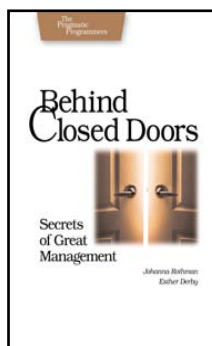
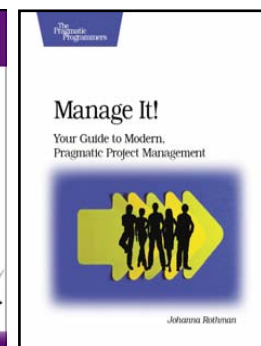
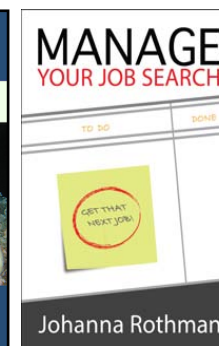
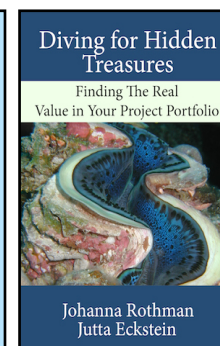
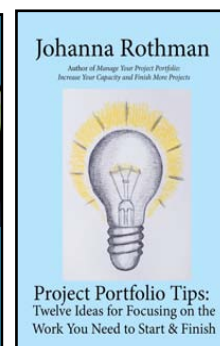
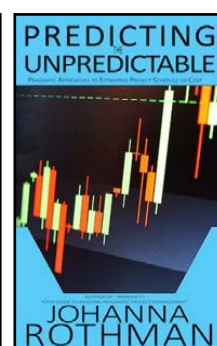
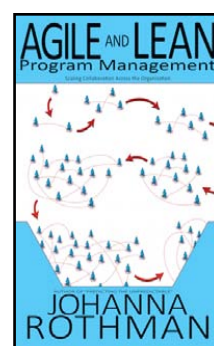
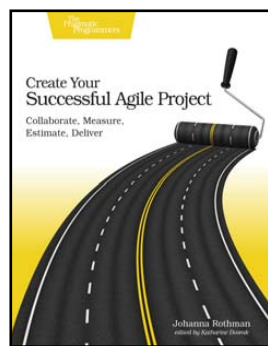


Free Your Agile Team: Focus on Flow Efficiency and Collaboration

Johanna Rothman
@johannarothman
www.jrothman.com



Why an Agile Approach?

- Can't release enough
- Can't change fast enough
- Too many defects discovered later
- Move from project-based people to product-orientation
- More...



Two Kinds of Agile Approach

Iteration-Based Agile

Requirements Analysis Design Build Test Release Deploy	Requirements Analysis Design Build Test Release Deploy	Requirements Analysis Design Build Test Release Deploy	Requirements Analysis Design Build Test Release Deploy	Repeat as needed ...	Requirements Analysis Design Build Test Release Deploy	Requirements Analysis Design Build Test Release Deploy
--------------------------------------------------------------------------	--------------------------------------------------------------------------	--------------------------------------------------------------------------	--------------------------------------------------------------------------	----------------------------	--------------------------------------------------------------------------	--------------------------------------------------------------------------

Each timebox is the same size. Each timebox results in running tested features.

Flow-Based Agile

Feature: Clarify Req't, Analysis Design Build Test Release Deploy	Feature: Clarify Requirement, Analysis Design Build Test Release Deploy	Feature: Clarify Requirement, Analysis Design Build Test Release Deploy	Repeat as needed ...	Feature: Clarify Requirement, Analysis Design Build Test Release Deploy	Feature: Clarify Requirement, Analysis Design Build Test Release Deploy
-------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------	----------------------------	----------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------

In flow, the team limits the number of features active at any time with WIP limits for each team activity. There is no timeboxing built into flow.

Commonalities

- Limit WIP
 - Timeboxes limit scope
 - Flow limits team's WIP
- Based on collaboration
- Focused on throughput
- Result in running, tested features



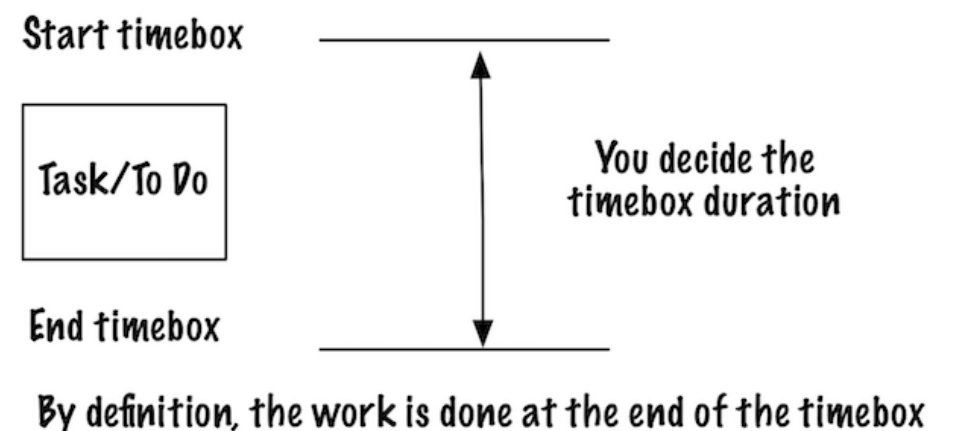
When to Use Each?

- Iterations work when the team can predict the work
- Flow works when the team has interruptions



Iterations, Flow and Cadence

- Iterations are a timebox that the team uses to define the work they commit to and deliver
- Flow is seeing how work flows through your team
- Cadence provides your project a rhythm for specific activities such as retros



**All models are wrong, but
some are useful
— George Box, 1976**

“Framework Hell”

- “A-gile” overtaken by frameworks
- “Mine’s better than yours”
- Peer pressure
- Certifications!!!
- Coaches, coaches, coaches



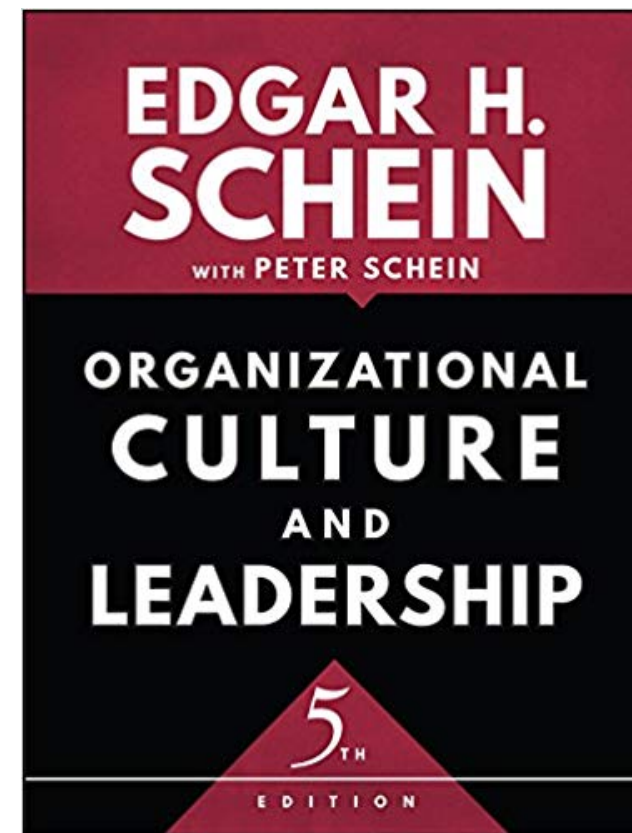
Frameworks Don't Change the Culture

- The team isn't a team
- Managers want to measure people, not the team
- Too many people (managers and team members) want 100% utilization
- The reward system recognizes and rewards people for individual work



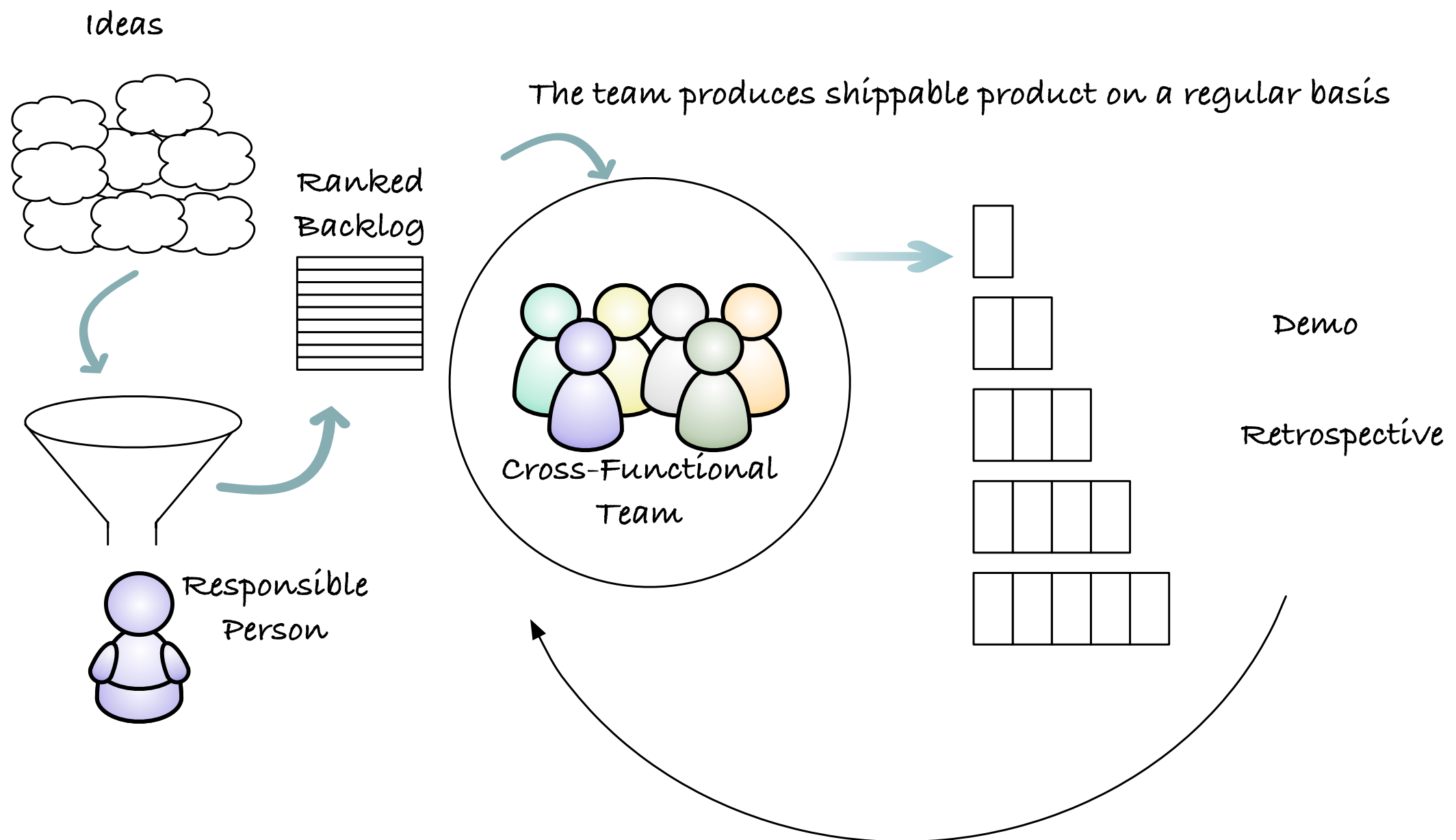
Consider Your Culture

- What can we discuss?
- How do we treat each other?
- What do we reward?



Johanna's General Agile Picture

General Agile Picture

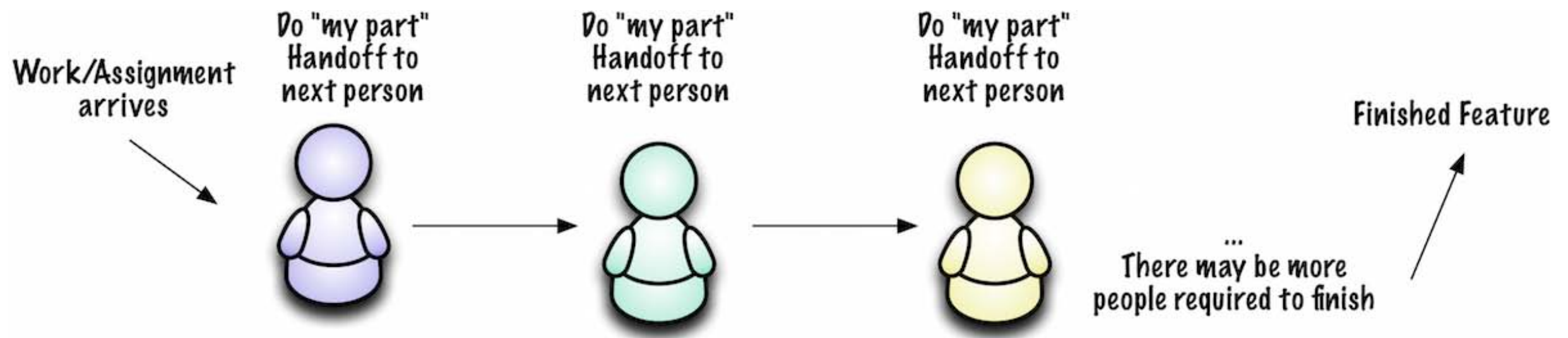


Agile culture

	Agile culture	Your culture
What can we discuss?	Almost anything because we collaborate on everything	?
How do we treat each other?	With respect and courtesy	?
What do we reward?	Team effort, not individual effort	?

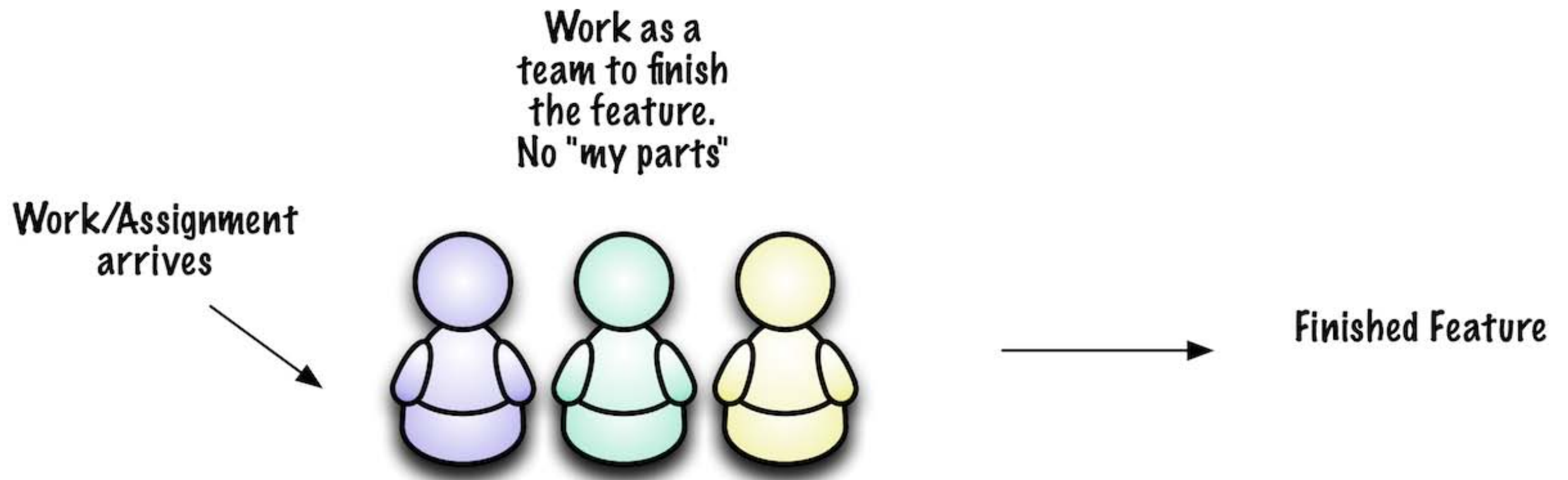
Resource Efficiency

- Focuses on (and measures) each person's individual contribution
- Creates and retains experts (which creates multitasking and many more Costs of Delay)



Flow Efficiency

- Focus on team's throughput (outcomes, not outputs)
- Optimize “up” at the level of the team



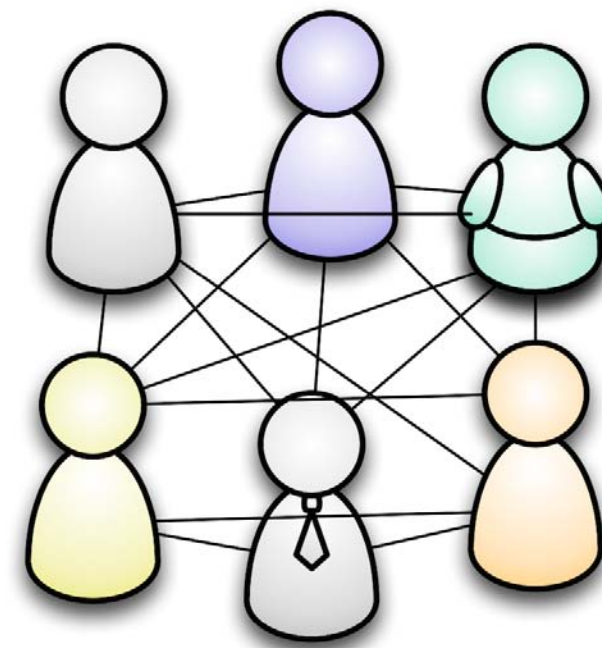
What Can You Do?

1. Work as a team
2. Visualize the work and the bottlenecks
3. Measure throughput
4. Retrospect and improve



Define Agile Team

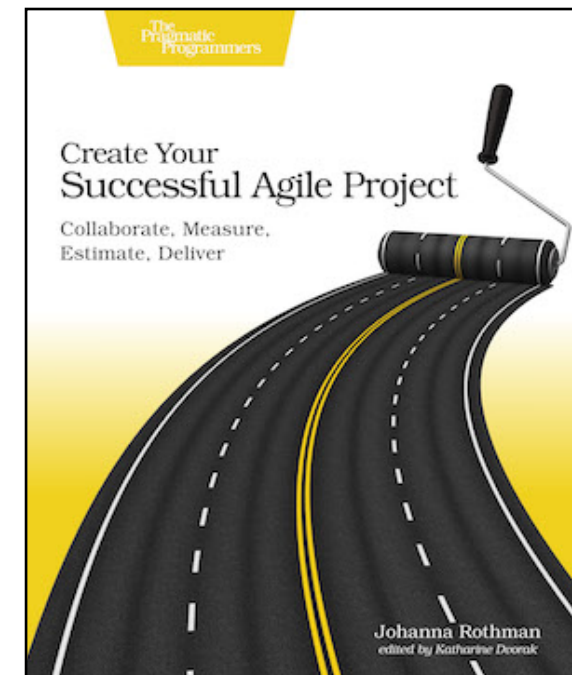
- Enough people to cover the work
- Specific and clear goal
- Collaborate on interdependent work
- Small enough that they can work together
- Team owns all of its work



Total nodes= 15

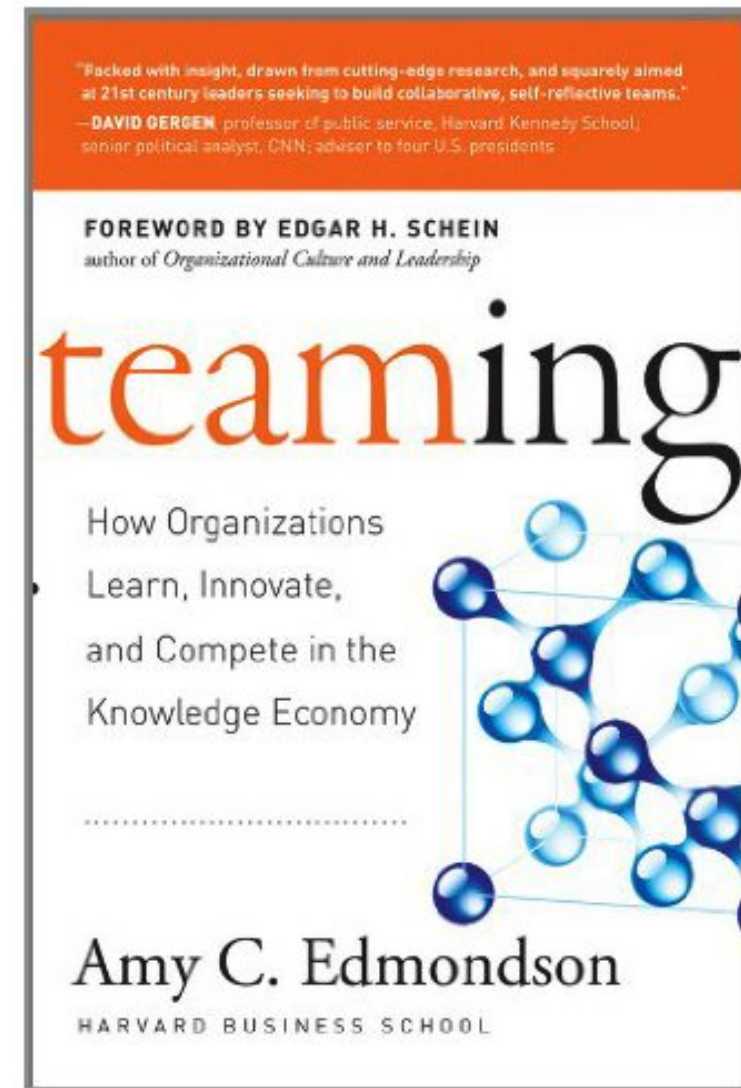
Do You Have a Team?

- Clear charter (you know what you have to do)
- Release criteria (you know when you're done)
- Psychological safety



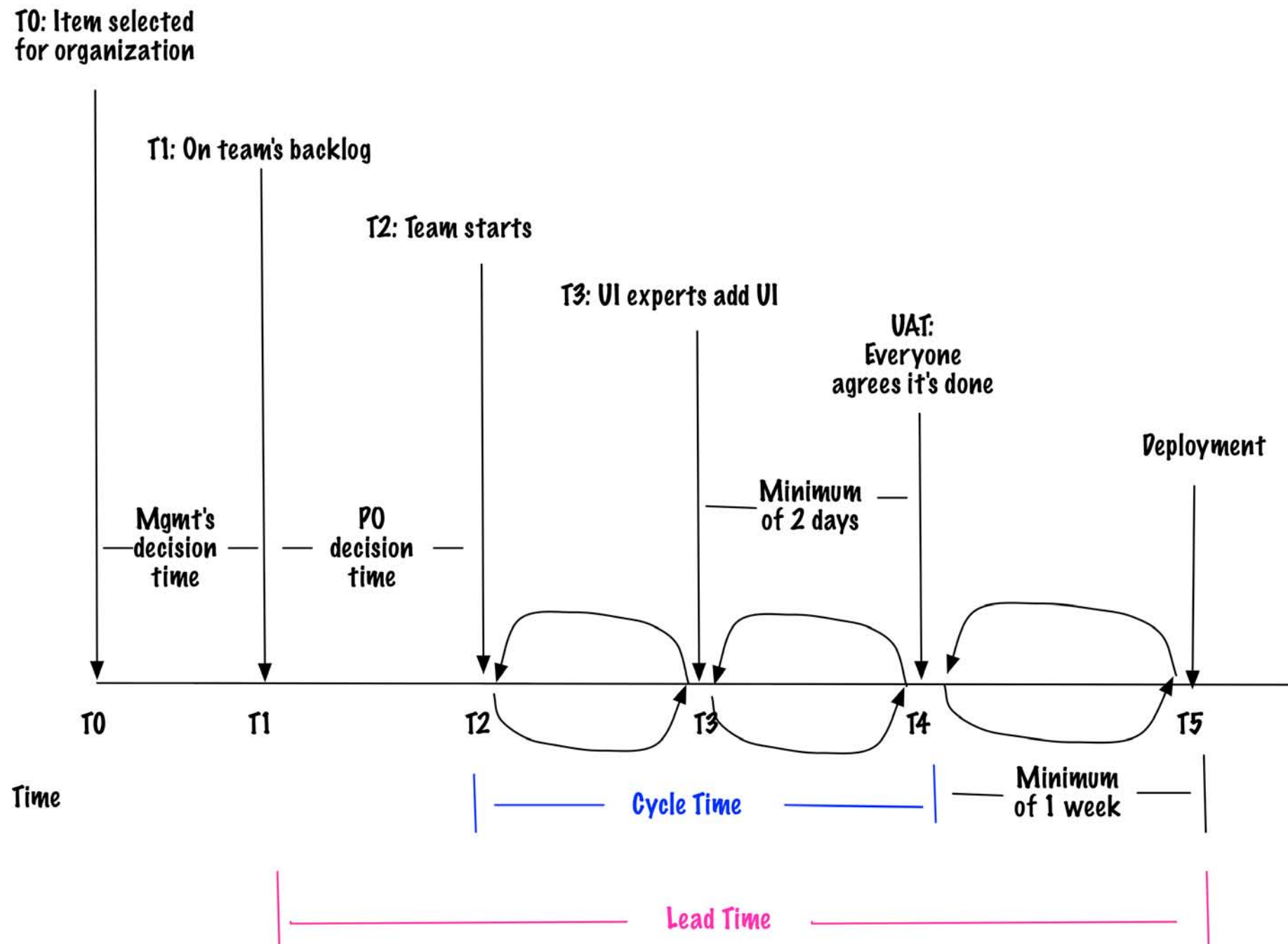
Psychological Safety

- Encourage learning from small experiments
- Use clear and direct language
- Admit when we don't know
- Acknowledge when we fail
- Set boundaries for what is a personal or team decision and what is not.

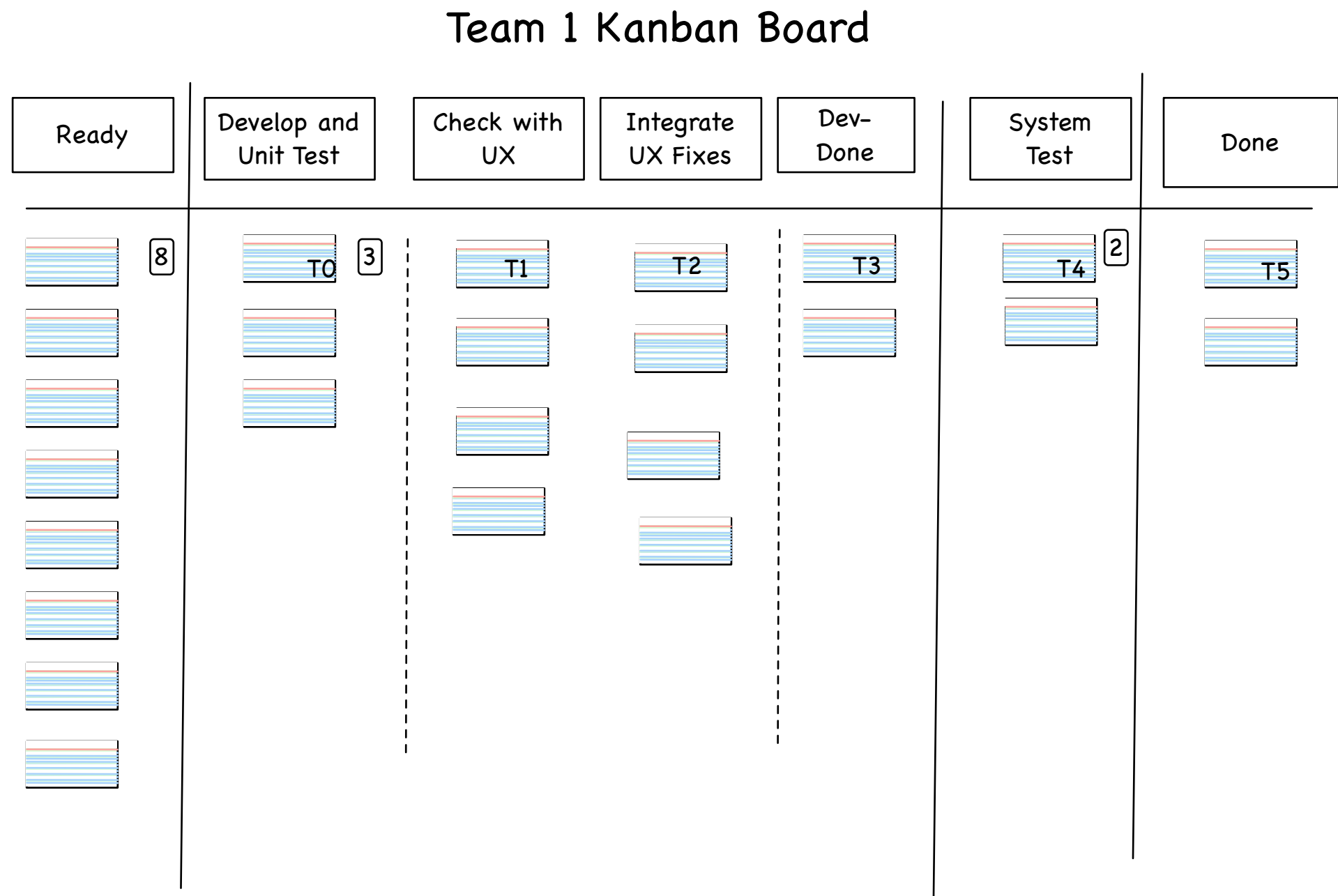


Visualize Work & Bottlenecks

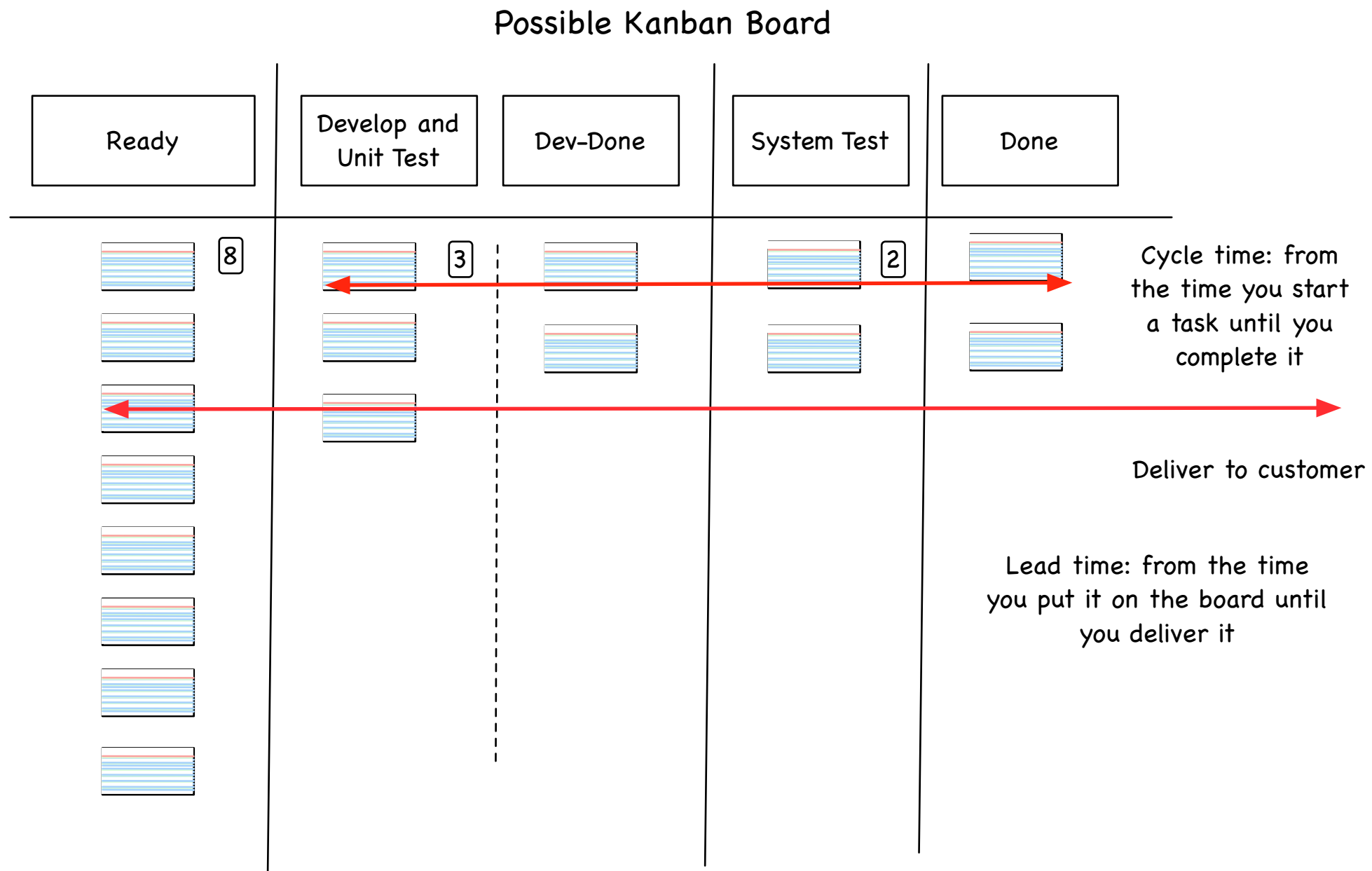
Cliff's Team's Organizational Lead and Cycle Time



Kanban Shows the Work



Cycle and Lead Time



How to Use Cycle Time for Estimation

Story	Story Start Day	Story End Day	Story Duration
1	Day 1	Day 3	2 days
2	Day 3	Day 4	1 day
3	Day 4	Day 6	3 days
4	Day 7	Day 8	2 days
5	Day 8	Day 10	2 days
Totals:			
5 Stories	10 days	Average Cycle Time:	2.4 days

Stop Measuring Velocity

- Velocity is a measure of capacity, not productivity
- Velocity varies with complexity and team familiarity
- Not always predictable
- Individual to each team, and can vary with domain
- Misused when it's “dev” velocity and “QA” velocity

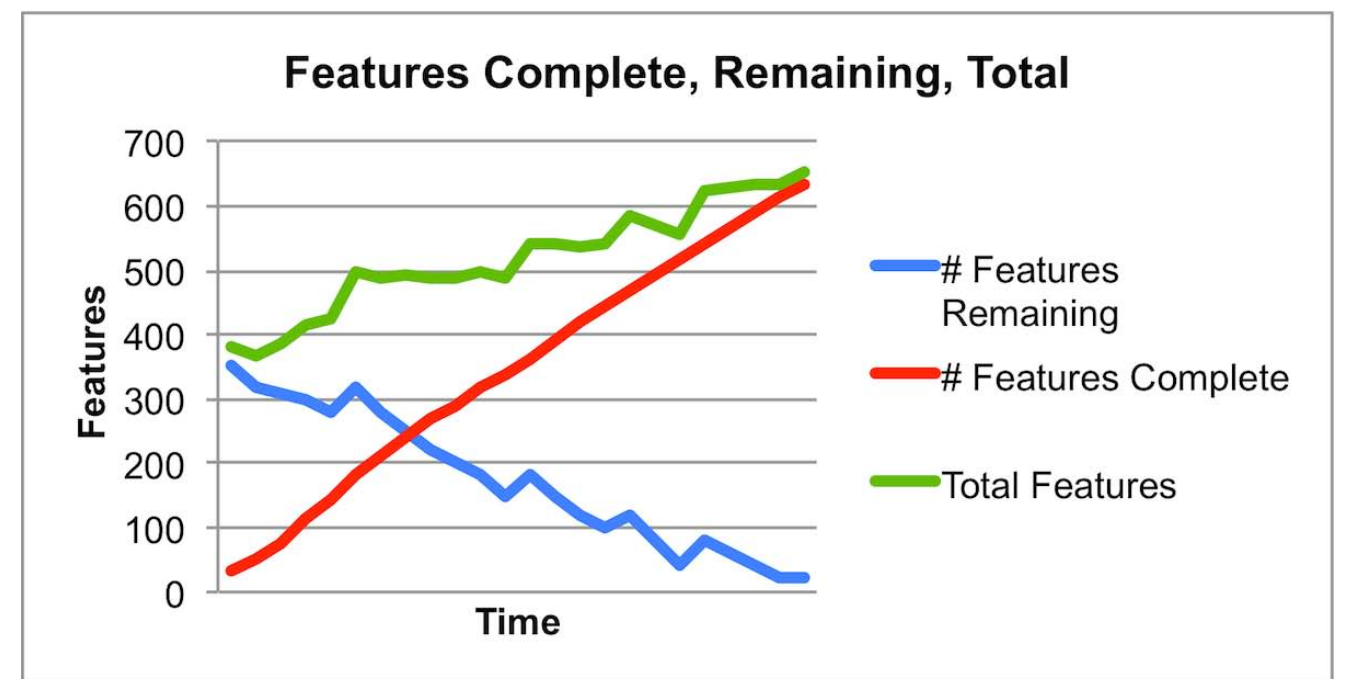


Team-Based Measures

- Cycle time
- Features completed, etc
- Product backlog burnup
- Cumulative flow and WIP

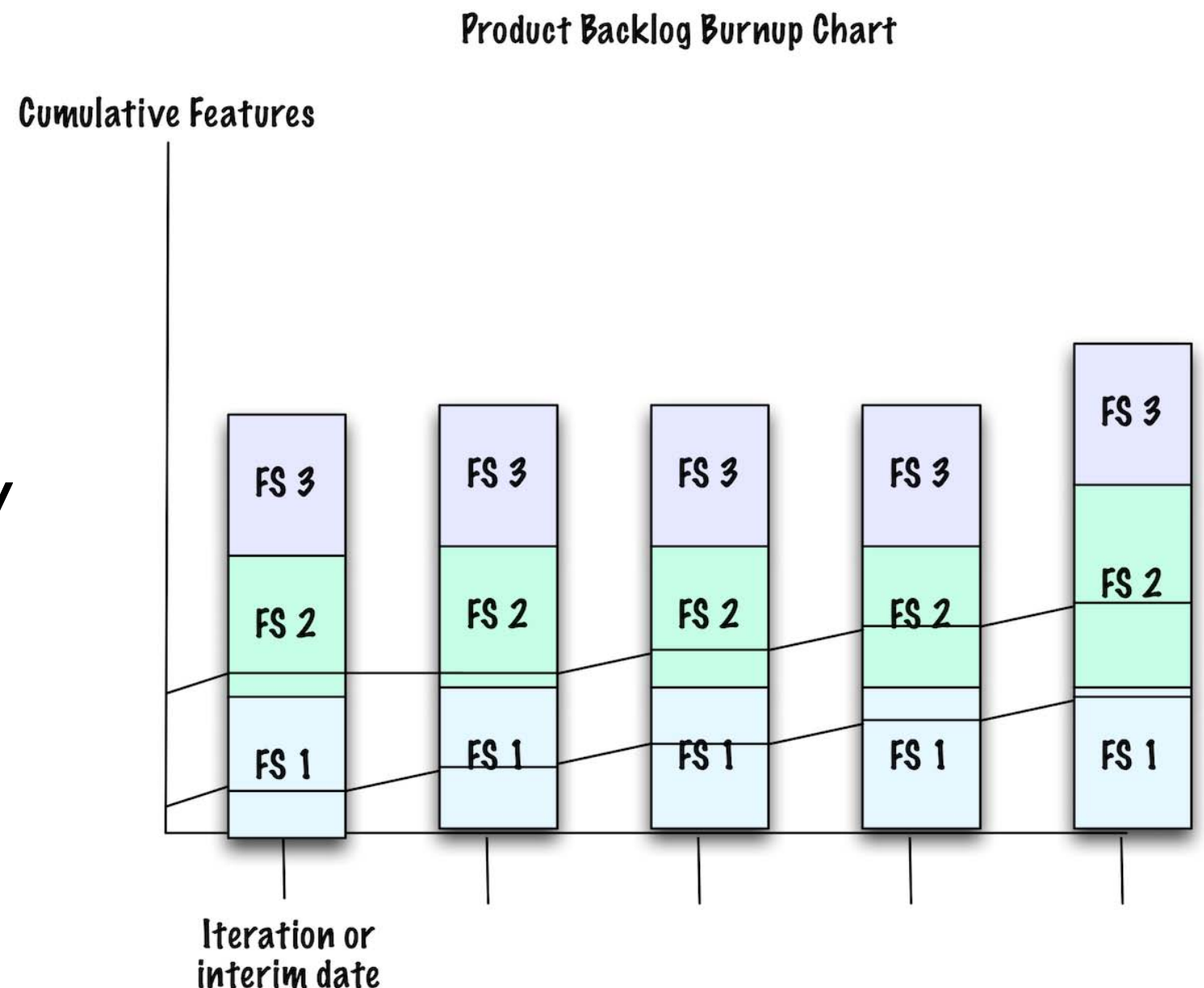
Measure Completed Features

- Completed features (running, tested features)
 - Your customers use them
 - You can release them
 - They are valuable
- Include total and remaining features so we have a sense of where we are
- Depends on deliverables, not epics or themes



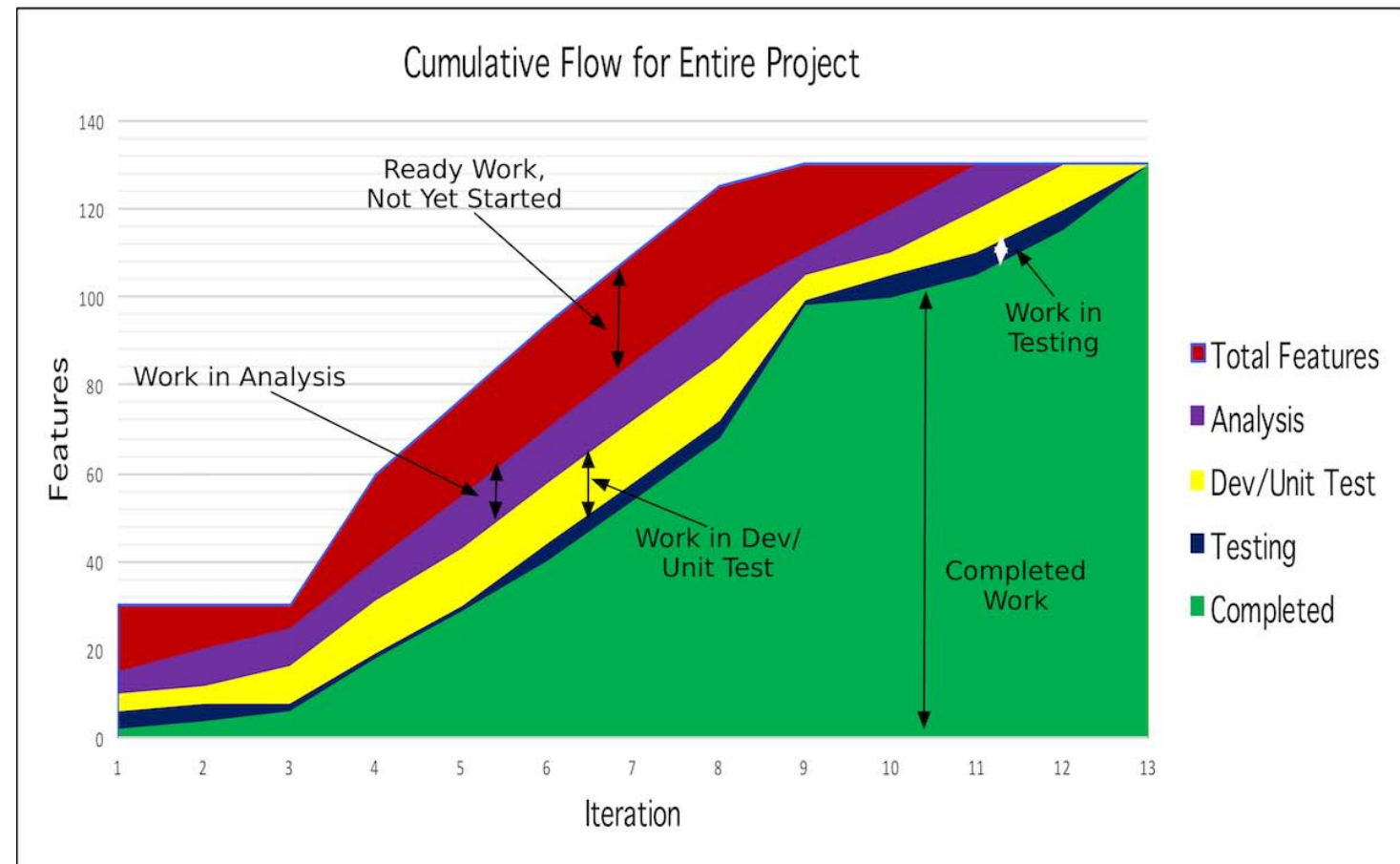
Product Backlog Burnup

- Real earned value
- Partial answer to “Where are we?”
- Shows value feature-by-feature
- Shows when features grow



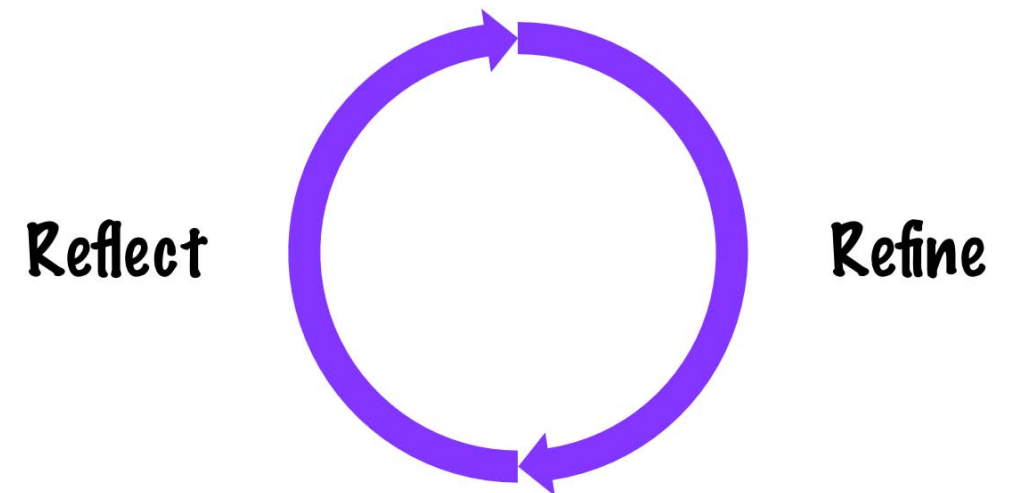
What Do You Want Less of?

- Work In Progress (across entire program)
- How often release
- Defects
- Other “Less of”:
 - Multitasking
 - ?



Retrospect and Improve

- Retrospectives
- Kaizen
- Choose one thing to experiment with every week or two
- This is more important than any other meeting you have



How Agile Approaches Change the Culture

Culture Changes From	Culture Changes To
Individual work	Collaborative work
Work assigned by someone else	Team members select work
Resource efficiency thinking and metrics	Flow efficiency thinking and measures
Management-planned details	Facilitated conversations and decisions
Gantt Charts and other documents as plans	Working product and empirical measures (and documents) to guide further work
Single-loop planning	Responding to and encouraging change with double-loop planning
Only understanding product quality at the end of the project	Continual learning and improvement of product quality as the team proceeds

Managers Change...

From

To

100% Utilization

Flow work through the team

Measure effort

Measure outcomes

Measure and “manage” individuals

Peer-based feedback

Individual recognition

Team-based recognition

Start Here to Free Your Agile Team

1. Work as a team
2. Visualize the work and your bottlenecks
3. Measure throughput
4. Retrospect and improve
5. Invite your managers to change the culture



Let's Stay in Touch

- Pragmatic Manager:
 - www.jrothman.com/pragmaticmanager
- Please link with me on LinkedIn

