backlog

Business conditions

Current product

Technology

## Sprint planning meeting

### Sprint prioritization

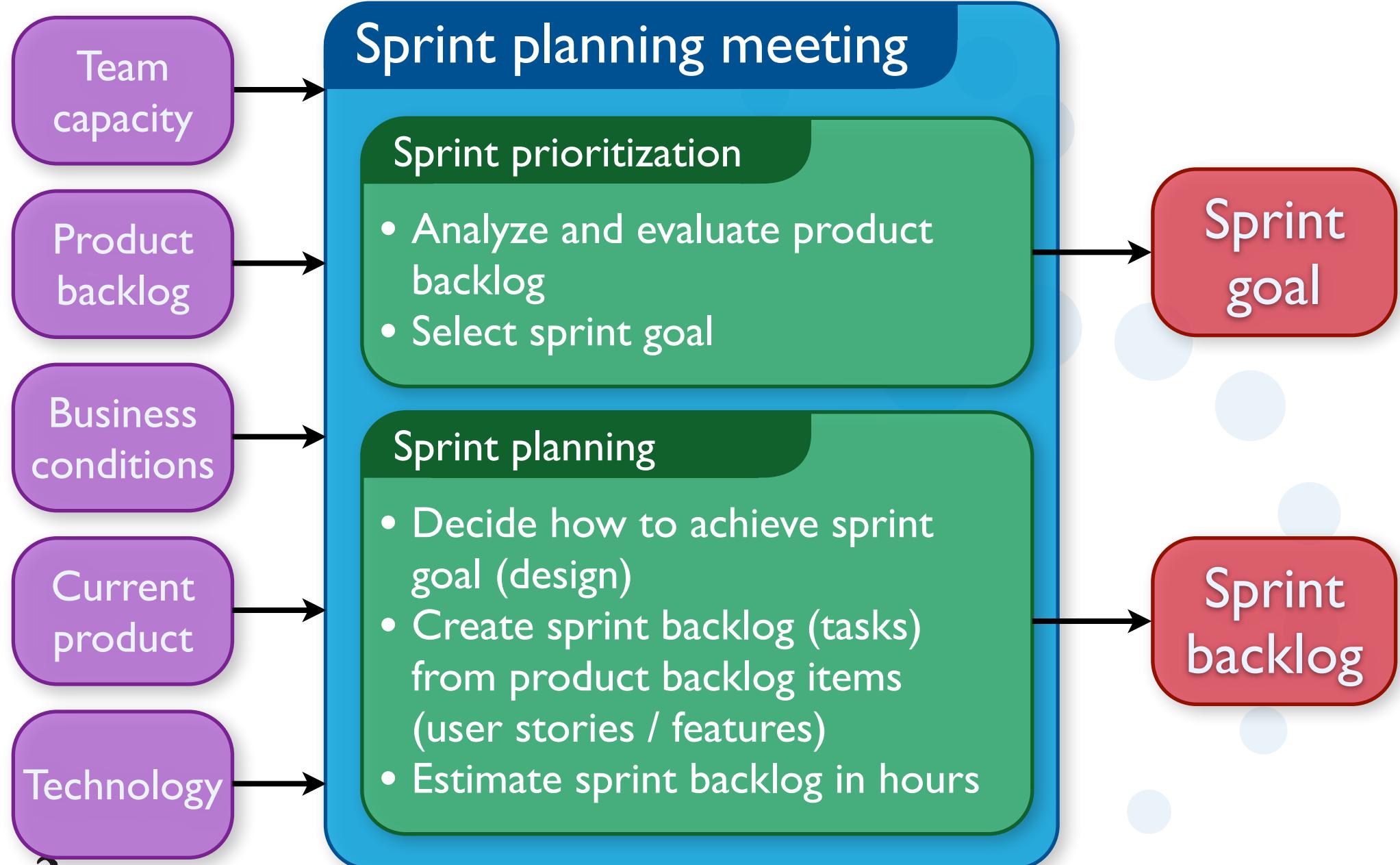
- Analyze and evaluate product backlog
- Select sprint goal

### Sprint planning

- Decide how to achieve sprint goal (design)
- Create sprint backlog (tasks) from product backlog items (user stories / features)
- Estimate sprint backlog in hours

Sprint goal





# Sprint planning

- Team selects items from the product backlog they can commit to completing
- Sprint backlog is created
  - Tasks are identified and each is estimated (1-16 hours)
  - Collaboratively, not done alone by the ScrumMaster
- High-level design is considered

As a vacation planner, I want to see photos of the hotels.



Code the middle tier (8 hours)  
Code the user interface (4)  
Write test fixtures (4)  
Code the foo class (6)  
Update performance tests (4)



# The daily scrum

- Parameters
  - Daily
  - 15-minutes
  - Stand-up
- Not for problem solving
  - Whole world is invited
  - Only team members, ScrumMaster, product owner, can talk
- Helps avoid other unnecessary meetings



# Everyone answers 3 questions

1

What did you do yesterday?

2

What will you do today?

3

Is anything in your way?

- These are **not** status for the ScrumMaster
  - They are commitments in front of peers



# The sprint review

- Team presents what it accomplished during the sprint
- Typically takes the form of a demo of new features or underlying architecture
- Informal
  - 2-hour prep time rule
  - No slides
- Whole team participates
- Invite the world



# Sprint retrospective

- Periodically take a look at what is and is not working
- Typically 15–30 minutes
- Done after every sprint
- Whole team participates
  - ScrumMaster
  - Product owner
  - Team
  - Possibly customers and others



# Start / Stop / Continue

- Whole team gathers and discusses what they'd like to:

Start doing

Stop doing

Continue doing

This is just one  
of many ways  
to do a sprint  
retrospective.



# Scrum framework

## Roles

- Product owner
- ScrumMaster
- Team

## Ceremonies

- Sprint planning
- Sprint review
- Sprint retrospective
- Daily scrum meeting

## Artifacts

- Product backlog
- Sprint backlog
- Burndown charts



# Product backlog

- The requirements
- A list of all desired work on the project
- Ideally expressed such that each item has value to the users or customers of the product
- Prioritized by the product owner
- Reprioritized at the start of each sprint



This is the  
product backlog



Mountain Goat Software, LLC



# User stories

Instead of Use Cases, Agile project owners do "user stories"

**Who** (user role) – Is this a customer, employee, admin, etc.?

**What** (goal) – What functionality must be achieved/developed?

**Why** (reason) – Why does user want to accomplish this goal?

As a [user role], I want to [goal], so I can [reason].

Example:

"As a user, I want to log in, so I can access subscriber content."

**story points**: Rating of effort needed to implement this story

common scales: 1-10, shirt sizes (XS, S, M, L, XL), etc.



# A sample product backlog

Backlog item	Estimate
Allow a guest to make a reservation	3
As a guest, I want to cancel a reservation.	5
As a guest, I want to change the dates of a reservation.	3
As a hotel employee, I can run RevPAR reports (revenue-per-available-room)	8
Improve exception handling	8
...	30
...	50



# Real example

## Product Backlog Estimating System Upgrade

Sprint	ID	Backlog Item	Owner	Estimate (days)	Remaining (days)
1	1 Minor	Remove user kludge in .dpr file	BC	1	1
1	2 Minor	Remove cMap/cMenu/cMenuSize from disciplines.pas	BC	1	1
1	3 Minor	Create "Legacy" discipline node with old civils and E&I content	BC	1	1
1	4 Major	Augment each tbl operation to support network operation	BC	10	10
1	5 Major	Extend Engineering Design estimate items to include summaries	BC	2	2
1	6 Super	Supervision/Guidance	CAM	4	4
	7 Minor	Remove Custodian property from AppConfig class in globals.pas	BC	1	
	8 Minor	Remove LOC_ constants in globals.pas and main.pas	BC	1	
	9 Minor	New E&I section doesn't have lblCaption set	BC	1	
	10 Minor	Delay in main.releaseform doesn't appear to be required	BC	1	
	11 Minor	Undo modifications to Other Major Equipment in formExcel.pas	BC	1	
	12 Minor	AJACS form to be centred on the screen	BC	1	
	13 Major	Extend DUnit tests to all 40 disciplines	BC	6	



# The sprint goal

- A short statement of what the work will be focused on during the sprint

Database Application

Make the application run on SQL Server in addition to Oracle.

Life Sciences

Support features necessary for population genetics studies.

Financial services

Support more technical indicators than company ABC with real-time, streaming data.



# Managing the sprint backlog

- Individuals sign up for work of their own choosing
  - Work is never assigned
- Estimated work remaining is updated daily
- Any team member can add, delete or change the sprint backlog
- Work for the sprint emerges
- If work is unclear, define a sprint backlog item with a larger amount of time and break it down later
- Update work remaining as more becomes known



# A sprint backlog

Tasks	Mon	Tues	Wed	Thur	Fri
Code the user interface	8				
Code the middle tier	16				
Test the middle tier	8				
Write online help	12				
Write the foo class	8				



# A sprint backlog

Tasks	Mon	Tues	Wed	Thur	Fri
Code the user interface	8	4			
Code the middle tier	16	12			
Test the middle tier	8	16			
Write online help	12				
Write the foo class	8	8			



# A sprint backlog

Tasks	Mon	Tues	Wed	Thur	Fri
Code the user interface	8	4	8		
Code the middle tier	16	12	10		
Test the middle tier	8	16	16		
Write online help	12				
Write the foo class	8	8	8		
Add error logging				8	



# A sprint backlog

Tasks	Mon	Tues	Wed	Thur	Fri
Code the user interface	8	4	8		
Code the middle tier	16	12	10	4	
Test the middle tier	8	16	16	11	
Write online help	12				
Write the foo class	8	8	8	8	
Add error logging			8	4	



# A sprint backlog

Tasks	Mon	Tues	Wed	Thur	Fri
Code the user interface	8	4	8		
Code the middle tier	16	12	10	4	
Test the middle tier	8	16	16	11	8
Write online help	12				
Write the foo class	8	8	8	8	8
Add error logging			8	4	



# Real sprint backlog

## Sprint 1

01/11/2004

		Sprint Day	1	2	3	4	5	6	7
		Mo	Tu	We	Th	Fr	Sa	Su	

**19 days work in this sprint**

		Hours remaining	152	152	152	152	152	152	152
--	--	-----------------	-----	-----	-----	-----	-----	-----	-----

Backlog Item	Backlog Item	Owner	Estimate	1	2	3	4	5	6	7
				Mo	Tu	We	Th	Fr	Sa	Su
1 Minor	Remove user kludge in .dpr file	BC	8	8	8	8	8	8	8	8
2 Minor	Remove cMap/cMenu/cMenuSize from disciplines.pas	BC	8	8	8	8	8	8	8	8
3 Minor	Create "Legacy" discipline node with old civils and E&I content	BC	8	8	8	8	8	8	8	8
4 Major	Augment each tbl operation to support network operation	BC	80	80	80	80	80	80	80	80
5 Major	Extend Engineering Design estimate items to include summaries	BC	16	16	16	16	16	16	16	16
6 Super	Supervision/Guidance	CAM	32	32	32	32	32	32	32	32

## Sprint 1

01/11/2004

		Sprint Day	1	2	3	4	5	6	7
		Mo	Tu	We	Th	Fr	Sa	Su	

**19 days work in this sprint**

		Hours remaining	152	150	140	130	118	118	118
--	--	-----------------	-----	-----	-----	-----	-----	-----	-----

Backlog Item	Backlog Item	Owner	Estimate	1	2	3	4	5	6	7
				Mo	Tu	We	Th	Fr	Sa	Su
1 Minor	Remove user kludge in .dpr file	BC	8	8	8	4	2	0		
2 Minor	Remove cMap/cMenu/cMenuSize from disciplines.pas	BC	8	8	8	4	0			
3 Minor	Create "Legacy" discipline node with old civils and E&I content	BC	8	8	8	6	0			
4 Major	Augment each tbl operation to support network operation	BC	80	80	80	80	78	78	78	
5 Major	Extend Engineering Design estimate items to include summaries	BC	16	16	16	16	16	16	16	
6 Super	Supervision/Guidance	CAM	32	32	28	26	24	24	24	



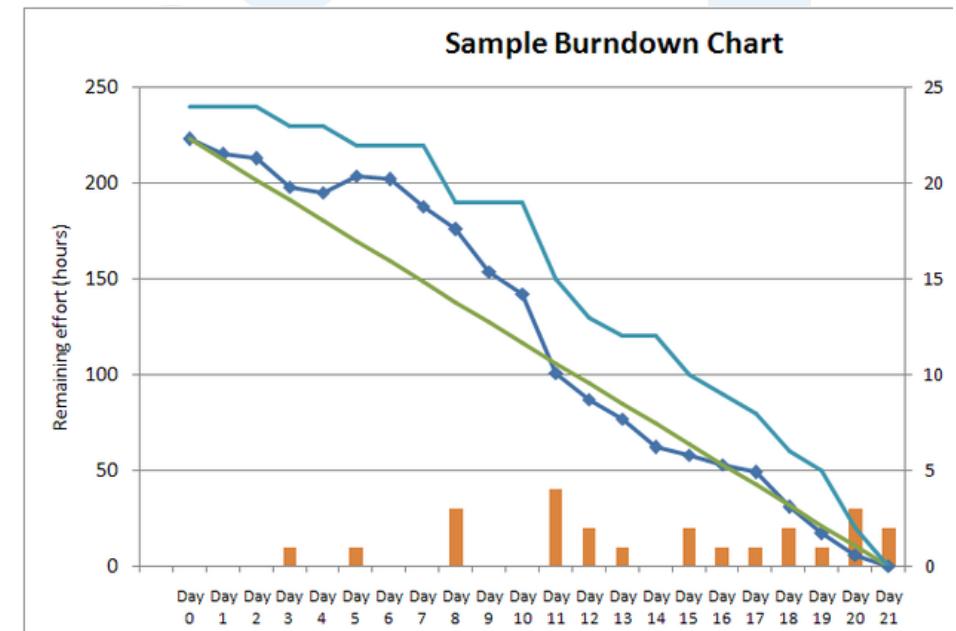
# Sprint burndown chart

A display of what work has been completed and what is left to complete

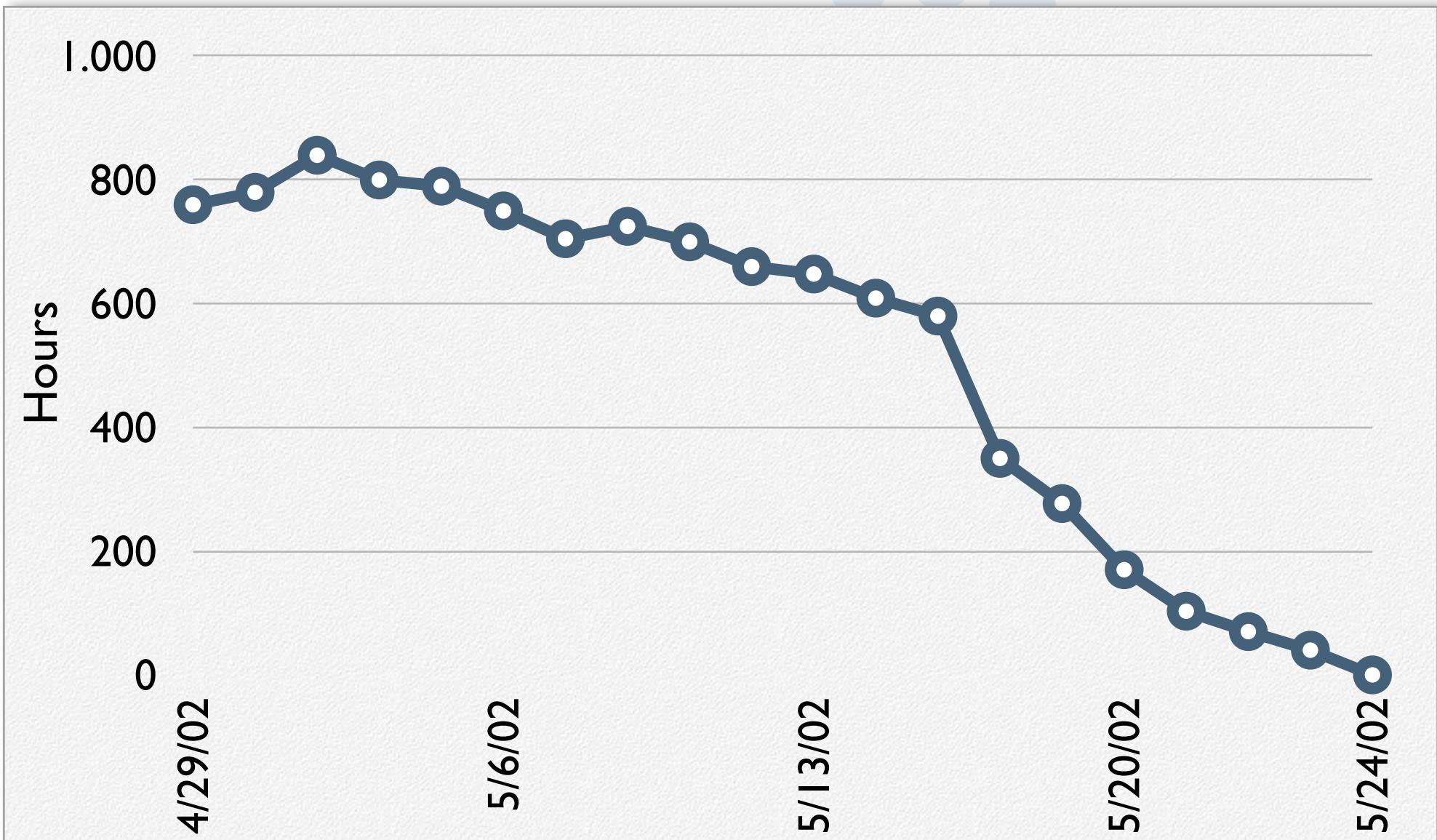
- one for each developer or work item
- updated every day
- (make best guess about hours/points completed each day)

variation: Release burndown chart

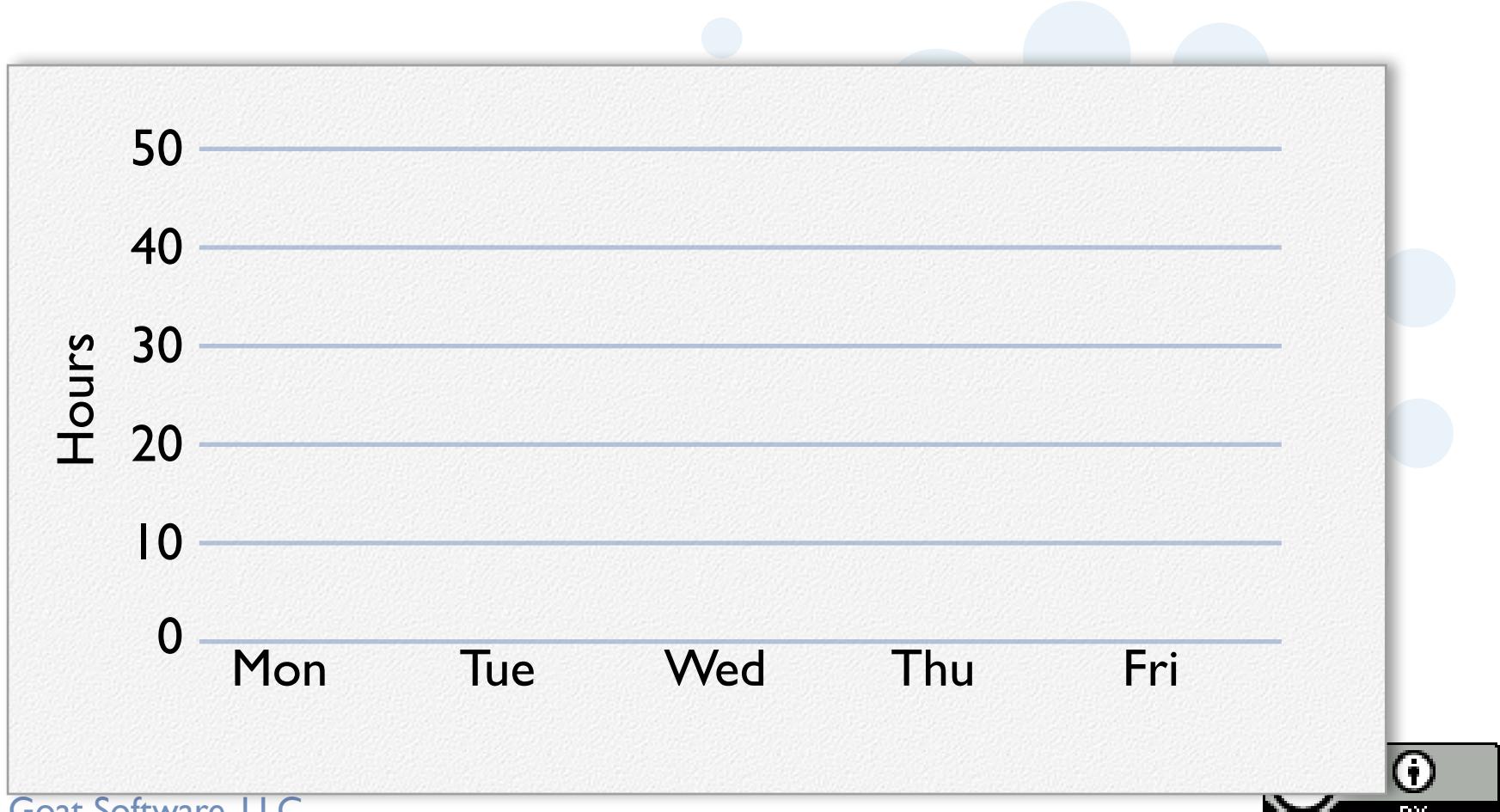
- shows overall progress
- updated at end of each sprint



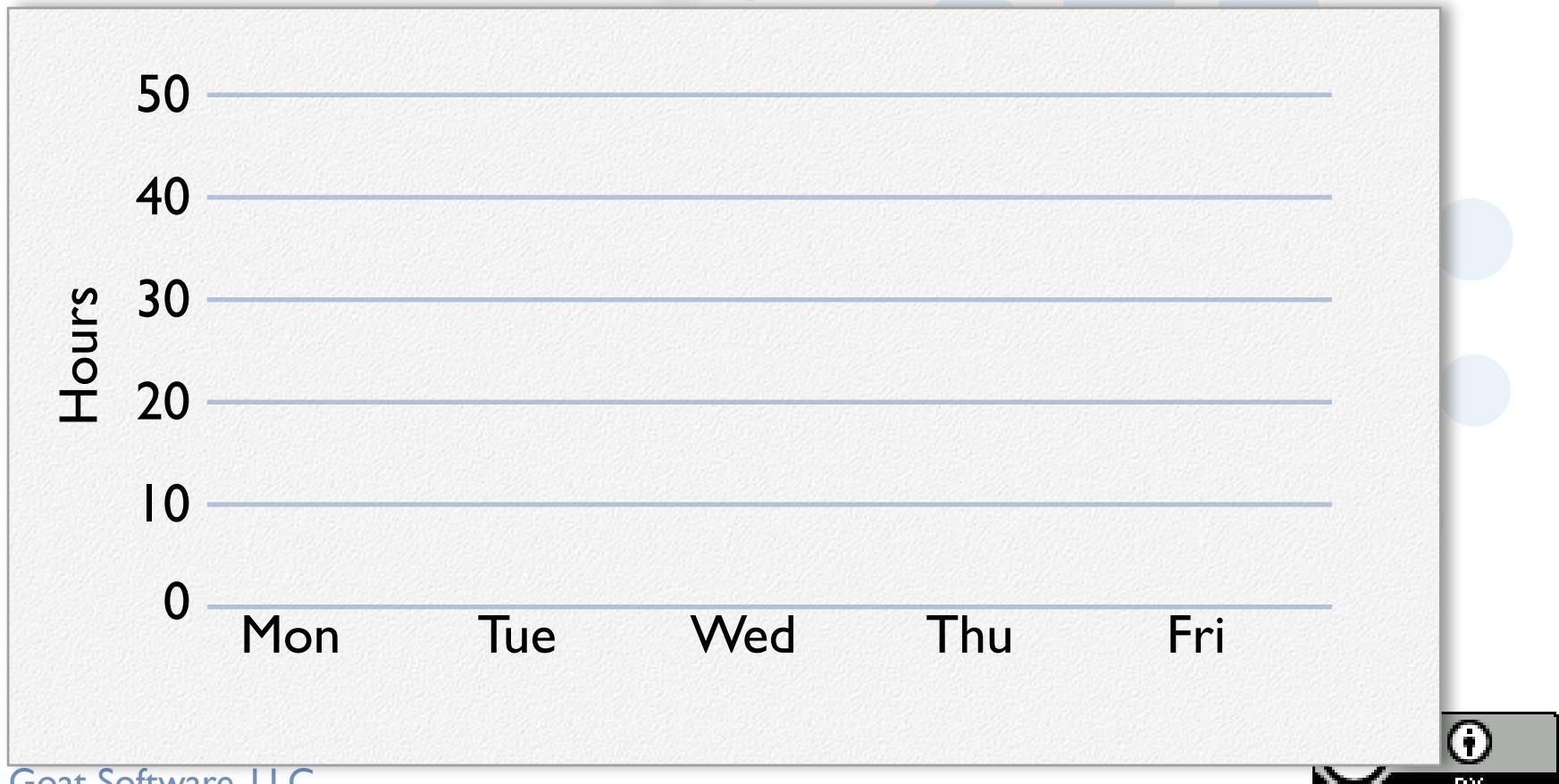
# A sprint burndown chart



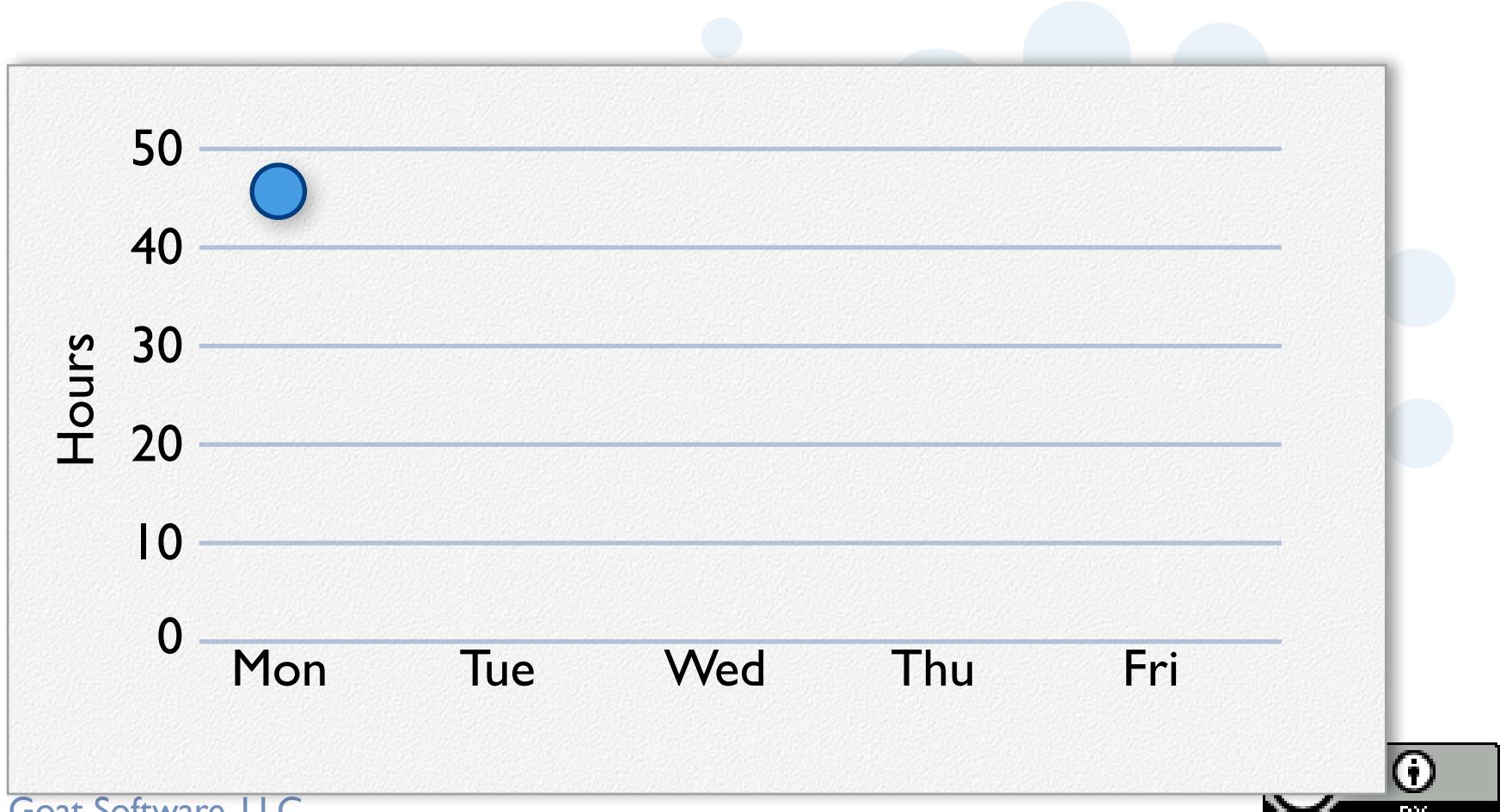
Tasks	Mon	Tues	Wed	Thur	Fri
Code the user interface	8				
Code the middle tier	16				
Test the middle tier	8				
Write online help	12				



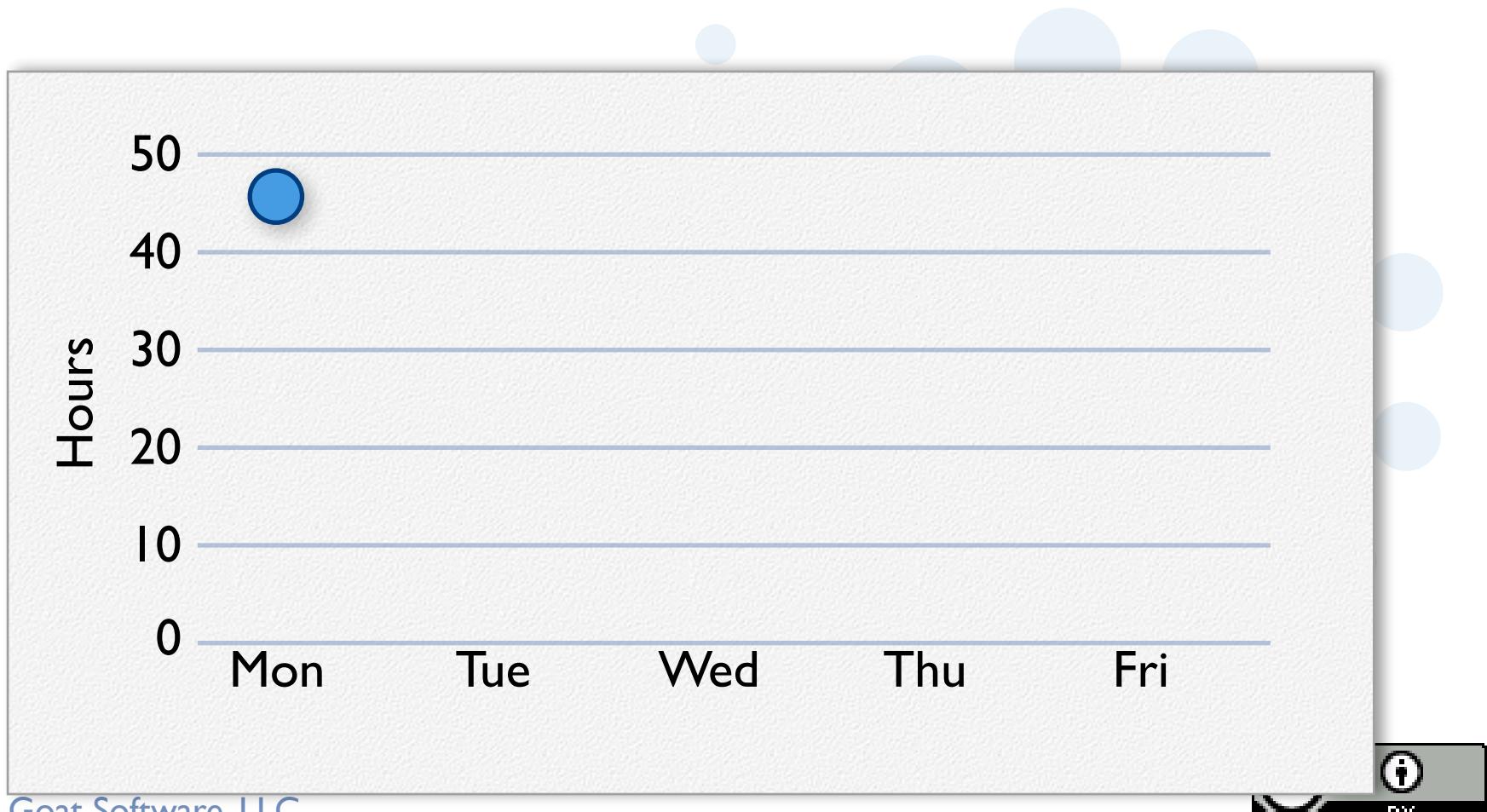
Tasks	Mon	Tues	Wed	Thur	Fri
Code the user interface	8				
Code the middle tier	16				
Test the middle tier	8				
Write online help	12				



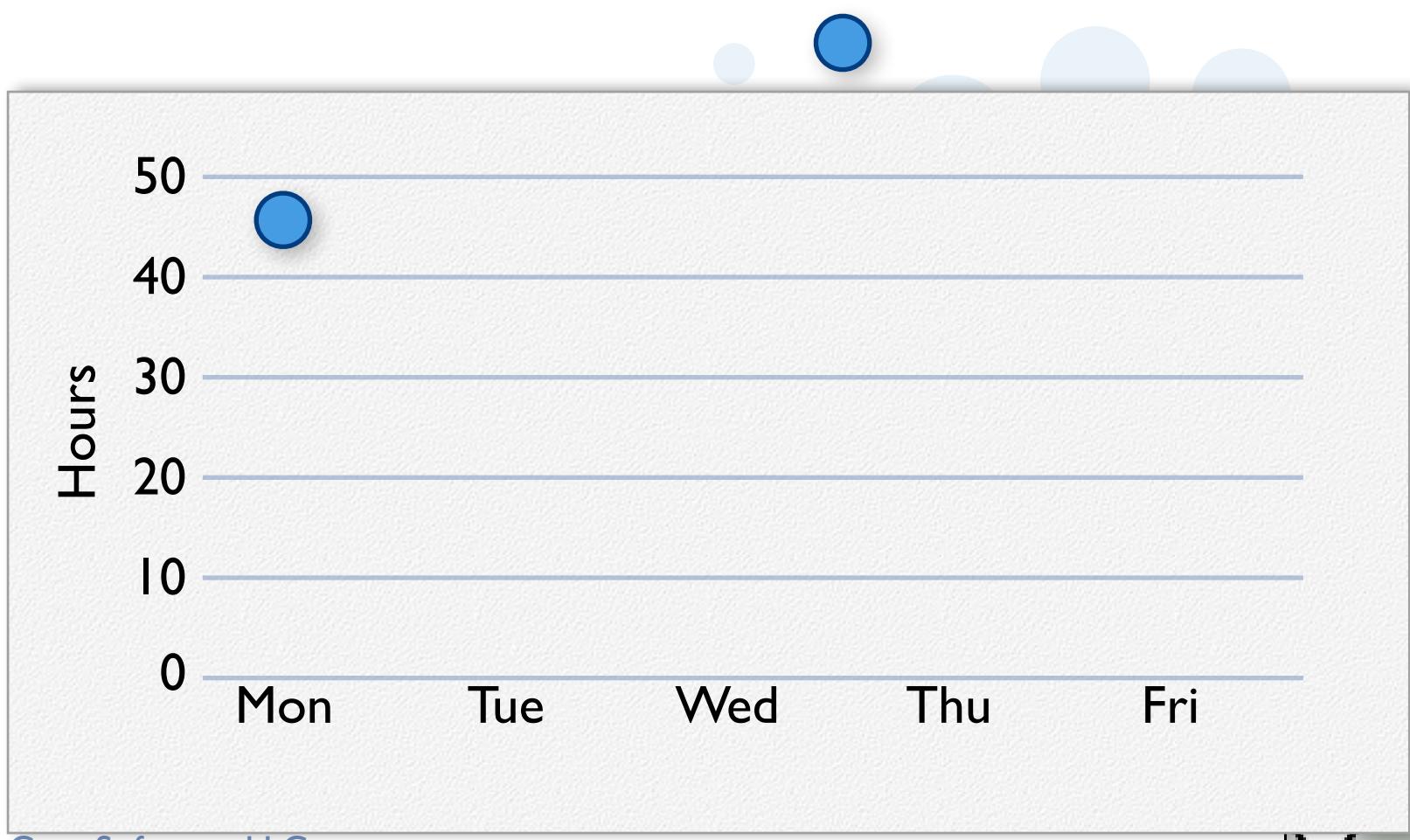
Tasks	Mon	Tues	Wed	Thur	Fri
Code the user interface	8				
Code the middle tier	16				
Test the middle tier	8				
Write online help	12				



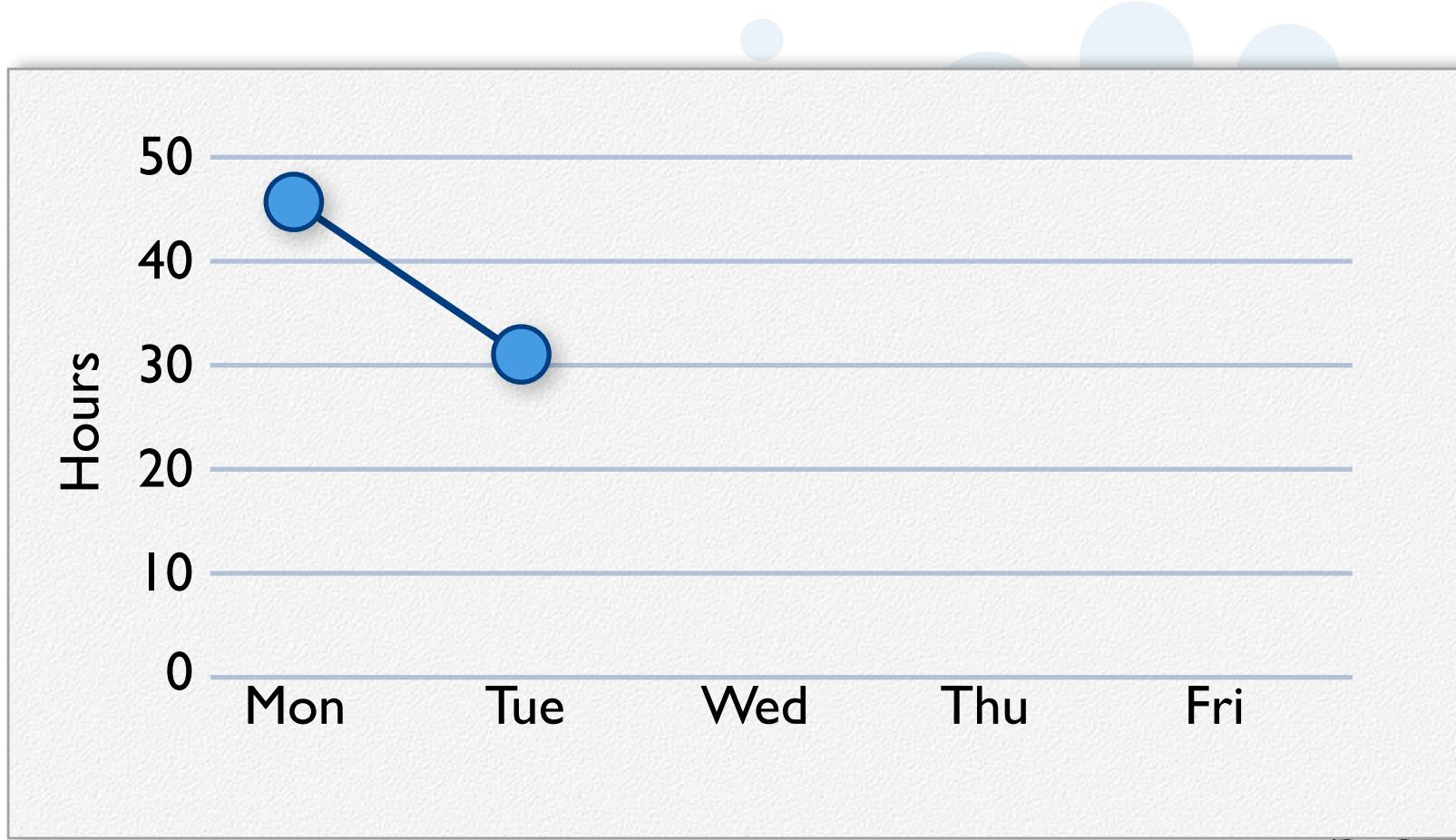
Tasks	Mon	Tues	Wed	Thur	Fri
Code the user interface	8	4			
Code the middle tier	16	12			
Test the middle tier	8	16			
Write online help	12				



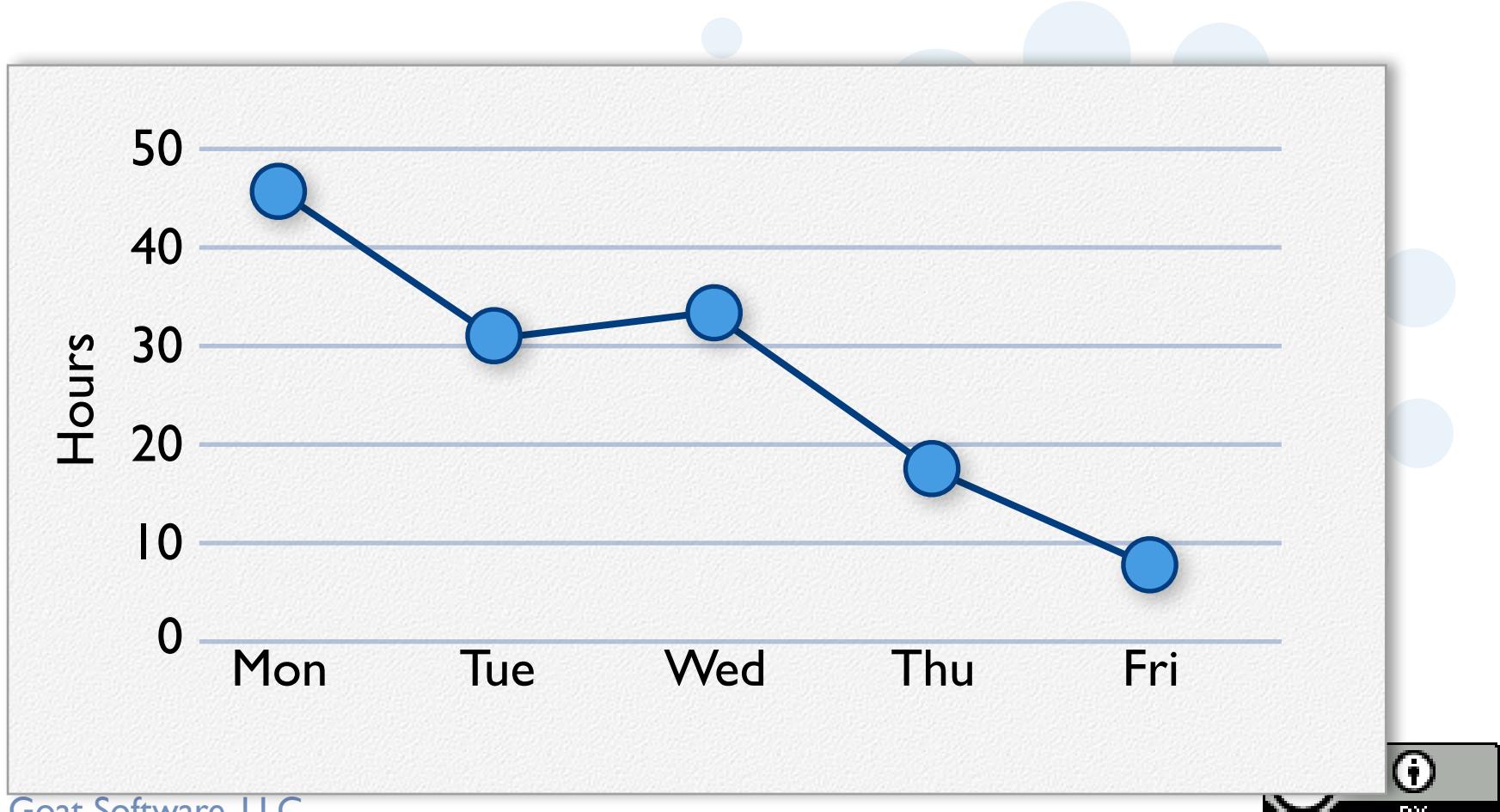
Tasks	Mon	Tues	Wed	Thur	Fri
Code the user interface	8	4			
Code the middle tier	16	12			
Test the middle tier	8	16			
Write online help	12				



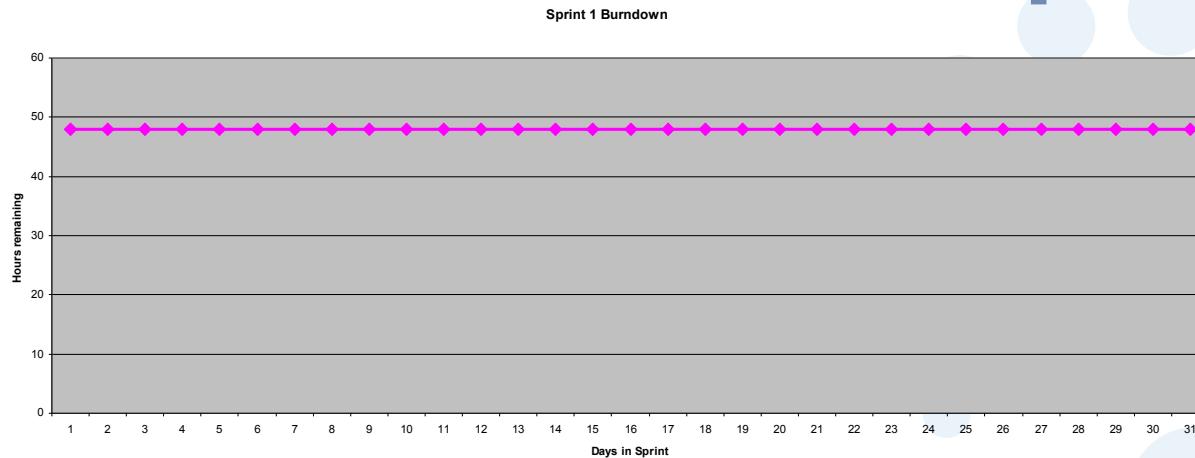
Tasks	Mon	Tues	Wed	Thur	Fri
Code the user interface	8	4			
Code the middle tier	16	12			
Test the middle tier	8	16			
Write online help	12				



Tasks	Mon	Tues	Wed	Thur	Fri
Code the user interface	8	4	8		
Code the middle tier	16	12	10	7	
Test the middle tier	8	16	16	11	8
Write online help	12				



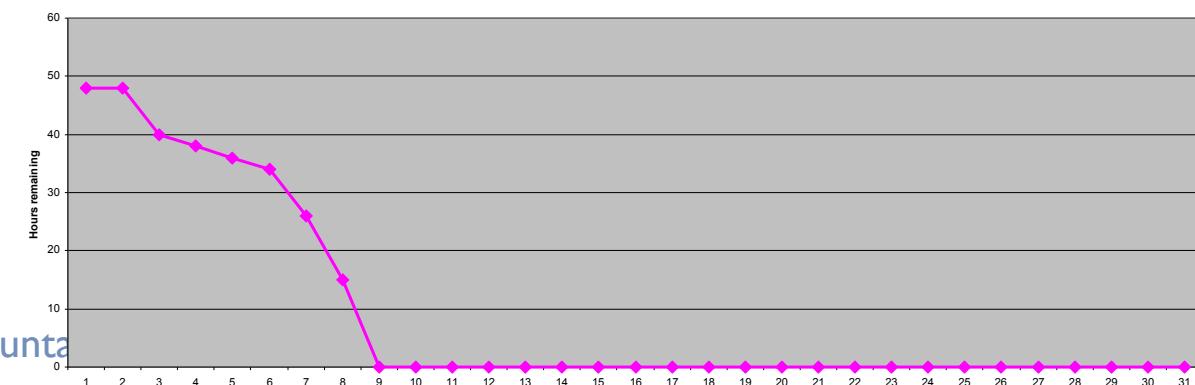
# Burndown examples



No work



Not  
fast enough



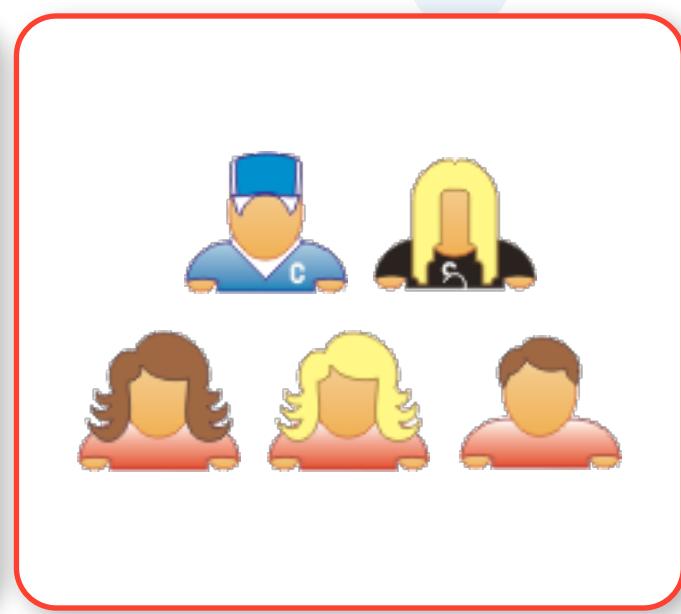
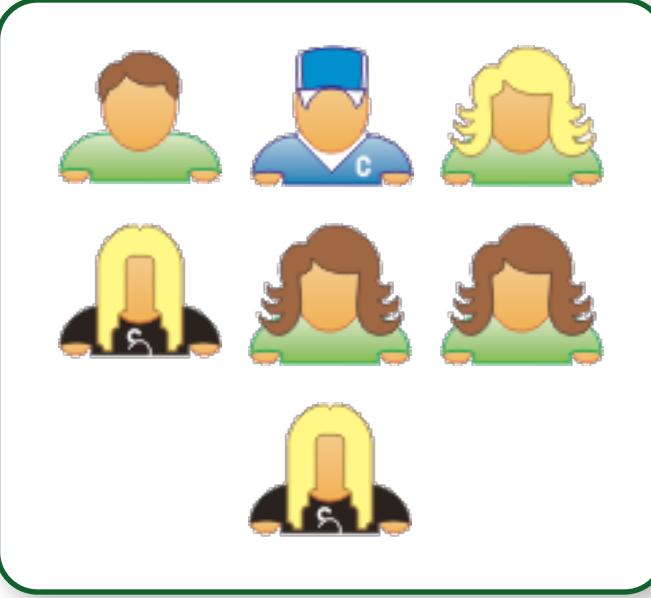
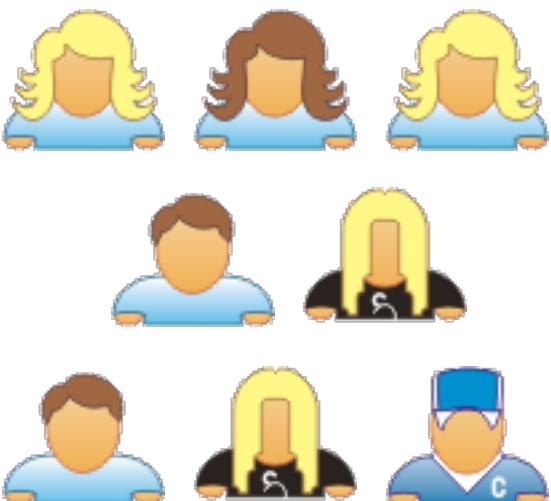
Too fast

# Scalability

- Typical individual team is  $7 \pm 2$  people
  - Scalability comes from teams of teams
- Factors in scaling
  - Type of application
  - Team size
  - Team dispersion
  - Project duration
- Scrum has been used on multiple 500+ person projects



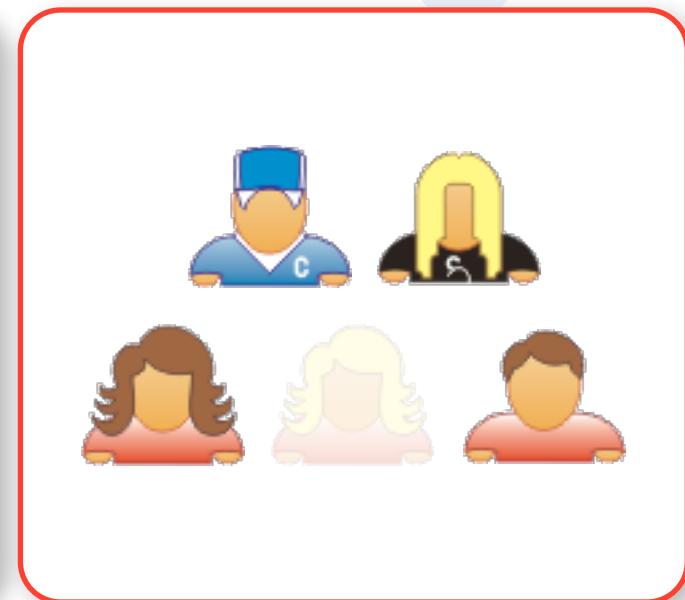
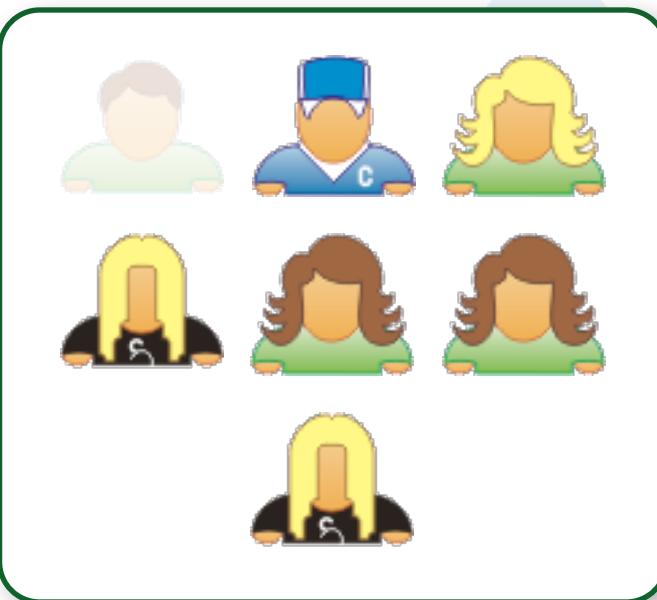
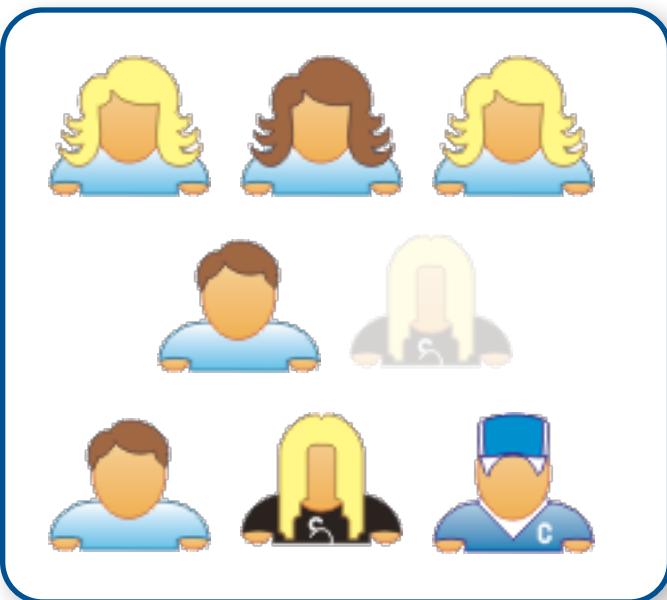
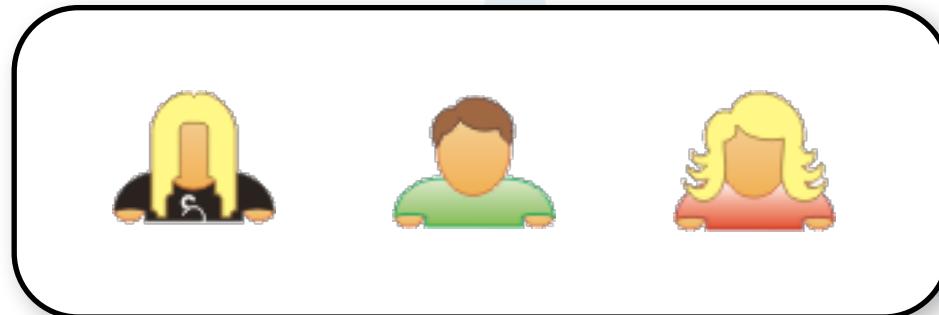
# Scaling through the Scrum of scrums



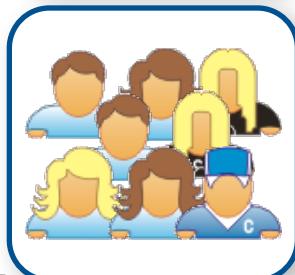
Mountain Goat Software, LLC



# Scaling through the Scrum of scrums



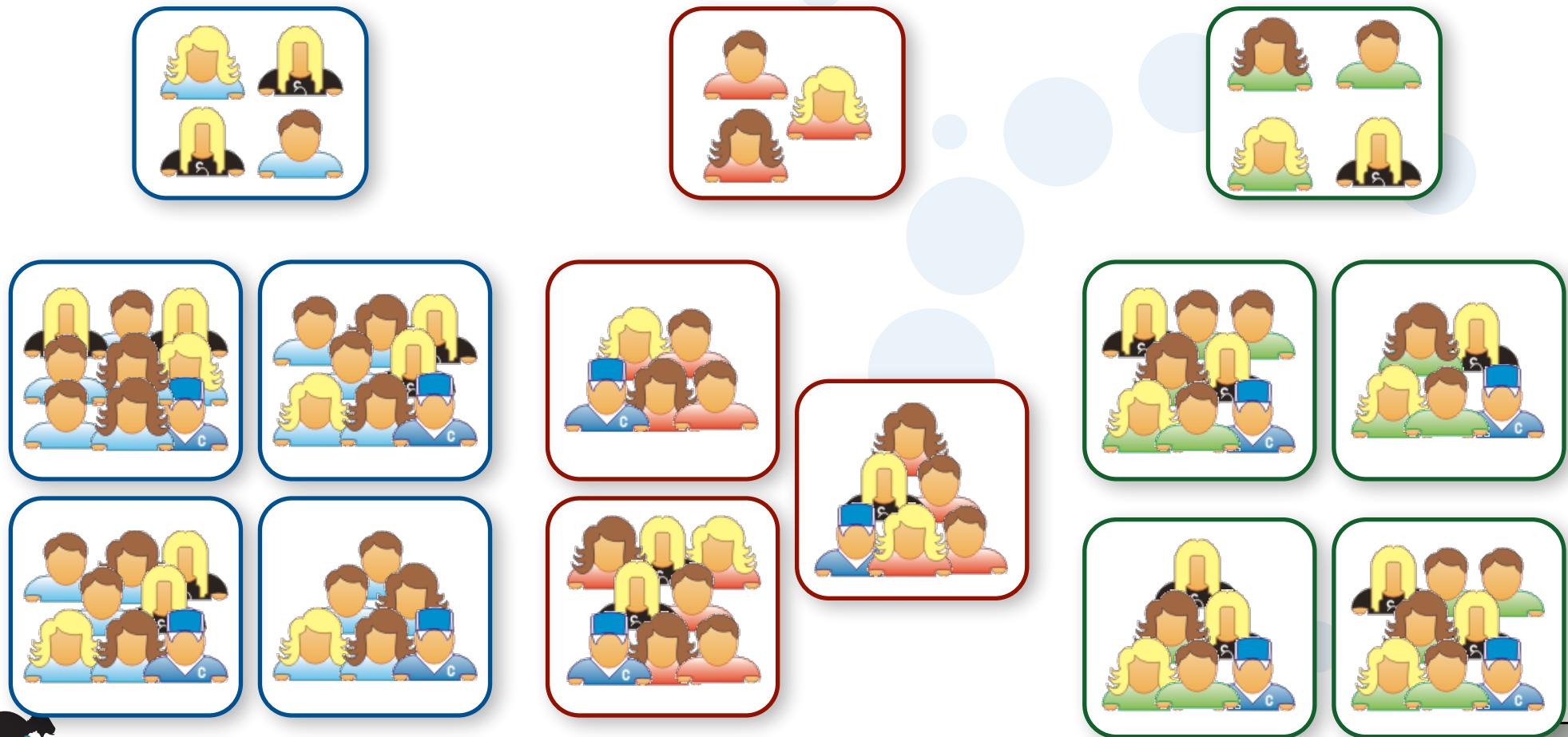
# Scrum of scrums of scrums



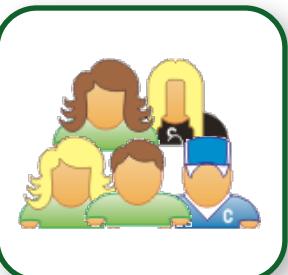
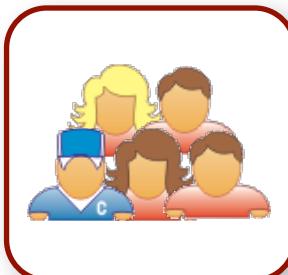
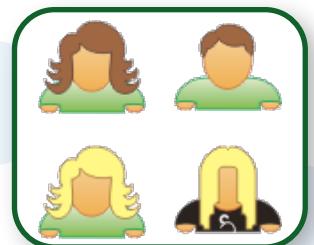
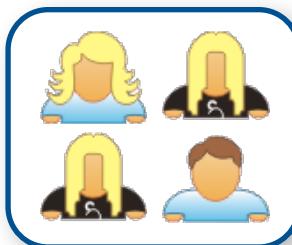
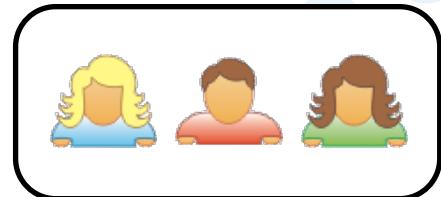
Mountain Goat Software, LLC



# Scrum of scrums of scrums



# Scrum of scrums of scrums



# Where to go next

- [www.mountaingoatsoftware.com/scrum](http://www.mountaingoatsoftware.com/scrum)
- [www.scrumalliance.org](http://www.scrumalliance.org)
- [www.controlchaos.com](http://www.controlchaos.com)
- [scrumdevelopment@yahoogroups.com](mailto:scrumdevelopment@yahoogroups.com)



# A Scrum reading list

- *Agile and Iterative Development: A Manager's Guide* by Craig Larman
- *Agile Estimating and Planning* by Mike Cohn
- *Agile Project Management with Scrum* by Ken Schwaber
- *Agile Retrospectives* by Esther Derby and Diana Larsen
- *Agile Software Development Ecosystems* by Jim Highsmith
- *Agile Software Development with Scrum* by Ken Schwaber and Mike Beedle
- *Scrum and The Enterprise* by Ken Schwaber
- *User Stories Applied for Agile Software Development* by Mike Cohn
- Lots of weekly articles at [www.scrumalliance.org](http://www.scrumalliance.org)



# Copyright notice



You are free:

- to Share—to copy, distribute and transmit the work
- to Remix—to adapt the work

Under the following conditions

- Attribution. You must attribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work).

Nothing in this license impairs or restricts the author's moral rights.

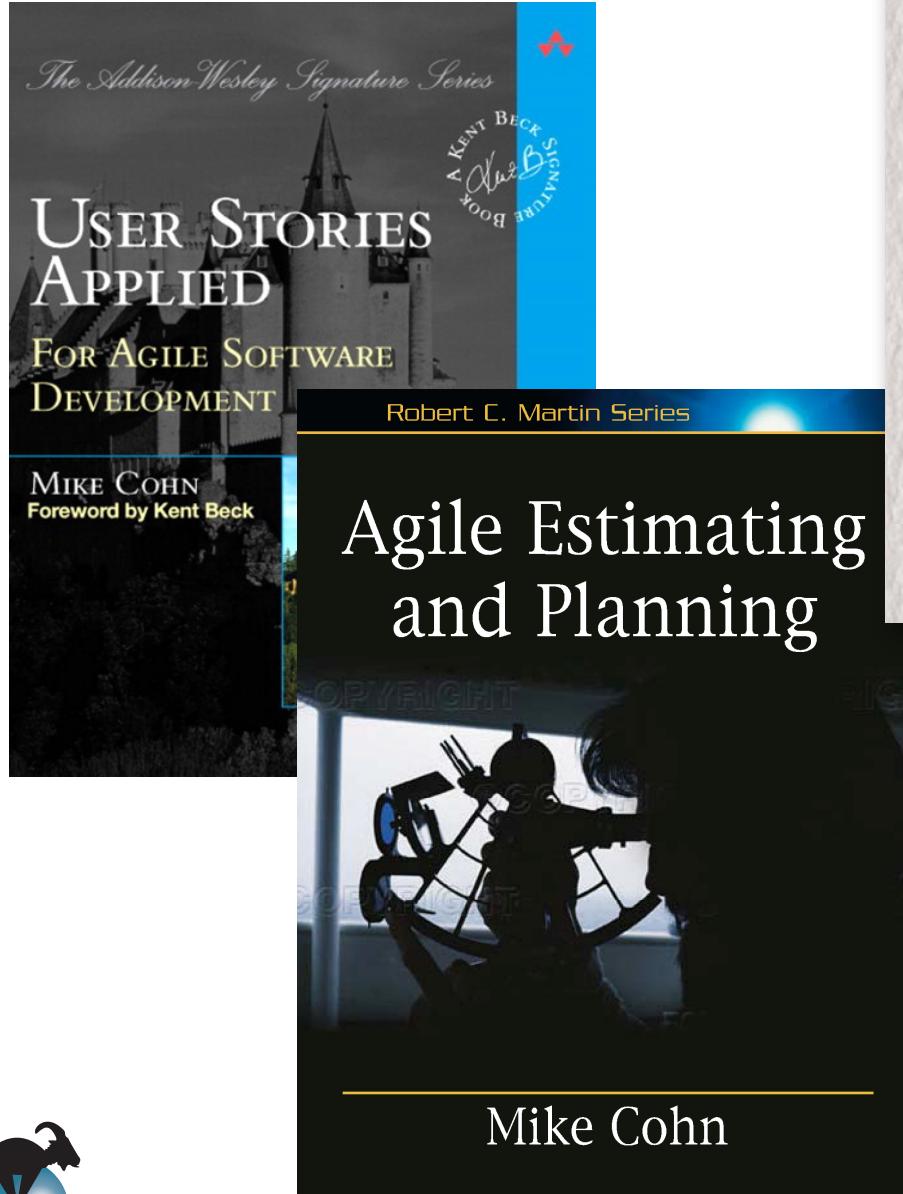
For more information see <http://creativecommons.org/licenses/by/3.0/>



Mountain Goat Software, LLC



# Contact information



Presentation by: Mike Cohn  
[mike@mountaingoatsoftware.com](mailto:mike@mountaingoatsoftware.com)  
[www.mountaingoatsoftware.com](http://www.mountaingoatsoftware.com)  
**(720) 890-6110 (office)**

You can remove this  
(or any slides) but  
please credit the  
source somewhere  
in your presentation

