

# Notes for Improving Scrum Clinic - Part 1 - Ver 34 (Work-in-Progress)

---

Joe Little — February 3, 2025  
Webinar — [LeanAgileTraining.com](https://leanagiletraining.com)

© Joseph Little 2025



# Intro



# Simple and Hard

---

- Scrum is...
- Easy: because you just follow the rules. And many of the basics are or will be obvious to you.
- Hard:
  - A different Mindset (for you & others)
  - Because there are all kinds of impediments out there
  - Hard for us, because we don't want to mislead you
  - Hard for you, because not easy to get advice on special situations
- Hard: because in general, while Scrum is simple, it is hard to do well.



## One Recommendation

---

- “**A Scrum Book**” by Jeff Sutherland, James Coplien et al.
- It’s full of about 94 Patterns in ~500 pages.
- Also: [ScrumPLOP.org](http://ScrumPLOP.org) or [ScrumBook.org](http://ScrumBook.org)
- Patterns movement.
  - See Christopher Alexander, “A Pattern Language.” “The Timeless Way of Building.”
  - See wikipedia (“A Pattern Language”).



Jeff Sutherland  
James O. Coplien  
The Scrum Patterns Group  
*edited by Adaobi Obi Tulton*



# Sections

---

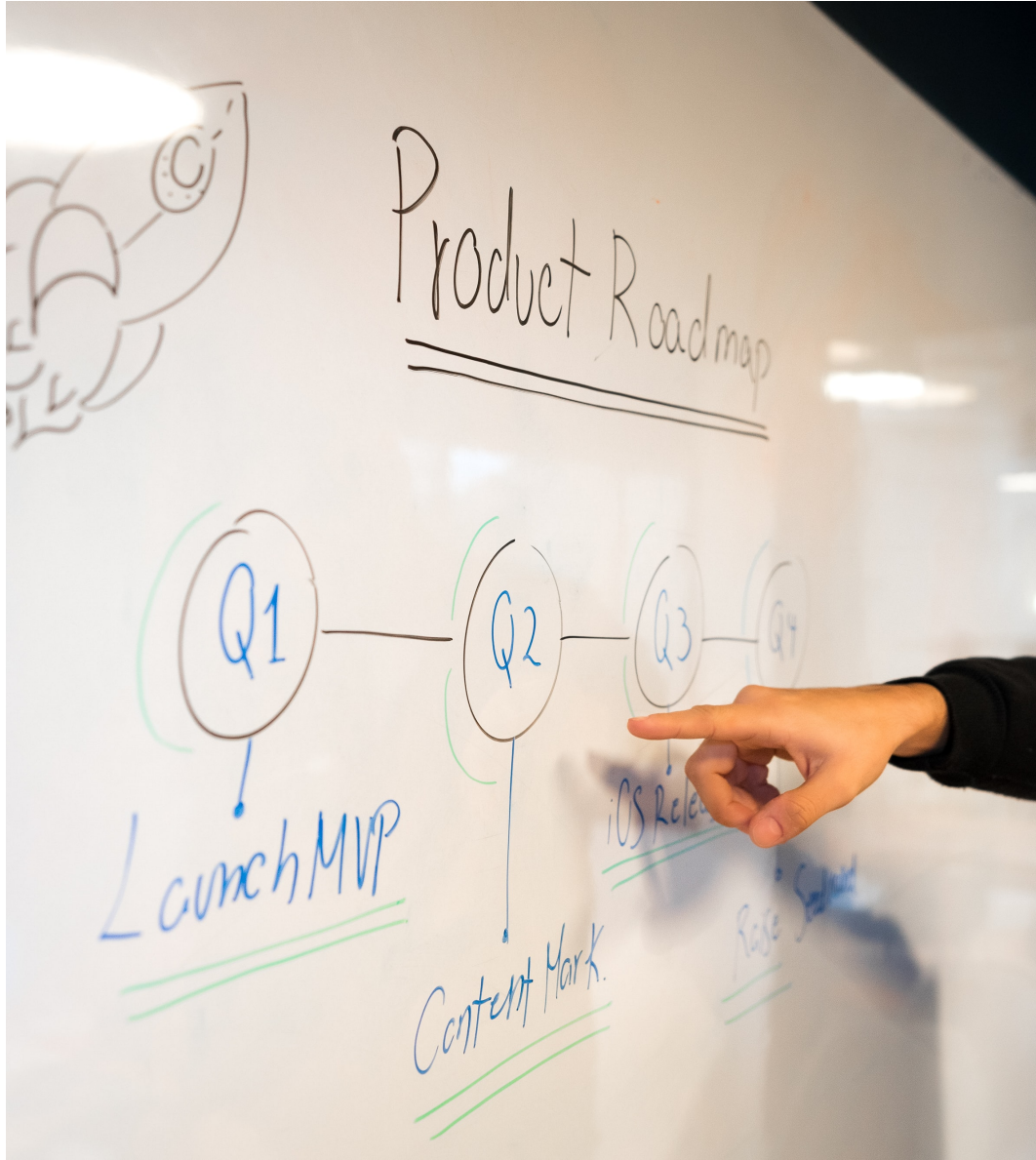
- We will build out this slide deck over time
- The overall work is now in 4 files: Parts 1-4
- We anticipate many more sections with these parts



# Contents

## 4 pages now - more to add

---



# Contents - Part 1

---

- Intro - slide 2
- Key Ideas - slide 11
- The simple case - slide 20
- Real-Work Scrum workshop - 23
- A Real Team - slide 27
- What is Scrum - slide 35
- Getting Started 1 - slide 46
- A Decent Team - slide 57
- Evaluating Fit - slide 65
- Good Team (cont'd) - slide 74
- Discussion, Questions - slide 82





## Contents - Part 2

---

- Intro - slide 2
- Getting Started 2 - slide 5
- Diagnosis In-Flight - slide 26
- Start Sprinting - slide 47
- Commit - slide 54
- Getting Started 3 - slide 64
- Sprint Zero? - slide 69
- Discussion, Questions - slide 79



# Contents - Part 3

---

- Intro - slide 2
- Pre-requisites to start? - slide 4
- Basics: Roles - slide 15
- Basics: Events - slide 27
- Basics: Artifacts - slide 35
- Requirements in Agile - slide 43
- Story Points - slide 54
- Business Value Points - slide 62
- Discussion, Questions - slide 68





# Contents - Part 4

---

- Intro - slide 2
- Team Too Small - slide 4
- Harder cases - slide 8
- Case: Not Software Development - slide 10
- Basic Problems - slide 14
- Case: The Team is “meh” - slide 23
- Case: Adding Complexity - slide 30
- Case: Scaling - slide 39
- Problem: Understand why? - slide 52
- Case: PO Sucks - slide 58
- Case : TBD - slide 64
- Questions - Discussion - slide 65



## Key Ideas





# Scrum is a bare framework

---

- Often simplified into:
  - 3 roles
  - 4 meetings
  - 5 artifacts
- Note: One could argue that the ideas are more important than these practicalities.



# Scrum is incomplete

---

- Definitely the Scrum Guide is incomplete
- Purposely, they tried to express the key things quickly. Which means, for example, that they will be (more) misunderstood
- You must ADD to it
  - And we always do, although often unconsciously or without thought





# Your situation is always unique

---

- Your company
- Your business/industry
- Your projects or products
- Your culture
- The demands and forces



# Your people are unique

- Your Team
- Your business stakeholders
- The “minions”
- Your managers
- Others
- ...and their mindset or culture
- Note: your people do already have part of the agile mindset or agile culture. You will also feel “yeah, but not enough”
- Still: each culture is unique too





# Everything is imperfect

---

- Really everything has an impediment (a Scrum word) — an opportunity for improvement
- You must start, even though everything is imperfect
- And: maybe things might be so bad — so that not starting is right — (seldom!)



## Hence - “adapt”

---

- Really everything has an impediment (a Scrum word) — an opportunity for improvement
- You must start, even though everything is imperfect
- And: maybe things might be so bad — so that not starting is right — (seldom!)
- So, let’s prioritize our improvements, and improve first in the most useful areas.
- (We’ll put the Top 20 improvements on the Impediment List.)





# No experts in your situation

---

- Yes, there are teachers and coaches and “people with a lot of experience”...
- BUT: No one is really that expert in implementing scrum
- No one is really good at getting people to change things
- SO: This is hard for you and hard for others.
- Experience: One is not so daunted. One knows it will work, probably, to a decent degree. One knows it is not “just me”. And one knows some “answers” that will be useful.



## But people can help

---

- We will help (eg, in this workshop)
- Other Scrum experts can help (books, blogs, videos, etc)
- Scrum practitioners in a similar situation can help
- Share patterns, try them, see if they help
- There is no **end** (final goal); you are always improving





## The Simple Case



## Situation - Why?

---

- Context is very important.
- Usually the basic principles that apply are seen more easily in a simple situation
- But not always...
- We can get to the “meat” sooner if we understand the situation, at least basically





# Simple Situation

---

- Preference to consider (and work with) a simpler situation (Keep as KISS as possible)
  - One Team
  - You have a full team (7 people)
  - No show-stopper impediments
  - Not a weird product
  - Other?



# Real-World Scrum Workshop





## RWS workshop: what

---

- Taking one day to “start”, to get organized, at least at the high level
- We recommend a workshop where everyone is there; a coach also.
- Team + BSHs + Others?
- Very adaptive



# High Level Agenda

---

- Set initial more detailed agenda
- Work through some items
- Revise agenda
- Work through more items
- Identify future work on the Impediment List (or elsewhere)







## Starting to Diagnose





## Often it is quite basic

---

- That is, diagnosis is not hard
- Ask THEM what's the biggest problem. From 7, usually you have at least one of the key "impediments"
- Often it's something that should have been done right at the get-go



# Kaizen Mindset

---

- What I think is key is establishing a Kaizen Mindset
- So, when you “join” the Team, start talking about that.
  - Nothing is perfect
  - Everything can be improved
  - Let’s start to identify the top thing
  - Let’s have someone working on the top ONE at all times
- Very simple, in a way





A REAL Team



# Do they want to be a Real Team?

---

- For now:
- They want to or feel they NEED to work together
- They want to become a great Team (as they define that). A better Team, in any case.
- They must choose. Some of it is “we can succeed together” and some is chemistry, I think.
- They may not choose immediately, but fairly soon...





## Think of it this way

---

- If you were the New York Jets, how would you build a winning Team in Sept 2023?
- OK, funny example in Sept 2023, if you know.
- But you get the idea. (Yes?)
- BUT: It is clear. Your goal is to build a better team, to win more.
  
- Are all the right people thinking this way?



## IDEO's idea - HOT Teams

---

First, they were totally dedicated to achieving the end result.

Second, they faced down a slightly ridiculous deadline.

Third, the group was irreverent and nonhierarchical.

Fourth, the team was well-rounded and respectful of its diversity.

Fifth, they worked in an open, eclectic space optimal for flexibility, group work, and brainstorming.

Finally, the group felt empowered to go get whatever else it needed.

Tom Kelly, GM of IDEO, 2001





## 5 Fundamentals

- Small number
- Complementary skills
  - Technical or functional expertise
  - Problem solving and decision-making
  - Interpersonal skills
- Common purpose, common set of specific performance goals
- Commonly agreed work approach
- Mutually accountable

“The Wisdom of Teams” - Katzenbach & Smith



# They set goals for the Team

---

- The Team is a thing
- The Team has goals
- We work together to achieve the goals





## The Product Goal - yes.

---

- I like this idea too. Aka Vision.
- Part of Scrum (see Scrum Guide)
- Highly variable, depends what you are doing.
- AND: I do think we also need a set of Team Goals





## Example: A Team's Goals

---

- Higher Happiness / Fun
- Double the Velocity in 6(?) months
- Fewer hours (~40 hrs)
- Higher quality
- More Learning
- Less Stress
- More business value (eg, higher BVP per SP)
- “I never want to leave this Team”



# What is Scrum?





# Scrum is...

---

- A way...
- A discipline
- A path towards some goal, maybe development of a product
- Not a silver bullet, not a panacea
- It helps the person on the path.
- But the person (team) must walk the path and face what may come





# A way of learning

---

- For knowledge workers, for the Team
- In 18 domains (of knowledge)
- Learning is fun
- We learn a lot by doing (trying) and by making mistakes



# A way to learn how to work together

---

- What is collaboration really?
- How many hours a week should we collaborate? As 2 people, as 3 people, as 5 people?
- How do we learn to collaborate better (per 25 minutes)
- Promiscuous pairing (you know some Boomer came up with that one)





# A way of Product Evolution

---

- What? I thought we could foresee every twist and turn of the Product?
- Ride the wave of change



# A way of adapting better to change

---

- There is going to be good and bad change
- How do we get the best result out of the Chaos?





# A way to decide and act faster

---

- We want less hesitation, or procrastination.
- We face imperfection
- We face incomplete knowledge
- We decide (hard!), quickly
- We do the best we can with what we have
- We win or we learn. (There is no try, and there is no failure, although we face those demons.)



## A way of learning who the customers are

---

- Of course there is a LOT we do not know about the customers
- Of course the customers' situation is changing (and we need to understand it better)
- Of course, the customers are changing (and we need to catch up with that)
- And then give them a better and better product





## A way of becoming our better selves

---

- A better Team (in every way, not just more damn widgets)
- You lose yourself and become more who you really are
- And the Team learns that they can do far more than they ever thought



# A way to build a Better Team

---

- Well, if the team is just BAD, then a way to discover that sooner, and give up
- More likely: A way to improve
- A way to build a great Team (at least by comparison to where we and our situation started)





# The road not taken

---

...

And both that morning equally lay  
In leaves no step had trodden black.  
Oh, I kept the first for another day!  
Yet knowing how way leads on to way,  
I doubted if I should ever come back.

I shall be telling this with a sigh  
Somewhere ages and ages hence:  
Two roads diverged in a wood, and I—  
I took the one less traveled by,  
And that has made all the difference.

Robert Frost. part of “The Road Not Taken”



## Getting Started - 1





# Getting Started

---

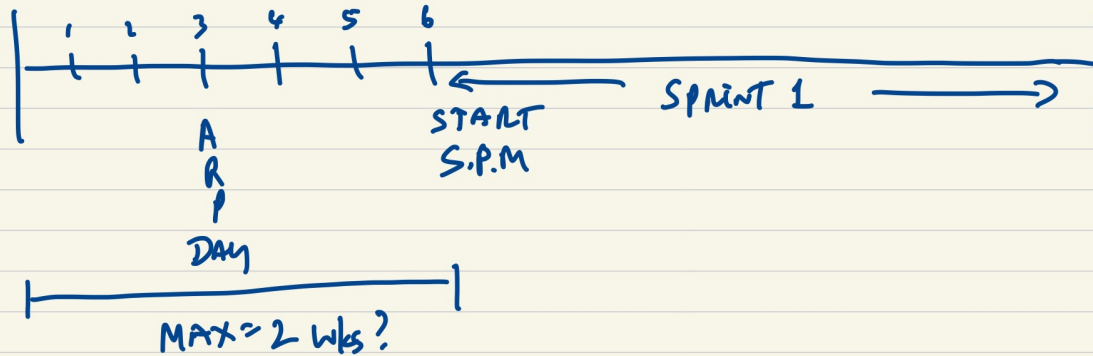
- This can happen many ways
- Notice that the Scrum Guide (the bare framework of Scrum) does NOT define this.
- BUT: We have a lot of experience, so let's share our advice for a "typical" situation
- Advice for an atypical situation might be a lot different



Let's put this in context

Get initiative  
↓

business days  
simple model





# Sprints give us Transparency

---

- Transparency enables: Inspect and Adapt.
- So, start sprinting ASAP, within reason.
- Sprints enable us to identify the biggest things to change better.
- Sprints enable us to make changes and see if they helped.



## Context

---

- We need (they need us) to start
- We learn from doing real work
- So: we must prioritize and do things in a time-box
- Still: It must be reasonable in your situation. Whatever that means. “Bend them but don’t break them.”

**Well begun is  
half done.**

Aristotle said that.



## 7 Simple things

---

- A top project or product
- A Team of 7 that is stable
- A decent team (that could be successful)
- They want to be successful, together
- We agree to be more honest & transparent
- We agree to improve continuously
- We get [Scrum] training, and Team agrees to try [Scrum+ or another way of working] for X months





## Simple: 7 Things, but...

---

- Can every Team get these things?
  - No, I think.
- Are some Teams in special situations where initial success is very difficult?
  - Less clear to me, but apparently yes.
  - Difficult for sure.
  - But therefore impossible?
- Should we at least ASK for these things?  
And point out when we don't have them?





# List 1

---

- Why start Sprinting ASAP? (xxx)
- Agree Team Goal (earlier) (31-33)
- Agree Product Goal [with ARP?]
- Evaluate Team skill sets (61)
- What does the Team need “to go”? (2-9)
- Business side commitment (2-10)
- Define Product Success [ARP?]
- Better automated testing (2-16)
- DOR/DOD (2-17)
- Agile Release Planning day (2-20)
- Commitment to Continuous Improvement (2-23)



# Missed?

---

- What about Sprint Zero?
- Infrastructure, Architecture, Design
- We agree: more cross-functional
- We agree: share knowledge across whole Team
- Others?





## Find an important project or product

---

- This comes first
- Yes, contrary to what you might think (find an easy one)
  - Easy and important often do not come together
- Improves motivation for all
- More likely to get help from the Organization
- Get everyone to understand the importance, value



## How big?

---

- In software, this would be at least one good release.
- Probably of some size. (Why?) By size, we mean a bunch of sprints, maybe 4-6
- ONLY one piece of work (one release, one product) - at least to work on first.
- Total: 6 months





**Getting Started - 1**  
**A decent Team**



## A Team of 7 that is stable

---

- Obvious, isn't it?
- PO, SM, Developers.
- 5 Developers probably means you have enough of the skill-sets inside the Team
- Stable is important. eg, You want them over Team to become a lot better. Only possible if stable.





# Get a good Team

---

- Scrum is about a real Team
- For now, let's keep it simple. Only ONE Team.
- How do we get a good Team?
  - You have to insist on having some influence, at least



## Get a good Team

---

- A good PO. Trained, wants the job, seems to have the skill and knowledge enough
- A good SM. Trained, wants the job. Who we at least think can drive impediments and help this Team climb (how high?)
- Developers include “builders” and “verification / validation” people (AKA QA).
- Trained in agile-scrum. Want to do it (try it).





# Attitude

---

- Managers have good intentions.
- Busy
- Know things we do not know
- And we know things they do not know
- Distract (and distracted)
- We need to remind them
- So, we must take initiative.



# Managers should evaluate the Team vs the Work

---

- Skill-set: Needed vs Have
- Evaluate
  - Do we have all the skill sets
  - Do we enough of each skill set
- Managers should identify any gaps.
- ...and show how they will address the gaps





# Team should evaluate Team vs the Work

---

- At beginning and later
- Evaluate
  - Do we have all the skill sets
  - Do we enough of each skill set
- Team should identify gaps
- Team can identify actions to fill the gaps



## So, are we good enough to start?

---

- Sometimes: Pretty clearly NO.
- Often: Well, maybe, not really sure yet
- Possible: We feel pretty good about our talent!
- Analyze.
- But decide early: yes, no, maybe, 60% confidence. And still be ready to react and adjust.





**Getting Started - 1**  
**Evaluating Fit - details**



# Managers should evaluate quickly - how?

---

- Skill-sets: List the Needed skill sets. Maybe 15? 20?
- If they want to include knowledge domains as skill sets, I'm ok. And personality traits.
- List the "Have" skill sets (per person). Quickly.
- 1 hour?
- Identify Gaps
- Include "how much" for each SS and the level ((1-5) per person). Have vs needed. Revise gaps.
- Identify how to address gaps





# Managers should evaluate?

---

- Honestly: I don't think they are professionals at this. Become better.
- Doing a good job takes time; BUT, doing it quickly also gives us useful results
- When do we really know? Well, after the Team has done [2?] sprints of work. Much clearer.



# Managers: how to address gaps?

---

Lots of answers, here's some:

- Course / workshop
- Video
- Books / articles
- From coach / expert (outside Team)
- From more experienced person in Team
- On-the-job (learn by doing)
- Chickens (part time helpers)
- Other?





## Team should evaluate - the same as for managers

---

- How they do this is about the same as for the managers.
- Why?
  - One: Team knows their skills (or lack) better
  - Two: Team often understands the work better.
  - Three: Team is usually less objective at identifying gaps, but anyway, different.
  - Four: Team will care in a different way. This is their baby.



## Team should evaluate - 2

- Skill-set: List the Needed skill sets. Maybe 15? 20?
- If they want to include knowledge domains as skill sets, I'm ok. And Personality traits.
- List the "Have" skill sets (per person). Quickly.
- 1 hour?
- Identify Gaps
- Include "how much" for each SS and level (1-5) per person. Have and needed. Revise gaps.
- Identify how to address gaps





## Team should evaluate - 3

---

- Honestly: Typically they have never done this before.
- But: If they are going to self-manage and be responsible, they **MUST** understand this.
- Doing a good job takes time; **BUT**, doing it quickly also gives us useful results
- Do it quickly up-front.
- Do it later also. When? After Sprint 2?
- When do we really know? After the Team has done [2?] sprints of work. Much clearer.



## Team should evaluate - 4

---

Lots of answers to fill gaps, here's some (same):

- Course / workshop
- Video
- Books / articles
- From coach / expert (outside Team)
- From more experienced person in Team
- On-the-job (learn by doing)
- Chickens (part time helpers)
- Other?





## Who acts on the Gaps?

---

- Someone must. (Or multiple people.)
- Can the SM do it alone?
- Who else?
- How do you divide the work?



**Getting Started - 1  
Good Team (cont'd)**





# They want to win, together

---

- Some fire in the belly
  - They want to be in a real Team
  - They want to be in this Team
  - They want to win, together
- 
- Can you determine this clearly up-front?  
And if there now, will it stick?



# Transparency

---

- They agree to be more transparent than before. Continually more transparent
  - I suck
  - We suck
  - They suck
  - It sucks
- And at least try to do it in a human way
- The organization agrees to be more transparent about itself





## Agree to Improve

---

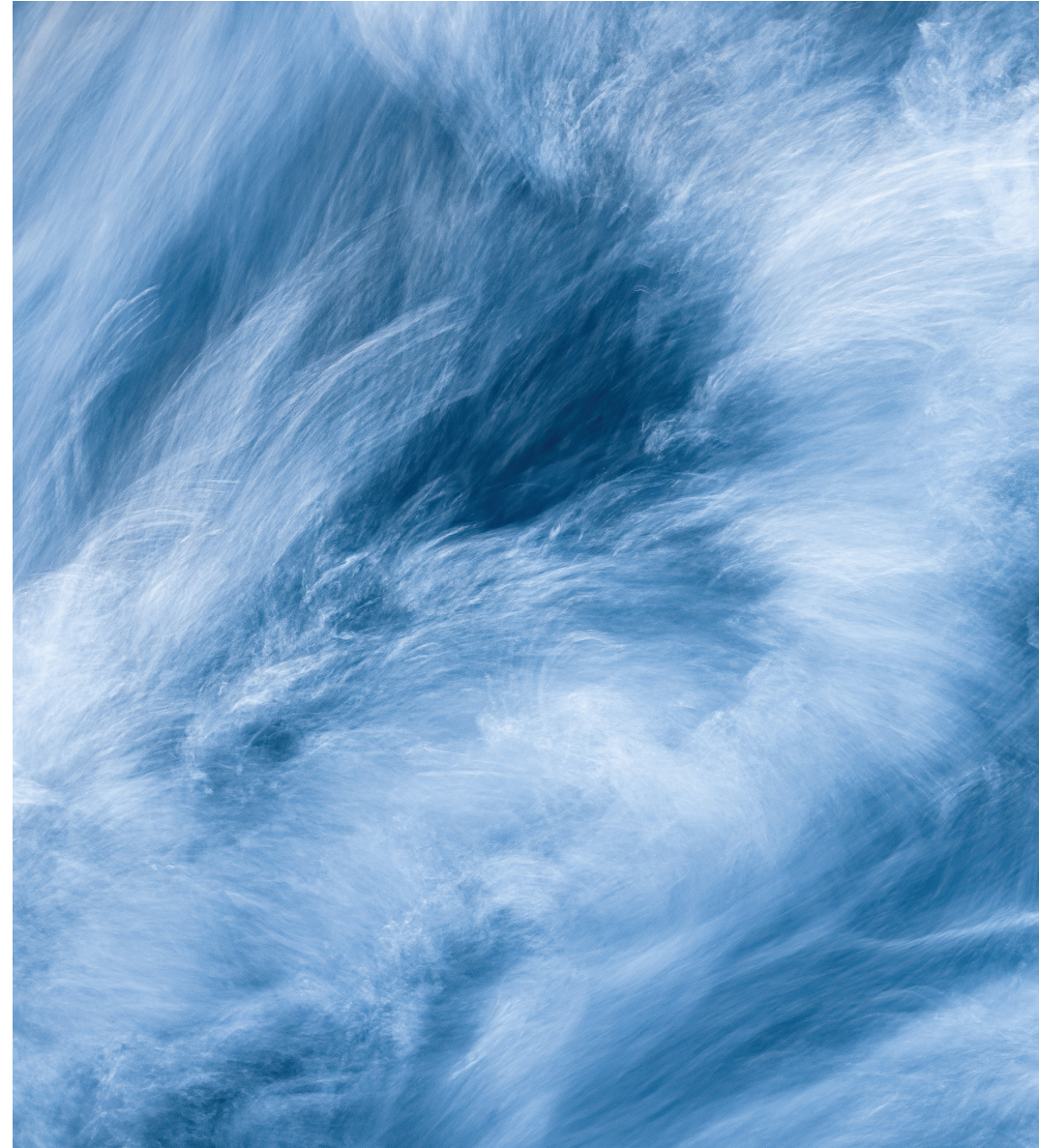
- Seem people just want to “do the work”.  
“We don’t have time to improve.”
- We agree to be changing things.
- How much change - let’s agree. But NOT zero change and NOT infinite change.
- Some people feel: You can start “scrum” at any level, as long as you agree to improve or at least try things to improve.
- How patiently do we wait for “real change” to happen?



## Issue: Speed of change

---

- Address the rate of change now?
- Or later?
  
- But some people will become a lot more comfortable if they understand what speed of change you are bringing.





## Get good Scrum training

---

- New(-ish) people: The CSM course together, recommended
- Experienced people: “Team Level Up workshop” is recommended (Level Set and agree to Level Up)



## Agree to **Try** Scrum

---

- They agree to try Scrum
- For a period of time (3 months?)
- They volunteer to “see if we can get it to work”
- Their choice / volunteering / experiment





## More than Scrum

---

- They need to agree to their whole “way of working” and Scrum is only a bare framework
- They should start to be more conscious and more explicit about the additional things they are adding to Scrum.
- And about the parts of Scrum they are not using, and why.
- The Team Level Up workshop addresses this well.



## Discussion, Questions





# Discussion, Questions

- Don't be shy!
- Turn off the mute
- OR: Type them in the Chat



## Other webinars

---

- Two Types:
  - About courses and workshops (30 mins)
  - About agile questions or issues (60 mins)
- Where:
  - [LeanAgileTraining.com](https://leanagiletraining.com)
  - [MeetUp](https://www.meetup.com)





## More Info about Us

---

- Courses & Workshops:  
[leanagiletraining.com/lean-agile-and-scrum-courses](https://leanagiletraining.com/lean-agile-and-scrum-courses)
- Joe Little
- [jhlittle@leanagiletraining.com](mailto:jhlittle@leanagiletraining.com)
- (704) 376-8881
- [leanagiletraining.com](https://leanagiletraining.com)

