



## 20 Great Agile Debates

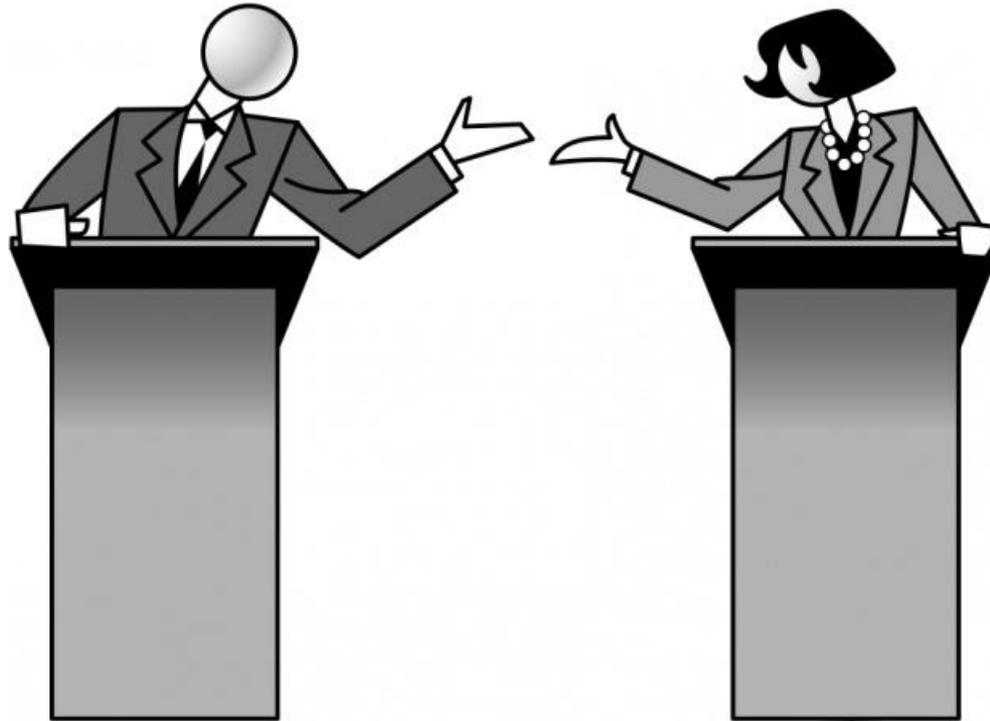
Rachael Wilterdink, CBAP | PMI-PBA | PSM | CSM

May 22, 2018



- **To understand some common items of confusion about agile**
- **To hear all sides of the debate, in order to make informed decisions**
- **To gain clarity about agile best practices**

# The Great Agile Debates



# 1) Do you need a Sprint 0?



## Yay?

- It helps team members to understand the work to be started
- Boosts developer confidence for tackling the work
- Allows the developers to self-organize to find the best way to deliver the project
- Gives team time to form and get to know each other

OR

## Nay?

- It establishes a precedent that certain sprints or sprint types have unique rules
- If the work is similar to work done previously, it's a waste of time
- There's nothing that can be potentially shippable at the end of it
- Doesn't usually result in an increment valued by the Product Owner
- It's been misused to describe the planning that occurs before to the first sprint

## 2) How much documentation is needed?

A little...



vs.



A lot...



### 3) Is there a “right” way to write user stories?



**As a** / I am a...

Role / User Type / Actor /  
Persona / Who

**I want** / would like / need /  
must have...

Feature / Functionality / Goal /  
Action / What

**So that** / so I can...

Business Value / Reason /  
Benefit / Why

## 4) What's the best way to document acceptance criteria?



### A bulleted list...

- A user cannot submit a form without completing all mandatory fields
- Information from the form is stored in the registration's database
- Protection against spam is working
- Payment can be made via credit card
- An acknowledgment email is sent to the user after submitting the form

vs.

### Behavior-Driven Development

Given: [pre-conditions]

When: [action/trigger]

Then: [expected outcome]

## 5) How long should sprints be?



# 1, 2, 3, 4... more?

1 week	2 weeks	3 weeks	4 weeks
Not common, but not unheard of	The most common sprint length, especially for IT / software development work	Second most common sprint length	Not usually recommended, but also not totally unheard of
This may be the best way to “fail fast”			

*Anything longer than 4 weeks is no longer considered agile...*

## 6) Does agile have to be run the “same” in an organization?



Yay?

- Organizations like consistency
- Enables teams with cross-functional members to move in/out of teams more easily
- Business partners know what to expect
- Terminology is the same

OR

Nay?

- This kind of goes against agile principles
- Team should be able to inspect and adapt, which means changing and evolving to improve performance
- People with different experiences are going to do things differently
- Trying to fit agile teams into cookie cutter molds won't be well-received

## 7) Which “flavor” of agile is best?



Agile Flavors	Creator	Description
eXtreme Programming (XP)	Kent Beck	Widest known, developer-focused for small teams
Scrum	Ken Schwaber	Focused on management practices
Feature-Driven Development (FDD)	Jeff De Luca	Agile approach closest to conventional development
Lean Software Development	Poppendieck	Inspired by Toyota Production System, particularly its product development practices
Crystal Clear methodologies	Alistair Cockburn	Set of methodologies
Dynamic Systems Development Method (DSDM)	DSDM Consortium	Popular approach for Rapid Application Development (RAD) among European IT shops since 1999

## 8) Can agile co-exist with waterfall?



*Yes, but...*

### PROS

- Reality is that most organizations will likely have need of both
- Provides the flexibility to choose the best approach for the project
- Gives the organization the option to take a hybrid approach, if needed (program run traditional, development run agile)
- Gives employees the opportunity to choose the best fit for them

VS.

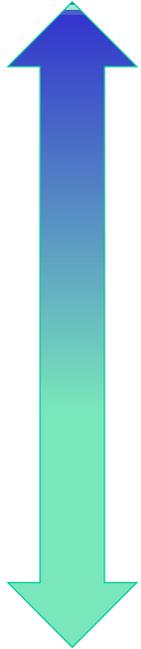
### CONS

- PMOs have a hard time accommodating agile and managing both
- It can introduce confusion and inconsistent terminology and experiences
- It may create an “us” versus “them” mentality between teams
- Inconsistent delivery may result
- Can be difficult when there are dependencies

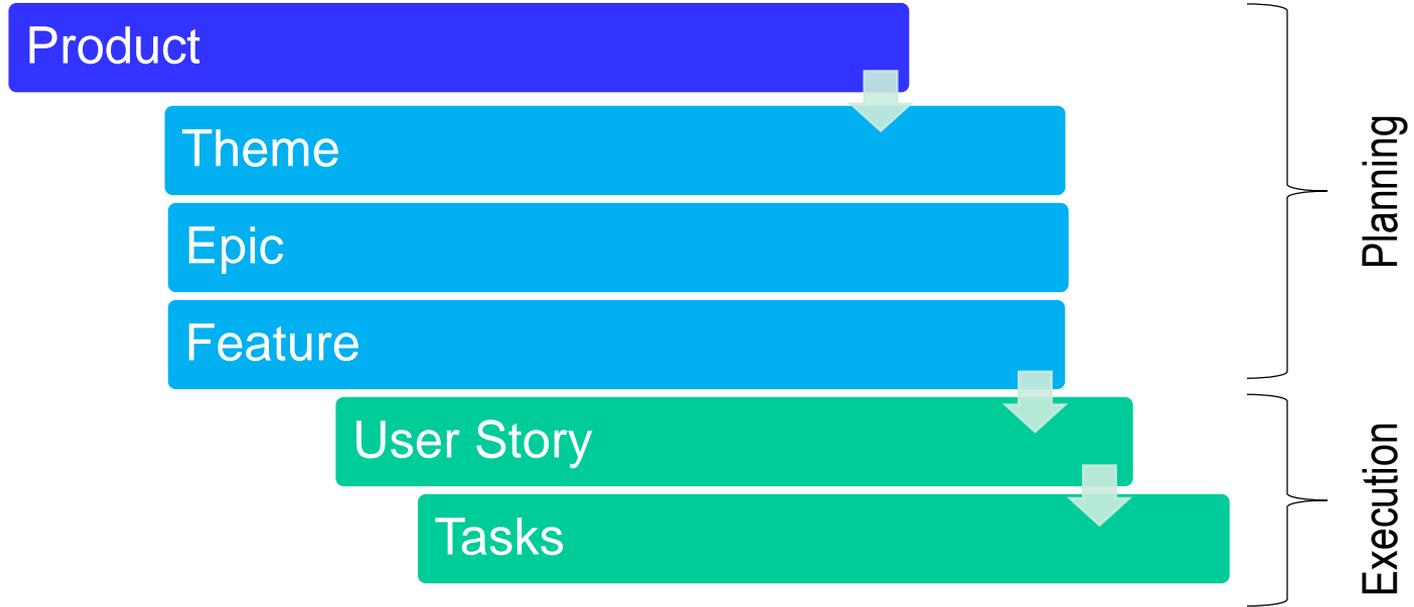
# 9) Story size – what's an epic, theme, feature...?



Low Detail



High Detail



## 10) Can the Scrum Master be a team member, too?



### Yay?

- If resources are limited, it may sometimes be necessary
- Scrum Master may not always be a full-time position
- Could rotate team members in this role to give them experience doing it

### OR

### Nay?

- It introduces the possibility of competing priorities for that person
- Context switching between roles reduces effectiveness and productivity
- It's harder to conduct retrospectives wearing both hats
- Could be a conflict of interest

## 11) Can agile work with offshore/distributed teams?



*Well, it's not really ideal for agile, but these tips may help...*

- Meet in person, at least once
- Co-locate as often as possible
- Use liaisons
- Record important sessions
- Keep open lines of communication
- Create a glossary of terms
- Use available communication tools
- Shared Calendar
- Establish overlapping working hours
- Build trust

## 12) What's the best way to estimate?



### Story Points

vs.

### Ideal Days

Story Points		Ideal Days	
PROS	CONS	PROS	CONS
Abstract; measures relative size	It's difficult for some to grasp the concept	Better understood by most	Individual estimates don't foster collaboration
Accounts for team differences	Tendency to inflate story points	May make the project less likely to fail	Tendency to forget team's experience
Encourages high-level discussion & fosters collaboration	Confusion between time and relative estimate	More intuitive than story points	May not account for reality; not all time at work is productive
Ignores variations of team members' productivity		May be a way to transition to story points	The ideal day will vary from person to person

## 13) Is it OK to add items to the current sprint?



### Yay?

- Only if everything planned for was completed
- If you plan for handling some amount of “unknowns” as a part of the sprint

### OR

### Nay?

- Once a Sprint begins, it's the equivalent of a baseline
- While agile means to be adaptive to change, that doesn't mean within the current sprint
- Team should focus on completing planned work and add changes to the backlog for prioritization in the next sprint

# 14) Is a wall or a tool better for management?



VS.

Wall



Tool

The screenshot shows a 'Product Backlog' interface with a table of tasks. The table has columns for Order, Title, State, Effort, and Iteration Path. The tasks are listed in a numbered order, with some grouped under 'Sprint 43', 'Sprint 44', and 'Release 3.0'.

Forecast	Order	Title	State	Effort	Iteration Path
Forecasting based on velocity of 15					
	1	Complex Deploy (with DE)	Approved	8	InRelease
	2	Automated Test Integration in InRelease	Approved	3	InRelease
	3	BitDark Deployer	Approved	3	InRelease
	4	ClickOnce Deployer Tool (Document)	Approved	0	InRelease
	5	XBAP Deployer Tool (Document)	Approved	0	InRelease
Sprint 43	6	Actions: Registry Tool	Approved	0.5	InRelease
	7	Add Business Rules to Deployment Editor	Approved	1	InRelease
	8	Change Release screen to accommodate for actions and deployment e...	Approved	3	InRelease
	9	Apply a deployment to another environment	Approved	5	InRelease
	10	Edit values in a table	Approved	3	InRelease
	11	Actions: Merge Deployers and Tools and keep a list of Actions instea...	Approved	1	InRelease
	12	Navigating away from a dirty object is problematic	Approved	1	InRelease
	13	[DN] Monitor service detects stalled deployment when there is not...	Approved	1	InRelease
Release 3.0	14	Allow more place to enter file names in component file extension fil...	Approved	0	InRelease

## 15) Can an agile team work on more than one project at a time?



It can work if...

- The Product Owner controls prioritization for all projects
- Product backlogs feed into one sprint backlog
- Multiple Product Owners can choose one person to prioritize
- You choose a “theme” to organize your sprints around

vs.

It won't work because...

- It is inefficient for team members to be context switching between projects
- Skillsets or expertise may not be available for different projects
- Divided effort leads to slower delivery

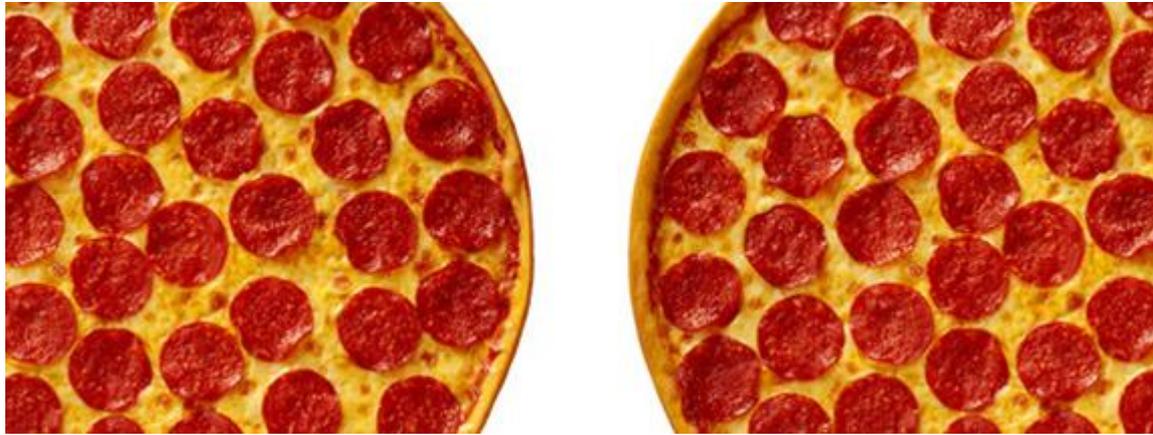
## 16) What is the optimal team size for agile?

*How many people can you feed with two pizzas?*

5

7 ± 2

9



3

6 ± 3

9

Large teams are inefficient; no more than 9

# 17) Should agile teams stay together?



## Yay?

- In a perfect world, yes
- In order to become high-performing, you need to keep the team together
- The longer a team is together, the faster their velocity becomes
- Each time team members enter/exit, the team has to re-form
- Once you become a high-performing team, it's "magical"

## OR

## Nay?

- It can be impractical to keep teams together, depending on the skills of the individuals
- It may be an expensive proposition to keep a team intact, if there's not sufficient work to feed the team
- People may get bored if they're working with the same product or people for too long

## 18) How are defects best handled in agile?



---

### Current Sprint

---

Fix them right away

### Previous Sprints

---

Add them to the backlog

Product owner prioritizes along with all other items

OR

Keep a separate backlog just for bugs

---

*Be sure to set clear definitions of defect severity...*

# 19) Can a hybrid approach work?



## Yay?

- Combines the benefits of both Agile and Waterfall processes
- Adapts to changing requirements
- Helps fill in gaps left by each method on its own
- May work better when advanced planning and budgeting is needed
- Better addresses the challenge of distributed software development

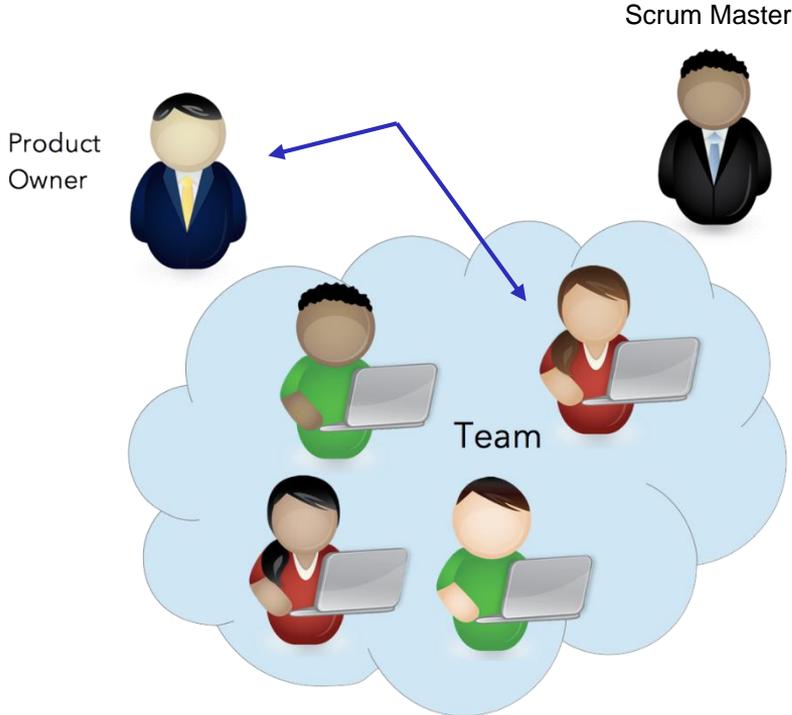
## OR

## Nay?

- Far from perfect and it's a compromise
- It's in a nascent stage
- Requires extensive planning
- May yield results, but won't ever be as optimal as those realized when operating fully Agile
- Introduces confusion about roles
- The two methodologies often conflict

# 20) Is there a role for Business Analysts in agile?

## Potential Roles



## Business Analysis Activities

- Write user stories with the Product Owner
- Analyze Product Backlog Items
- Work with Product Owner to ensure clear & complete Acceptance Criteria
- Prepare stories for Backlog Grooming
- Mock-up User Interfaces with the Product Owner
- Clarify and define User Experience criteria with team & Product Owner
- Build test scenarios
- Look ahead at releases and identify what must be done for the next release



# THANK YOU!

Rachael Wilterdink, CBAP | PMI-PBA | PSM | CSM

[rwilterdink@skylinetechnologies.com](mailto:rwilterdink@skylinetechnologies.com)

[www.skylinetechnologies.com](http://www.skylinetechnologies.com)