CSE 403
Lecture 24

Scrum and Agile Software Development

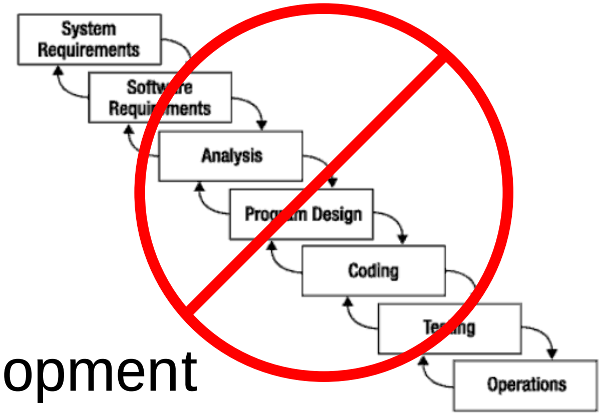
Reading:

Scrum Primer,
by Deemer/Benefield/Larman/Vodde

slides created by Marty Stepp
http://www.cs.washington.edu/403/

What is Scrum?

- **Scrum**: **It’s about common sense**
  - Is an agile, lightweight process
  - Can manage and control software and product development
  - Uses iterative, incremental practices
  - Has a simple implementation
  - Increases productivity
  - Reduces time to benefits
  - Embraces adaptive, empirical systems development
  - Is not restricted to software development projects
  - Embraces the opposite of the waterfall approach...



Scrum Origins

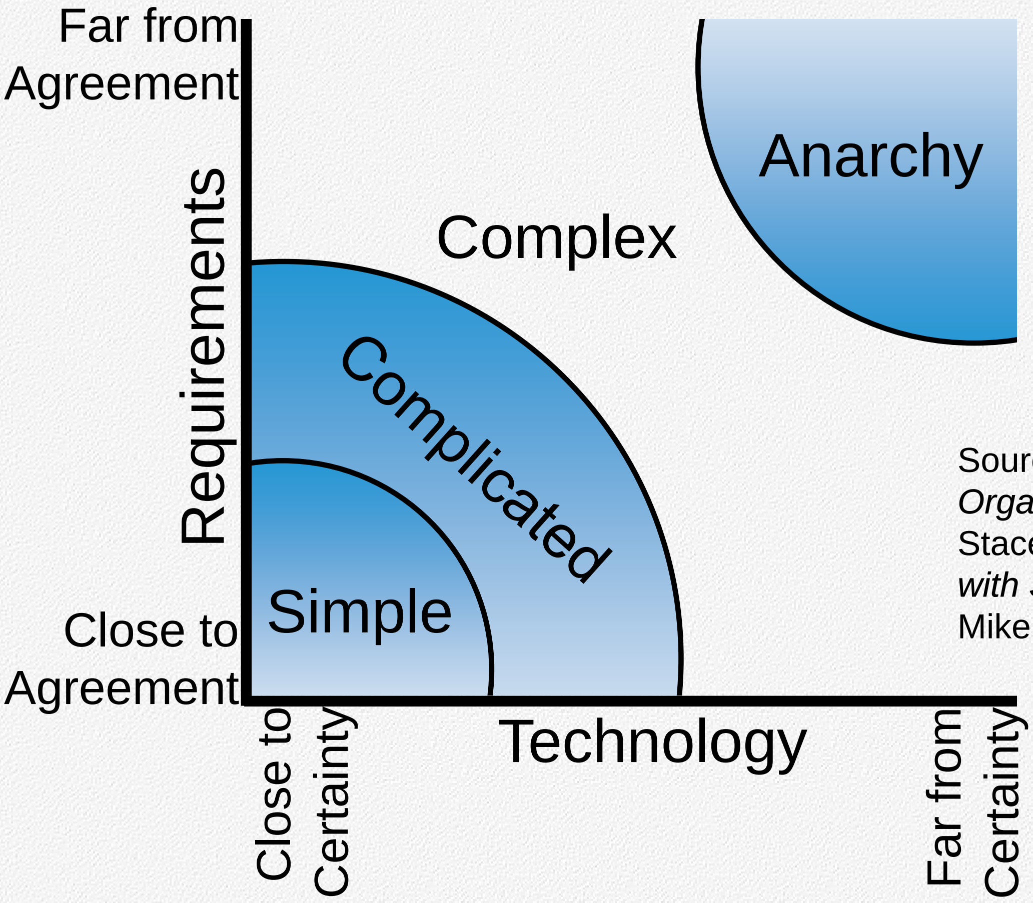
- Jeff Sutherland
  - Initial scrums at Easel Corp in 1993
  - IDX and 500+ people doing Scrum
- Ken Schwaber
  - ADM
  - Scrum presented at OOPSLA 96 with Sutherland
  - Author of three books on Scrum
- Mike Beedle
  - Scrum patterns in PLOPD4
- Ken Schwaber and Mike Cohn
  - Co-founded Scrum Alliance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

Agile Manifesto

Individuals and interactions	over	Process and tools
Working software	over	Comprehensive documentation
Customer collaboration	over	Contract negotiation
Responding to change	over	Following a plan

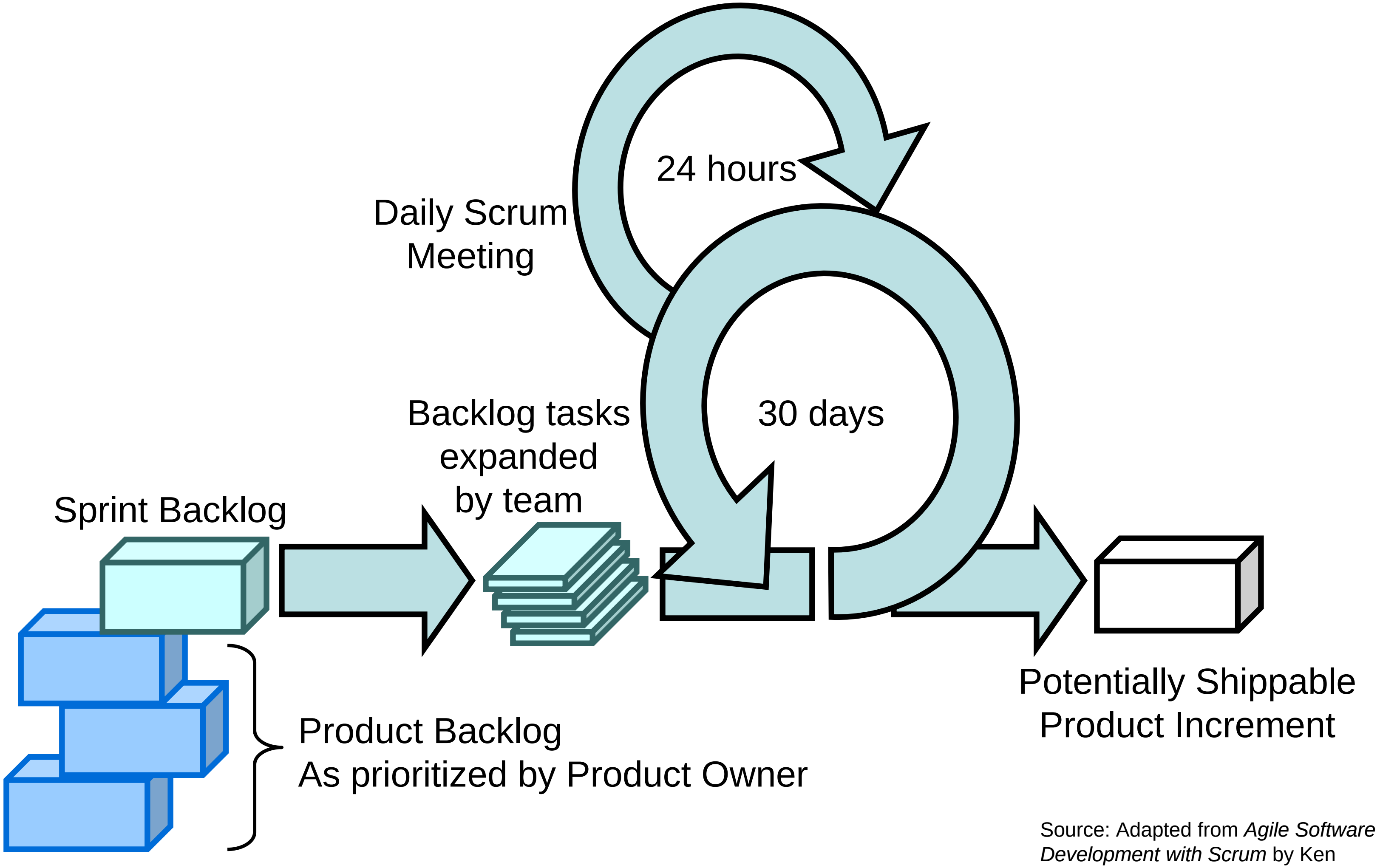
Source: www.agilemanifesto.org

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 at a Glance


24 hours
Daily Scrum Meeting
30 days
Backlog tasks expanded by team
Sprint Backlog
Product Backlog
As prioritized by Product Owner
Potentially Shippable Product Increment

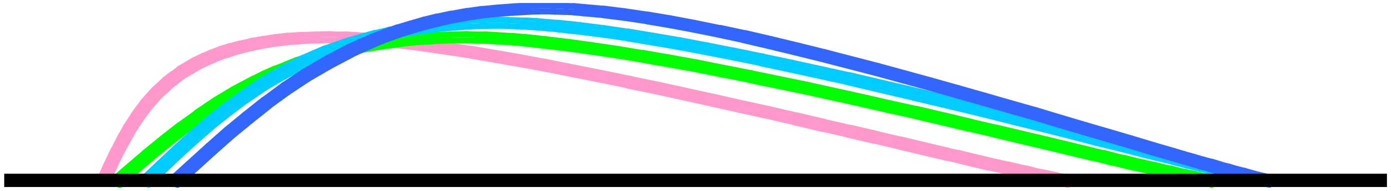
Source: Adapted from *Agile Software Development with Scrum* by Ken Schwaber and Mike Beedle.

Sequential vs. Overlap

Requirements | Design | Code | Test

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

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



Scrum Framework

Roles

- Product owner
- Scrum Master
- Team

Ceremonies

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

Artifacts

- Product backlog
- Sprint backlog
- Burndown charts

Scrum Roles

- Product Owner
  - Possibly a Product Manager or Project Sponsor
  - Decides features, release date, prioritization, $$$

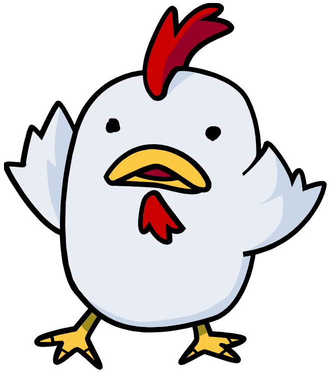
- Scrum Master
  - Typically a Project Manager or Team Leader
  - Responsible for enacting Scrum values and practices
  - Remove impediments / politics, keeps everyone productive

- Project Team
  - 5-10 members; Teams are self-organizing
  - Cross-functional: QA, Programmers, UI Designers, etc.
  - Membership should change only between sprints

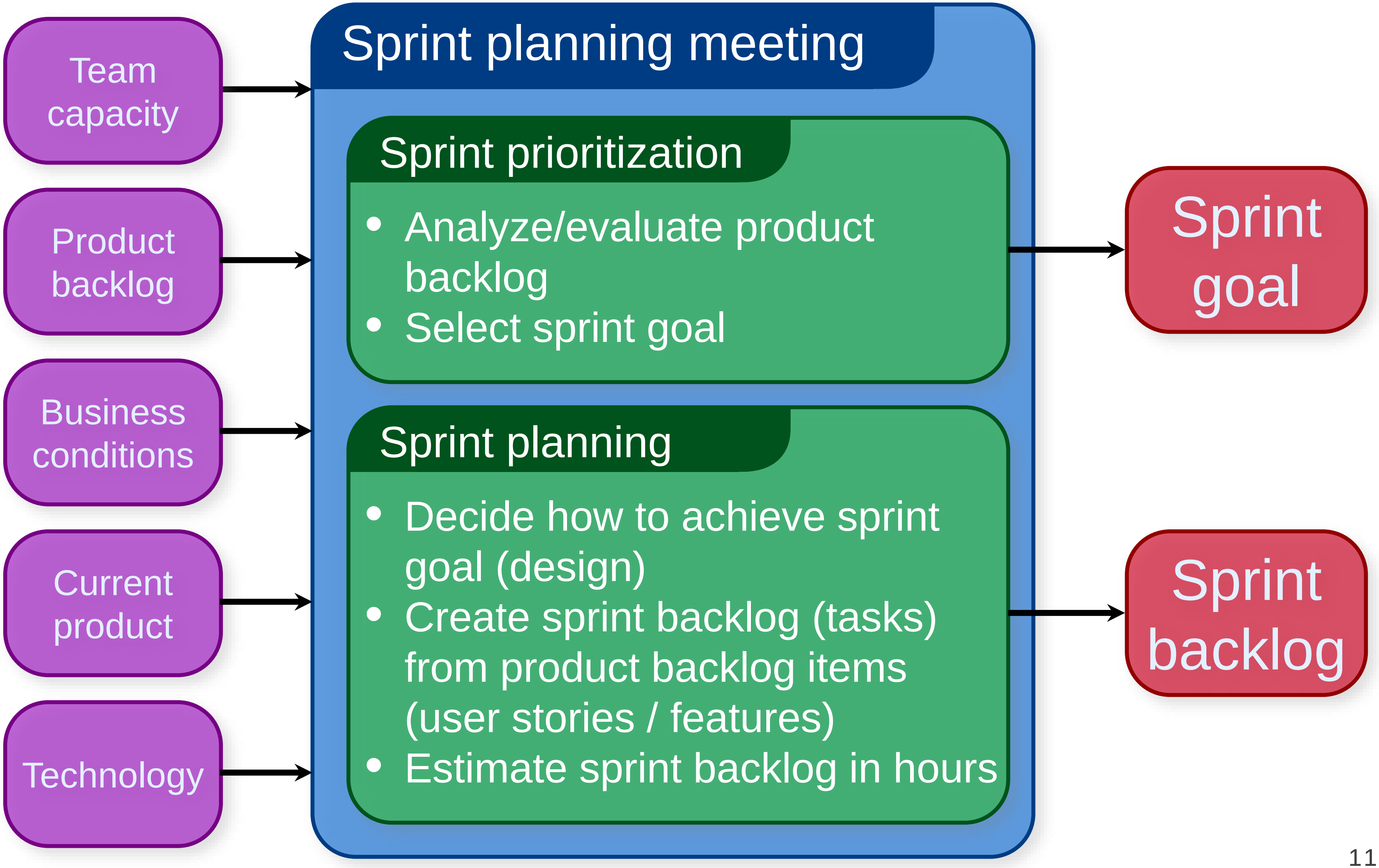
"Pigs" and "Chickens"

- **Pig**: Team member committed to success of project
- **Chicken**: Not a pig; interested but not committed

A pig and a chicken are walking down a road. The chicken looks at the pig and says, "Hey, why don't we open a restaurant?" The pig looks back at the chicken and says, "Good idea, what do you want to call it?" The chicken thinks about it and says, "Why don't we call it 'Ham and Eggs'?" "I don't think so," says the pig, "I'd be committed but you'd only be involved."



Sprint Planning Mtg.



Team capacity

Product backlog

Business conditions

Current product

Technology

Sprint planning meeting

Sprint prioritization

- Analyze/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 backlog

Daily Scrum Meeting

- Parameters
  - Daily, ~15 minutes, Stand-up
  - Anyone late pays a $1 fee

- Not for problem solving
  - Whole world is invited
  - Only team members, Scrum Master, product owner, can talk
  - Helps avoid other unnecessary meetings

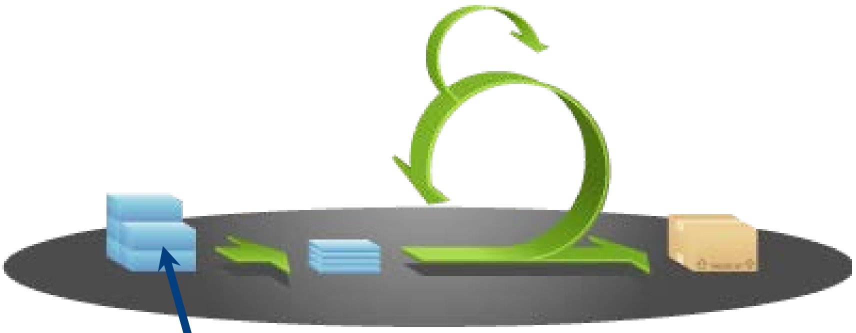
- Three questions answered by each team member:
  1. What did you do yesterday?
  2. What will you do today?
  3. What obstacles are in your way?

Scrum's Artifacts

- Scrum has remarkably few artifacts
  - Product Backlog
  - Sprint Backlog
  - Burndown Charts

- Can be managed using just an Excel spreadsheet
  - More advanced / complicated tools exist:
    - Expensive
    - Web-based – no good for Scrum Master/project manager who travels
    - Still under development

Product Backlog



This is the product backlog

- The requirements
- A list of all desired work on project
- Ideally expressed as a list of user stories along with "story points", such that each item has value to users or customers of the product
- Prioritized by the product owner
- Reprioritized at start of each sprint

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].

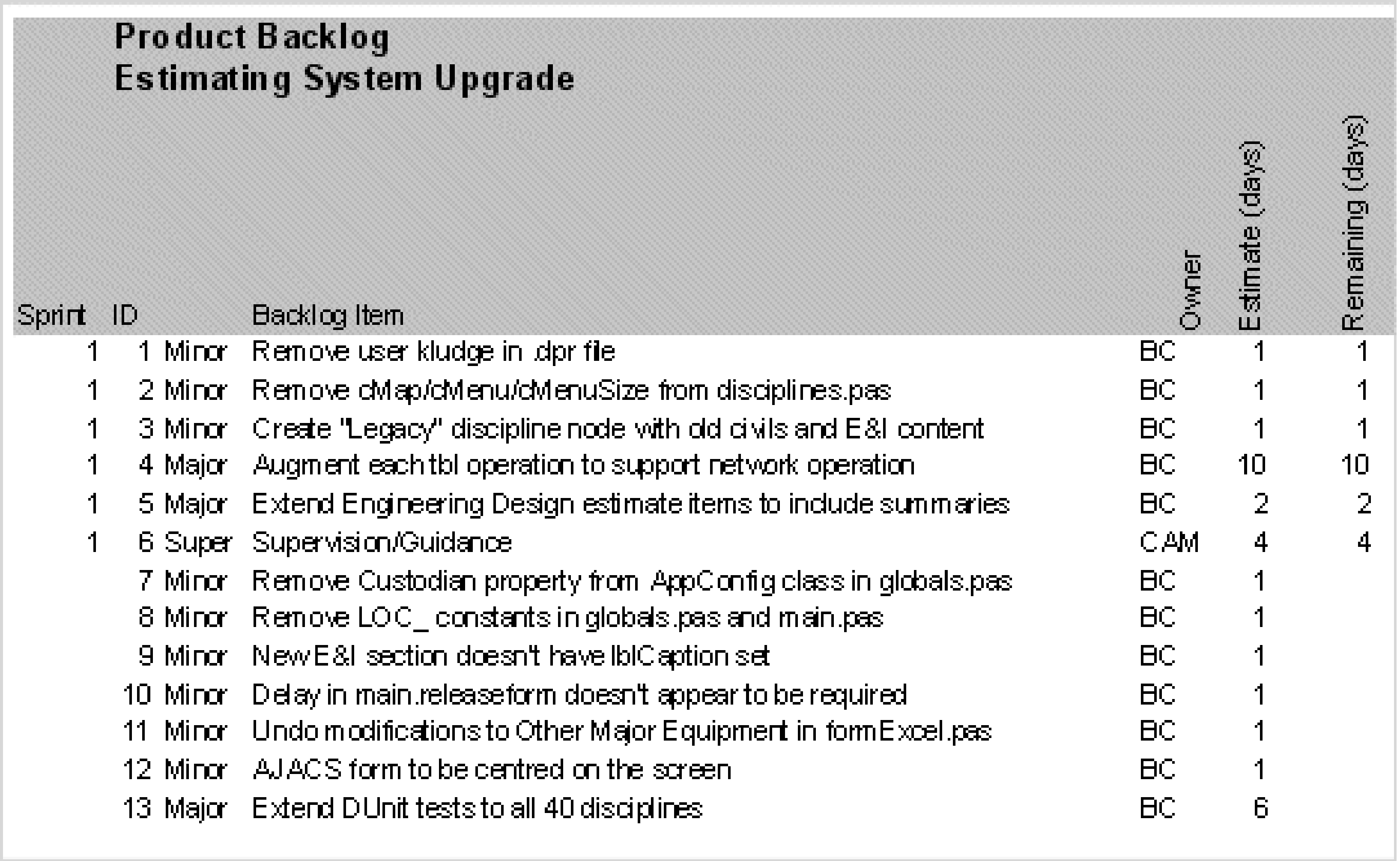
- Example:
  - "As a user, I want to log in, so I can access subscriber content."

- **story points**: Rating of effort needed to implement this story
  - common scales: 1-10, shirt sizes (XS, S, M, L, XL), etc.

Sample Product Backlog

Backlog item	Estimate
Allow a guest to make a reservation	3 (story points)
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

Sample Product Backlog 2



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	

Sprint Backlog

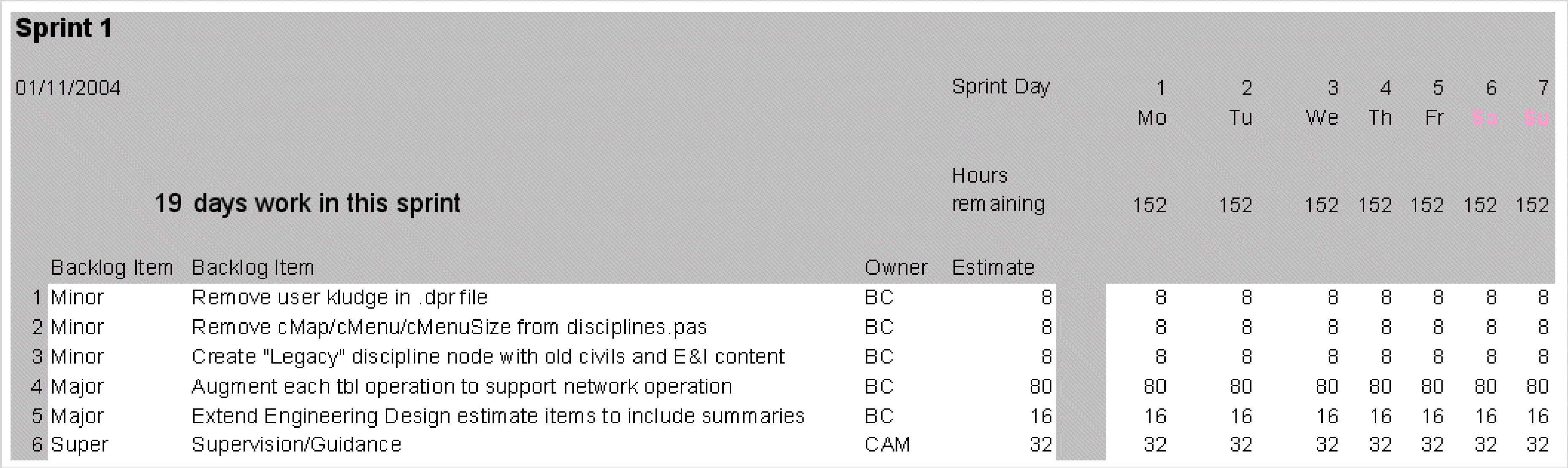
- Individuals sign up for work of their own choosing
  - Work is never assigned
- Estimated work remaining is updated daily

- Any team member can add, delete change 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

Sample Sprint backlog

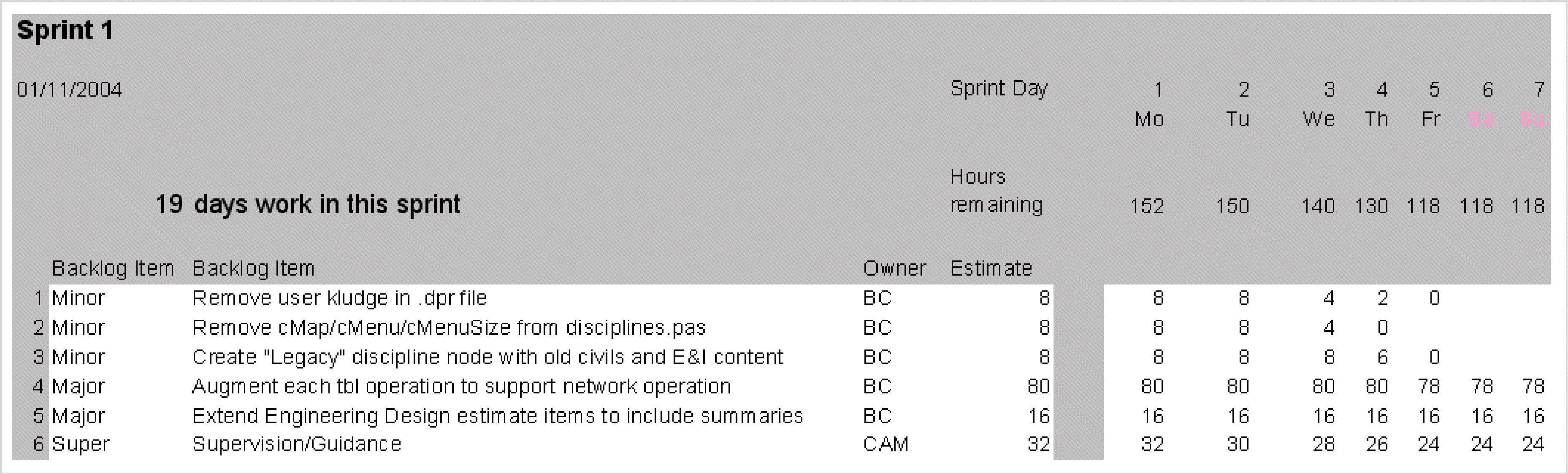
Tasks	Mon	Tue	Wed	Thu	Fri
Code the user interface	8	4	8		
Code the middle tier	16	12	10	4	
Test the middle tier	8	16	16	11	8
Write online help	12				
Write the Foo class	8	8	8	8	8
Add error logging			8	4	

Sample 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	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



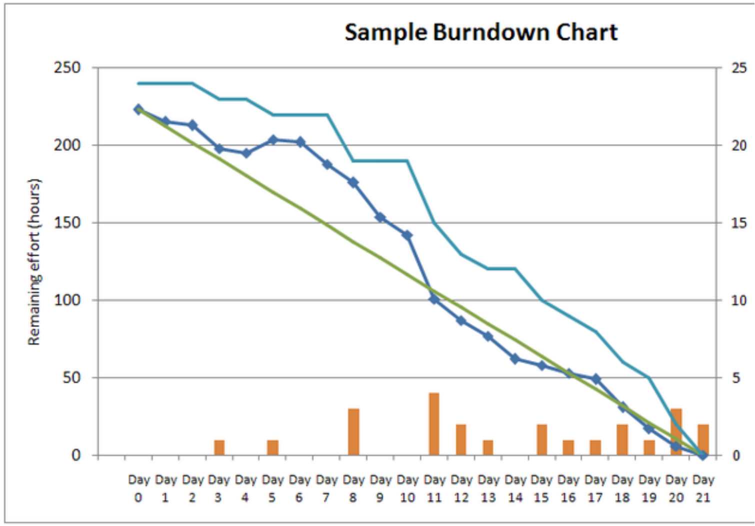
Sprint 1

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	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	8	6	0		
4	Major	Augment each tbl operation to support network operation	BC	80		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	16
6	Super	Supervision/Guidance	CAM	32		32	30	28	26	24	24	24

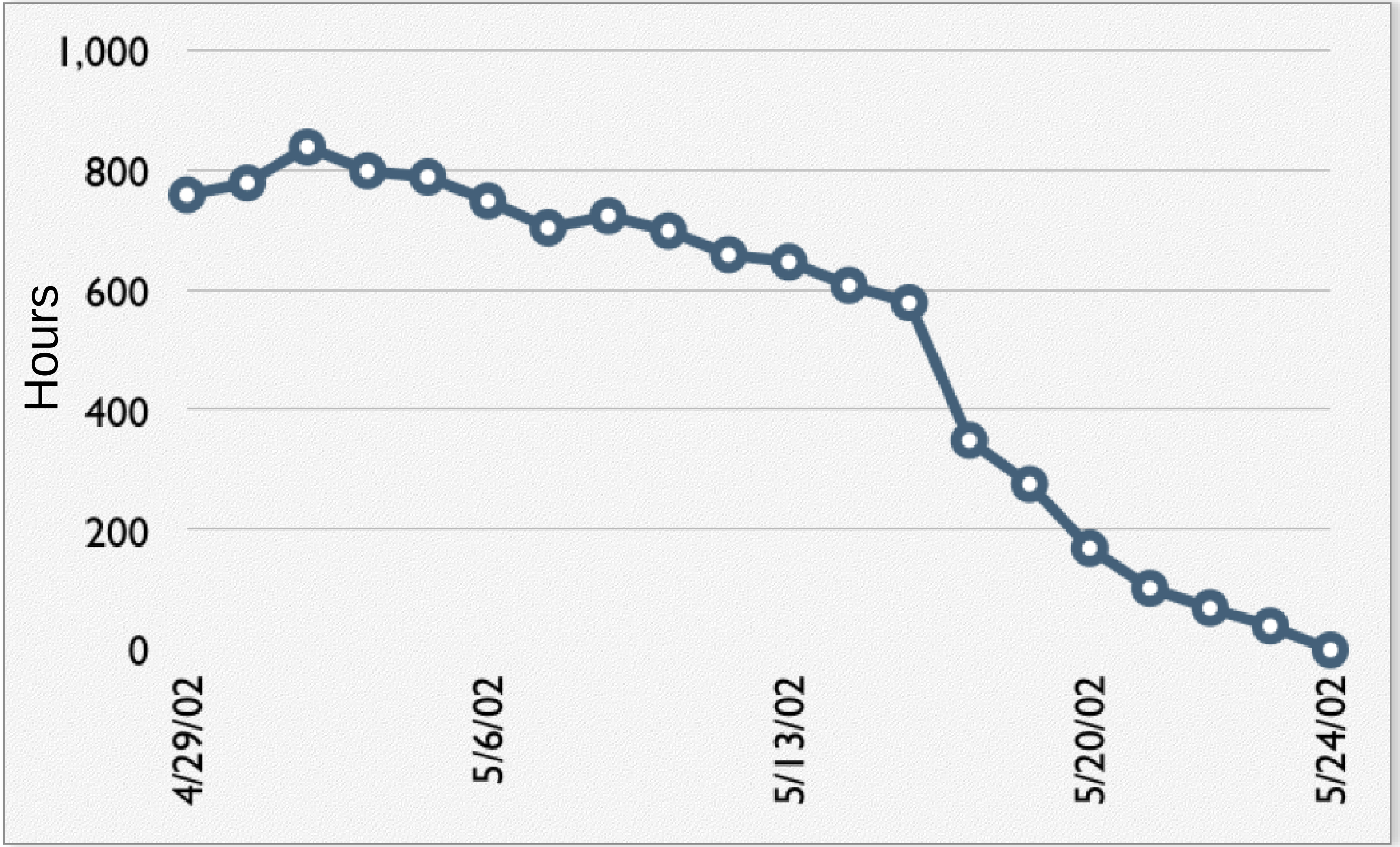
Sprint Burndown Chart

- 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)

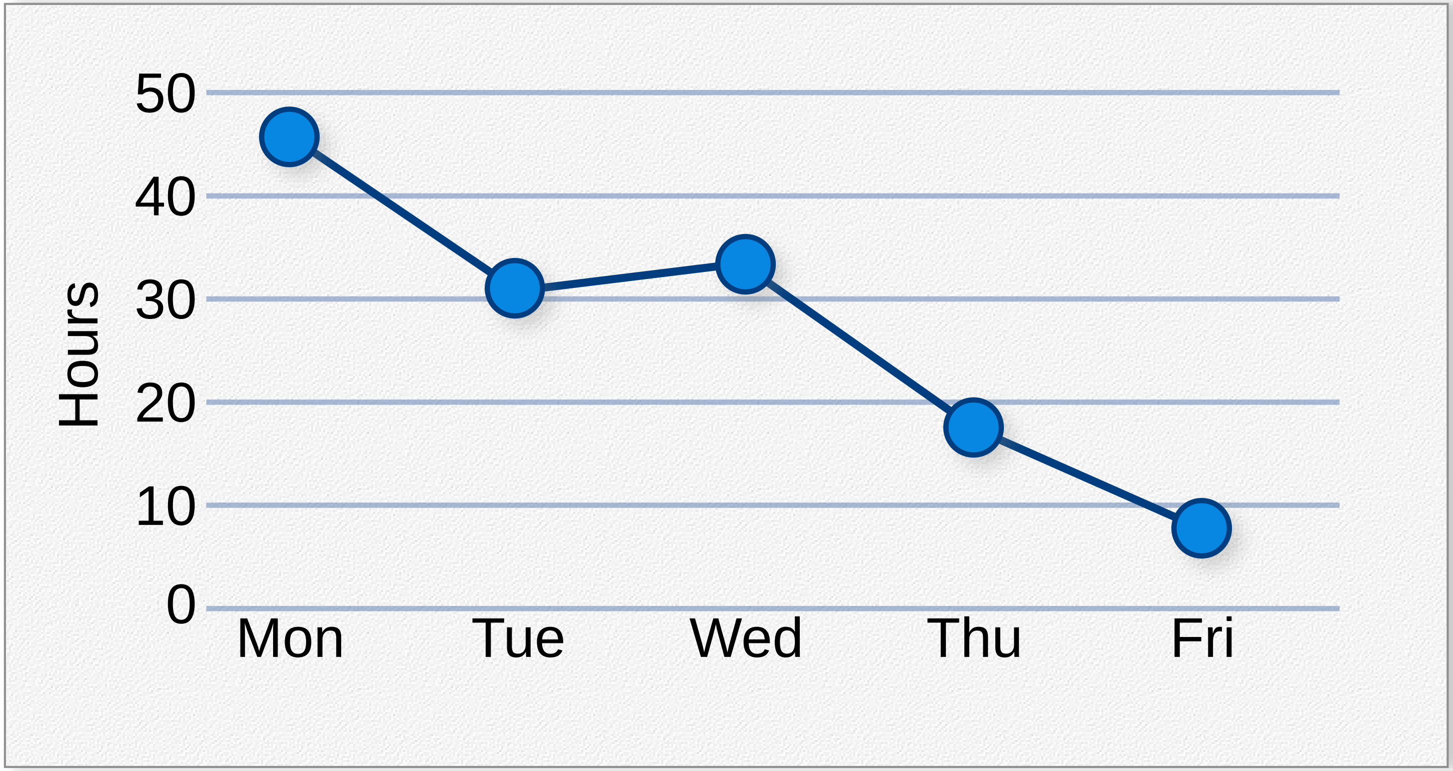
- *variation:* Release burndown chart
  - shows overall progress
  - updated at end of each sprint



Sample Burndown Chart

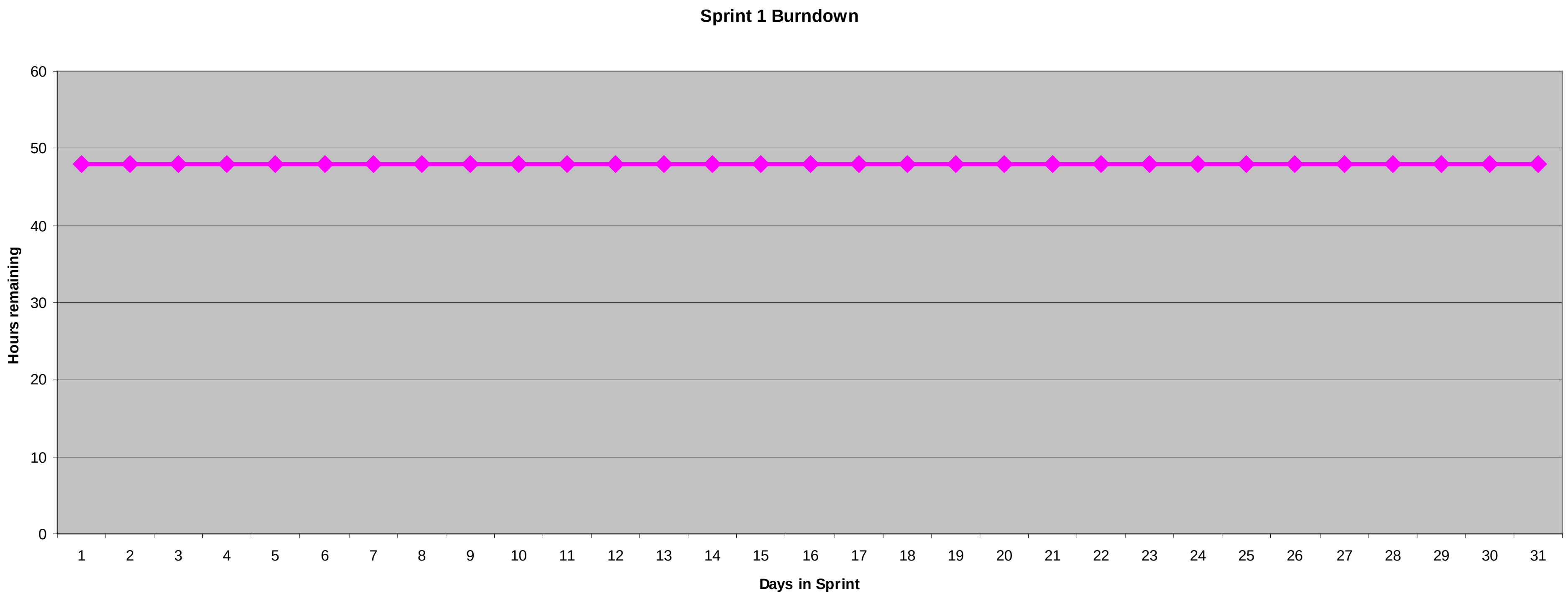


Tasks	Mon	Tue	Wed	Thu	Fri
Code the user interface	8	4	8		
Code the middle tier	16	12	10	7	
Test the middle tier	8	16	16	11	8
Write online help	12				



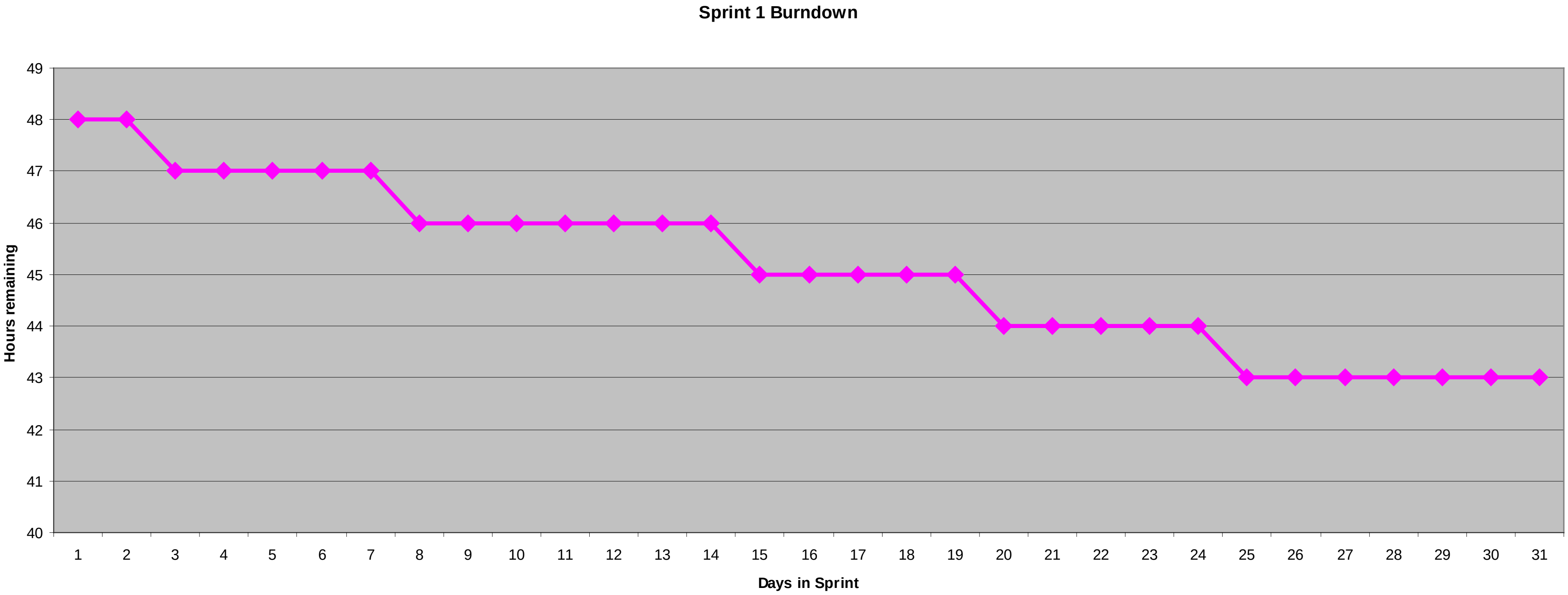
Burndown Example 1

No work being performed



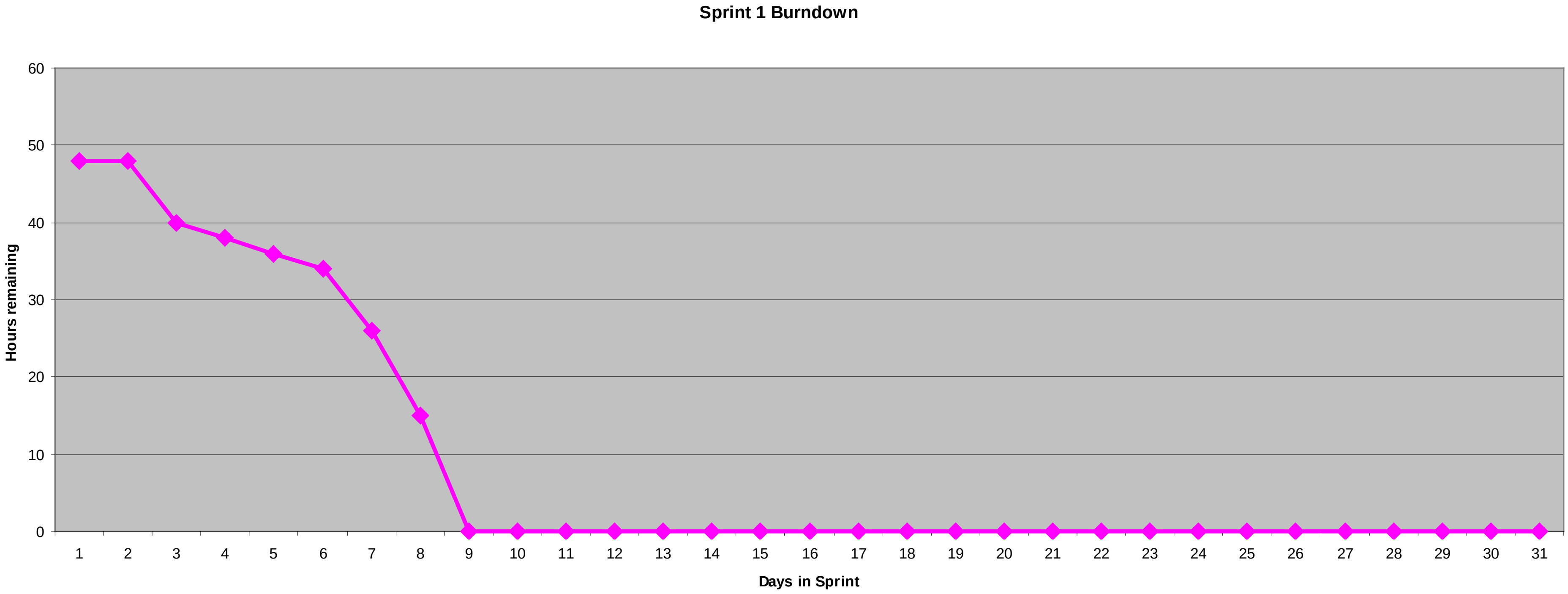
Burndown Example 2

Work being performed, but not fast enough



Burndown Example 3

Work being performed, but too fast!



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

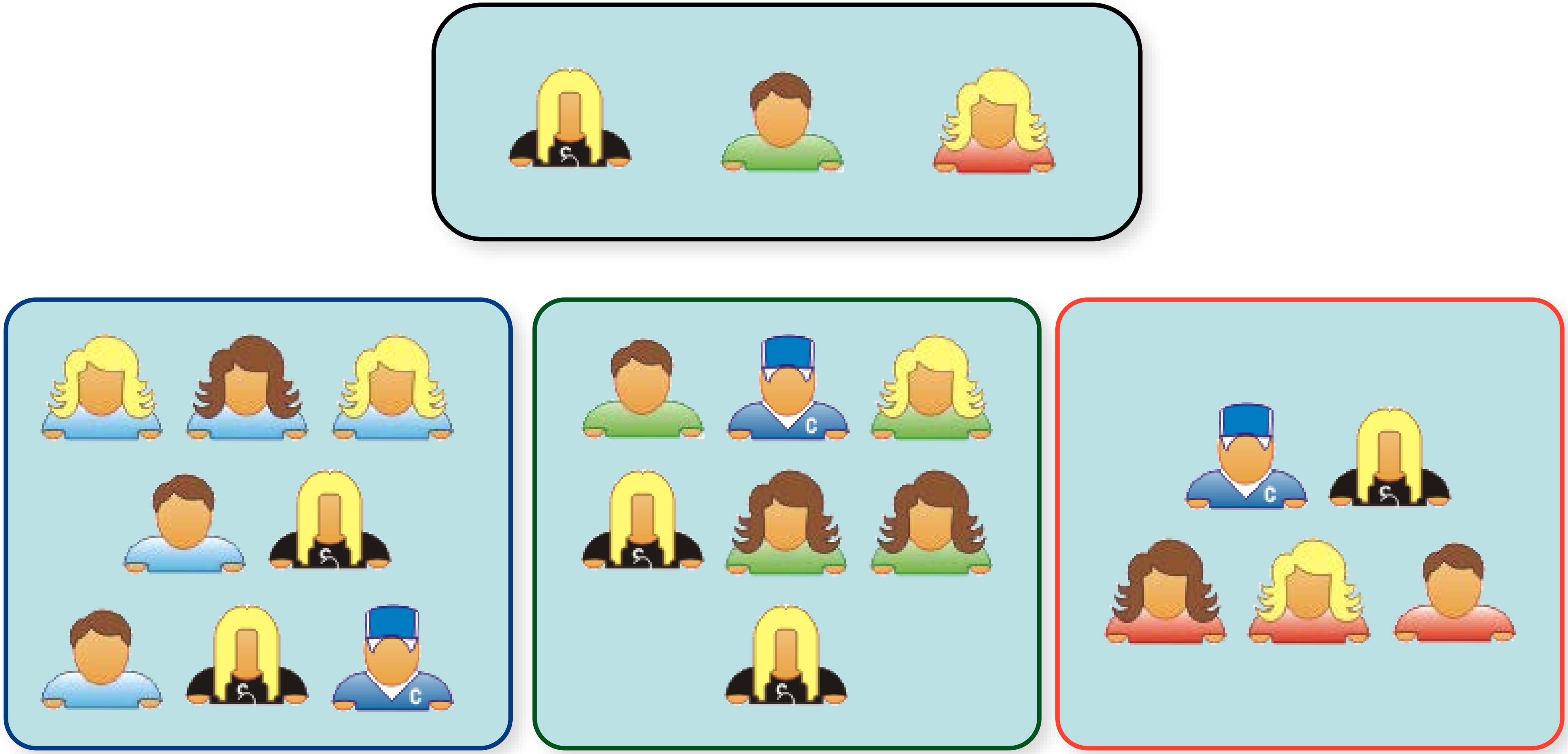
Scalability

- Typical individual team is 7 ± 2 people
  - Scalability comes from teams of teams

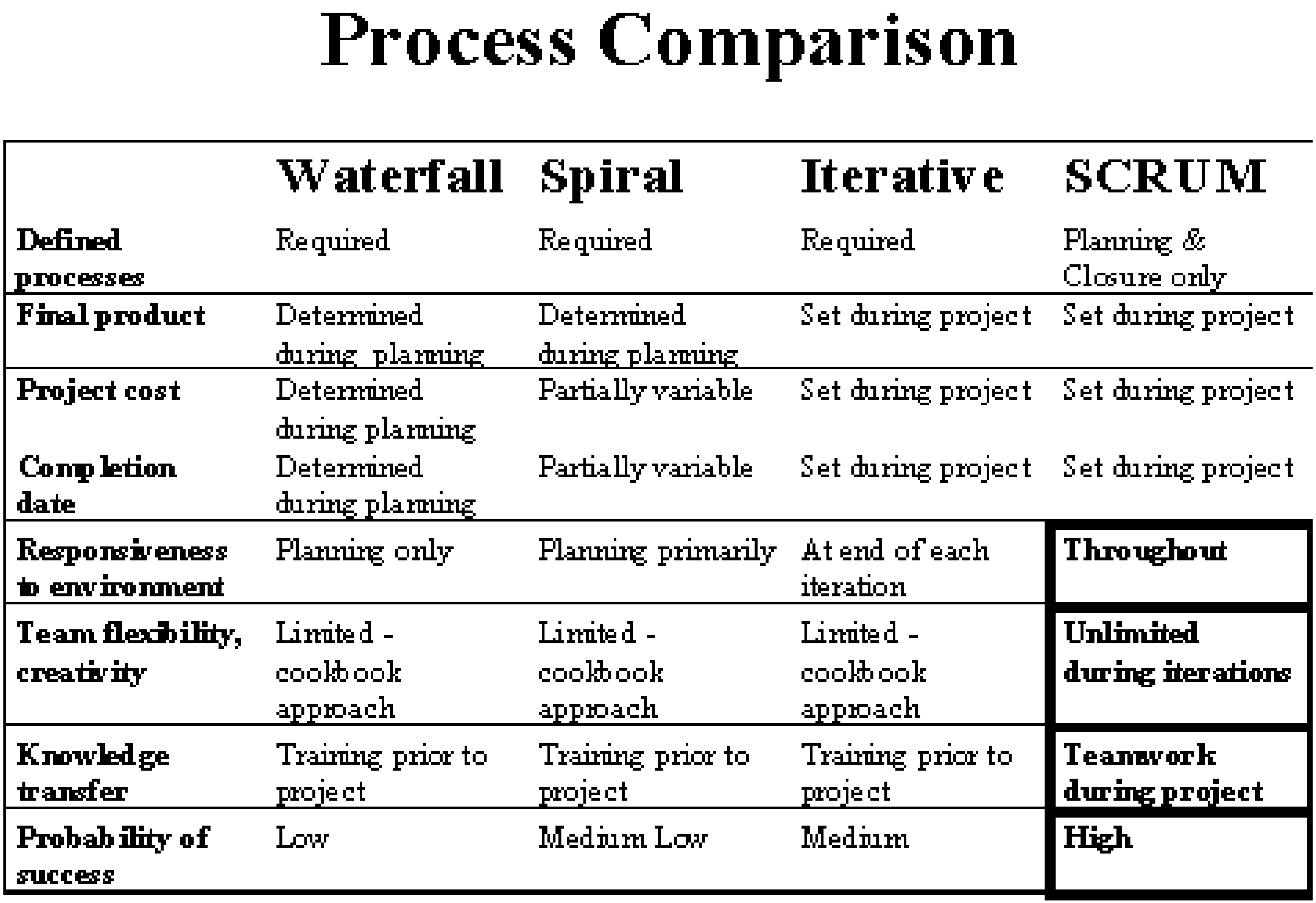
- Factors in scaling
  - Type of application
  - Team size
  - Team dispersion
  - Project duration

- Scrum has been used on multiple 500+ person projects

Scaling: Scrum of Scrums



Scrum vs. Other Models



Process Comparison

	Waterfall	Spiral	Iterative	SCRUM
Defined processes	Required	Required	Required	Planning & Closure only
Final product	Determined during planning	Determined during planning	Set during project	Set during project
Project cost	Determined during planning	Partially variable	Set during project	Set during project
Completion date	Determined during planning	Partially variable	Set during project	Set during project
Responsiveness to environment	Planning only	Planning primarily	At end of each iteration	**Throughout**
Team flexibility, creativity	Limited - cookbook approach	Limited - cookbook approach	Limited - cookbook approach	**Unlimited during iterations**
Knowledge transfer	Training prior to project	Training prior to project	Training prior to project	**Teamwork during project**
Probability of success	Low	Medium Low	Medium	**High**

Credits, References

- Mike Cohn, Mountain Goat Software
  www.mountaingoatsoftware.com
- *Scrum and The Enterprise* by Ken Schwaber
- *Succeeding with Agile* by Mike Cohn
- *Agile Software Development Ecosystems* by Jim Highsmith
- *Agile Software Development with Scrum* by K. Schwaber and M. Beedle
- *User Stories Applied for Agile Software Development* by Mike Cohn

- www.agilescrum.com/
- www.objectmentor.com
- jeffsutherland.com/
- www.controlchaos.com/scrumwp.htm
- agilealliance.com/articles/articles/InventingScrum.pdf