

# Why SCRUM sucks!



IOANNIS KOSTARAS  
AGILE CRETE 2016

# Agenda

2

- What is SCRUM?
- Why it sucks?

# What is Scrum?

3

# Scrum in a Nutshell

4

- **SCRUM** is
  - a methodology
  - an agile framework for software development
  - relies on **self-organizing, cross-functional teams**
  - **focuses on results**

# SCRUM

- Business Conditions & Requirements  
- Technology

5  
- Standards  
- Conventions  
- Guidelines

- Formal Evaluation Criteria



Product Owner

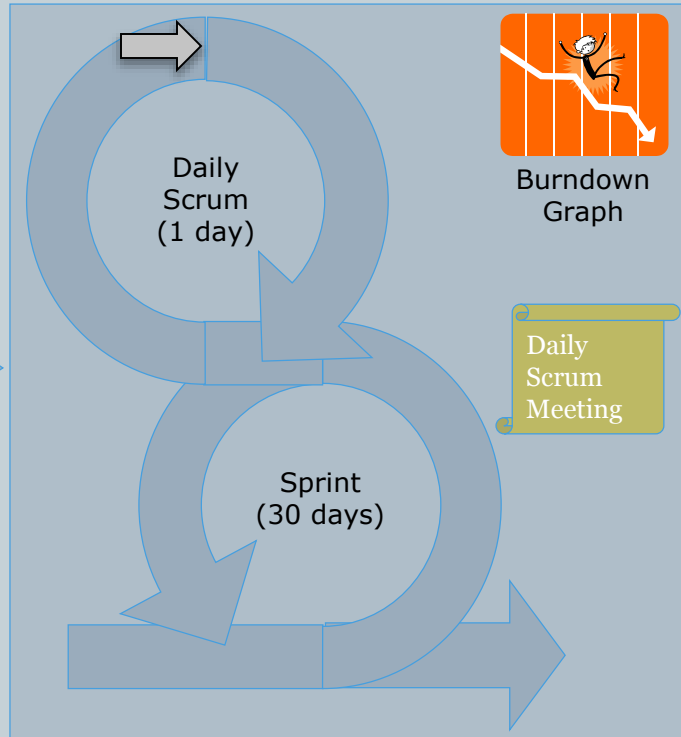
Product Backlog

Sprint Planning Meeting

Sprint Backlog



Team



Executable Product Increment

Sprint Review

Impediments List



Scrum Master

# Product Owner

6

- *“...represents the voice of the customer and ensures that the Scrum Team works with the right things from a business perspective”*
- S/he does **not** micro manage the team; s/he is not a product manager.
- *“When the tasks and required time has been determined, the Product Owner lets go.”*
- *“The team members decide how the work is arranged and how assignments are distributed”.*



# Scrum Master

7

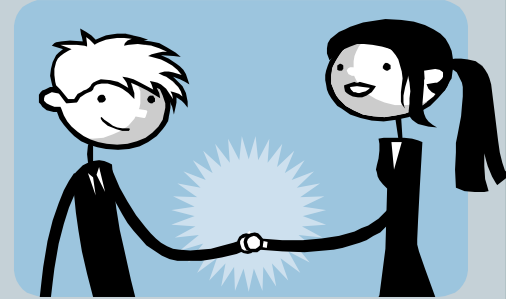
- *“The Scrum Master coaches the development team, **removes any possible impediments** and constantly works to ensure that the team has the best possible circumstances for realizing the goals fixed for the Sprint.”*
- S/he does **not** micro manage the team; s/he is not a product manager
- S/he does **more than** only collect the hours burned
- *“The team members decide how the work is arranged and how assignments are distributed”.*
- Team might require some leadership from time to time



# Team members

8

- *“The team members decide how the work is arranged and how assignments are distributed. There are no set project roles – **everyone should be able to swap tasks with another member.** Naturally, this does not prevent individual members from being experts”*
- In real life, a Scrum team consists of developers, testers, system engineers/ administrators, configuration managers, documentation experts, etc.
- **Cross-functional?**
  - **Not everybody can do everything**



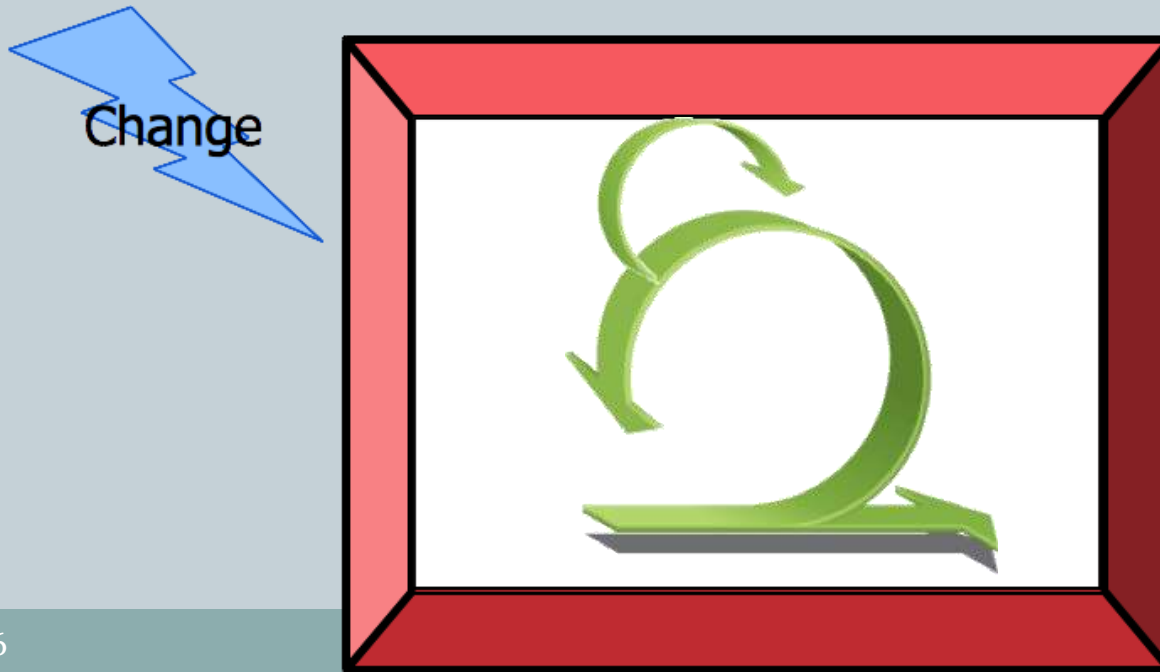
Team



# No changes during a sprint

9

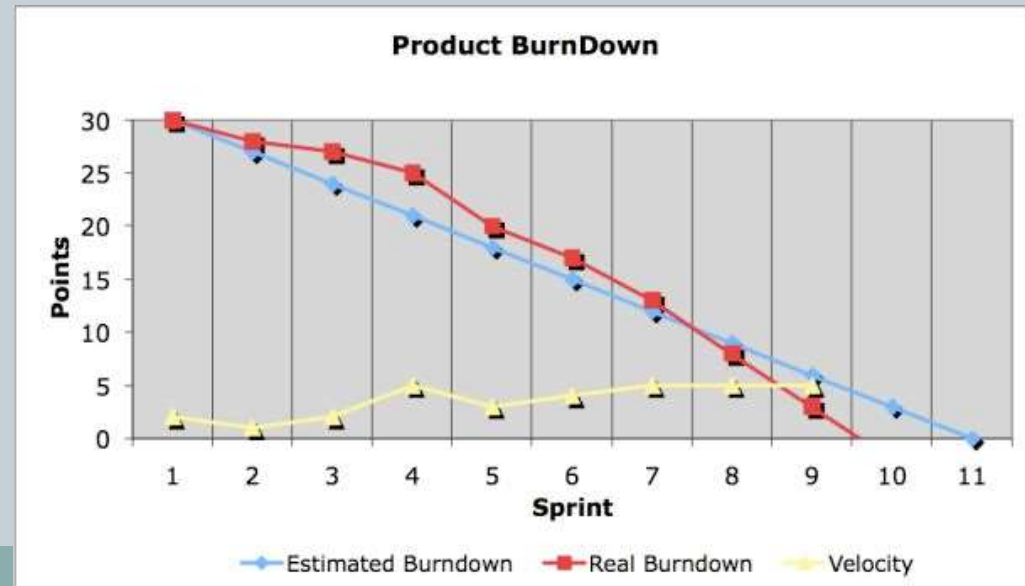
- Plan sprint durations around how long you can commit to keeping change out of the sprint
- In reality????
- What if you finish your tasks earlier than estimation?



# Predictions

10

- The rate of progress of a Scrum Team is called "velocity". It expresses the amount of e.g. story points completed per iteration.
- In reality the customer or the director or the PO decides when the features are to be ready →
- Agile methodologies require a change in the way people think.

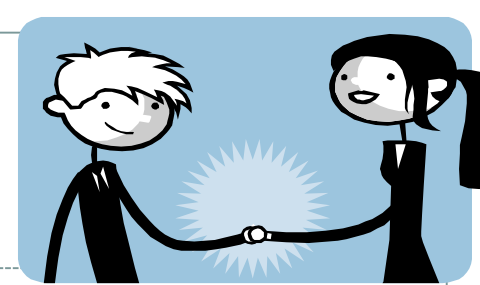


# Predictions (cont.)

- **Software development is**
  - a creative and experimental process
  - a complex system of multiple poorly understood feedback loops and interactions
  - an organic process of trial and error, false starts, experiments and monumental cock-ups
- **You can only predict something if you have done exactly the same thing before (a rare situation in software development)**
- **How accurate are your predictions?**



# Agile Values



Team

12

**Agile Teams prefer**

**Over**

Individuals and interactions

Processes and Tools

Working software

Comprehensive Documentation

Customer collaboration

Contract Negotiations

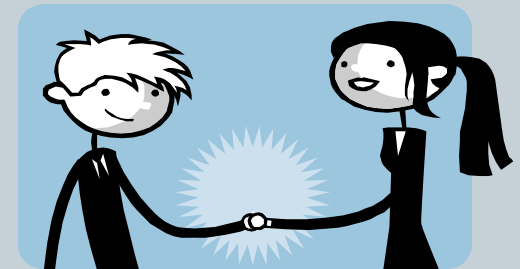
Responding to change

Following a plan

# Agile Values

13

- But is the organisation inline? Agile methodologies require a change in the way people think. A process cannot force organisational change.
- Scrum inside the team but waterfall outside of it?
- Minimum documentation for internal use but tons of other documentation to make management or customer happy by abiding to the procedures:
  - PMP
  - Release Note
  - SVD
  - ...



Team

# The meaning behind the rules

14

- All people seem to remember the rules of Scrum:
  - The definition of “done”
  - Morning meetings should take no more than 10’ ...
- BUT, they forget the meaning behind the rules.
- This is more evident on structural organisations who function based on procedures
- The purpose of the methodology is to focus on maximizing the team’s ability to deliver quickly and respond to changes better, however, a lot of teams are being caught up in the act of “doing” Scrum.

# What if ...

15

- **Product Owner**

- micro manages the team?
- asks questions all the time during the sprint meeting?
- sets milestones?
- proposes tasks and how long they will take?
- assigns tasks to team members?



# What if ...

16

- **Scrum Master**
  - simply collects the hours burned?
  - simply prepares the sprint report and the review meeting and retrospective?
  - does not participate in the planning?
  - does not do any other action?
  - does nothing when there are conflicts?





# What if ...

17

- **The team**

- does grooming in order to better identify the tasks and do better estimates?
- spends more time in planning than in doing?
- Cross-functional?
- « I will focus only on this task » attitude ?
- Nobody wants to take over one or more tasks due to various reasons (lack of knowledge, too risky, not well defined, requirements keep changing all the time, etc.)?
- Commitment before the spring?
- Horses vs. Donkeys?

# Must teams be collocated (?)

18

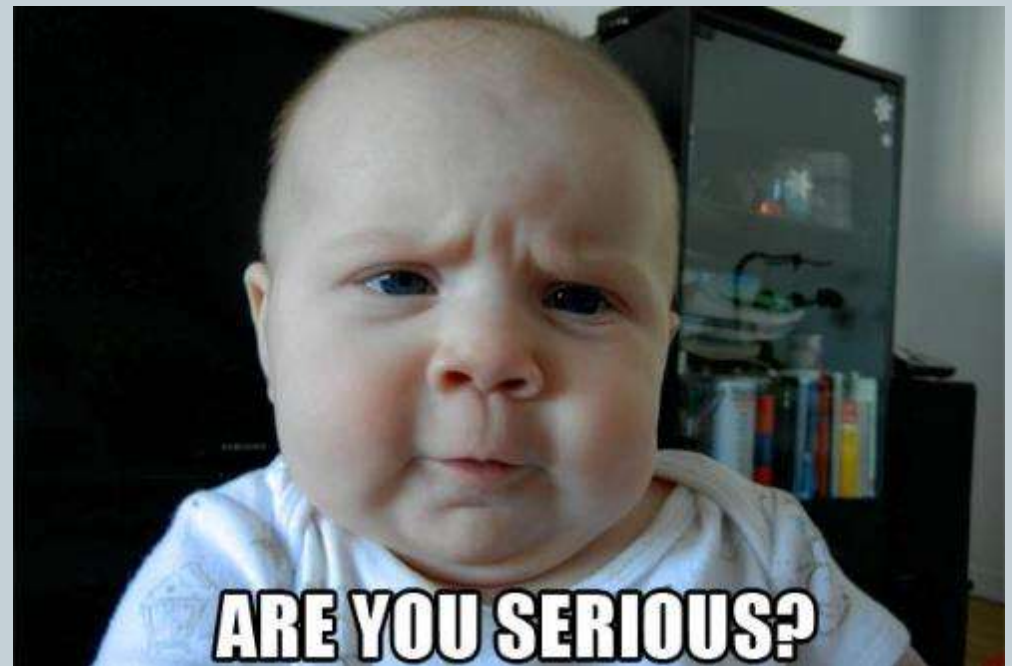
- Collocation allows team members to communicate more effectively and may become more productive  
→
- End up with people stacked in small rooms
- Collocation of team members is ***not*** mentioned in the SCRUM guide



# The Reality

19

- We do Scrum but my company is also:
  - CMMI
  - PMI
  - Prince2
  - ITIL
  - ...



# Scrum is used everywhere in my company

20

- Not only for software development but also for testing, integration, management...(Scrum madness)
- What if your product is “project based?”, i.e. has deadlines, etc.
- What if you need to support other teams (e.g. you are the documentation or quality/safety team) and you cannot afford to have a member in every scrum team?

# Tools?

21

- Software or paper (post-its)?
- If software then which tool?
  - Agilo, JIRA, Redmine,
- Does the tool make your life easier or more difficult?



# Looking Ahead

25

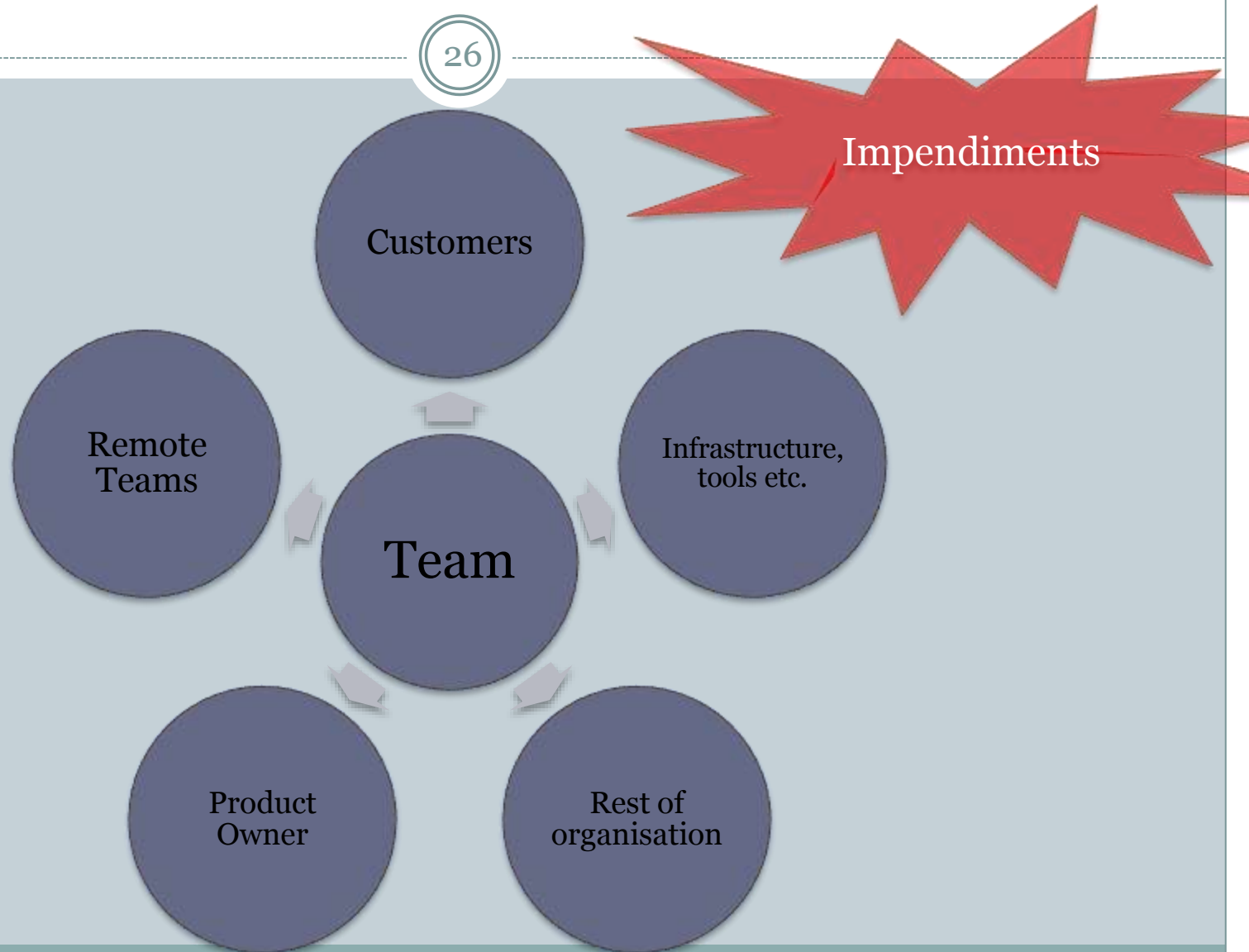
- Customers are interested in 3 things:
  - How much it will cost
  - What the product will look like
  - When it will be done
- However, agile stresses:
  - Agility - responsiveness to change and uncertainty
  - Not accurate high-level planning



Incompatible with agile methodologies

# Dependencies and Resources

26



# Epilogue

27



# Epilogue

28

- Easy to blame the methodology
- Difficult for people to change (paradigm shift)
- A process cannot force organisational change
- Finely grained management is a recipe for ‘talent evaporation’.
- Most developers dislike micromanagement
- What about software architecture?

# Questions

29

