# You're Doing Agile Right, but agile wrong

# Audience

- Looking to get more out of your Agile processes
- People feeling like they're doing the Agile things without the agile outcomes

scottsauber

# Poll

- How many people doing Scrum?
- SAFe?
- Kanban?
- Scrumban?
- Their own thing?
- Waterfall?

# Agenda

- What is agile?

- Where did it come from?

- Where did it all go wrong?

- Why are we still talking about agile in 2026?

- My opinions on:
  - Scrum
  - SAFe
  - Sprints
  - Estimating
  - Measuring what's working and what isn't
  - Various anti-patterns I see

scottsauber

# Goals

- Better understanding of what agile is and its roots
- Practical ideas to apply to your agile processes immediately

scottsauber

# Who am I?

- Director of Engineering at Lean TECHniques
- Microsoft MVP
- Dometrain Author
- Redgate Community Ambassador
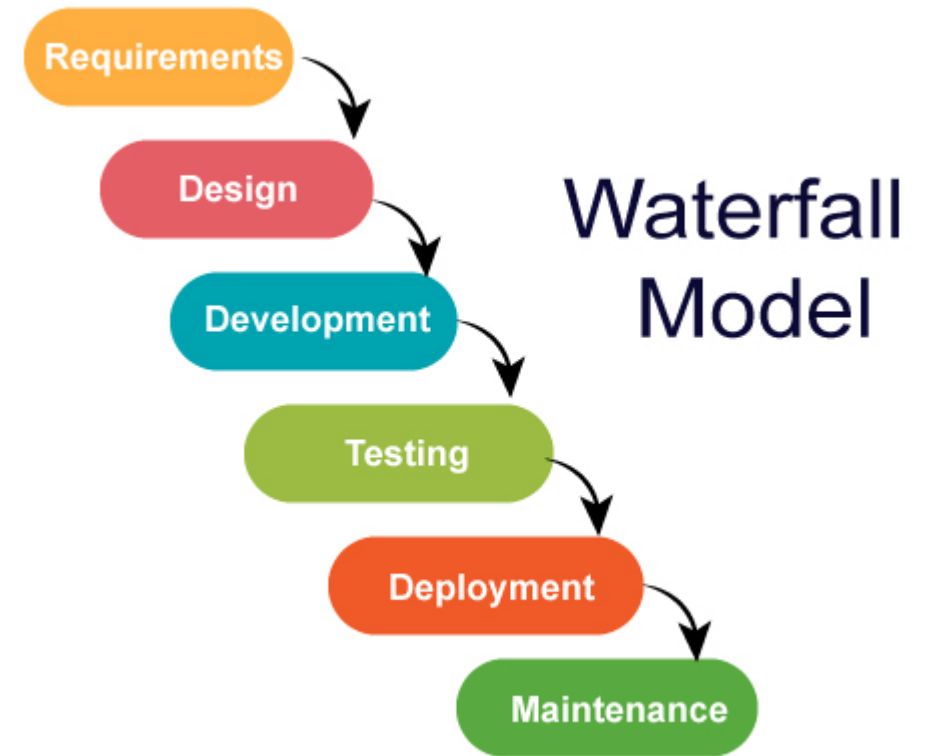- Co-organizer of Iowa .NET User Group


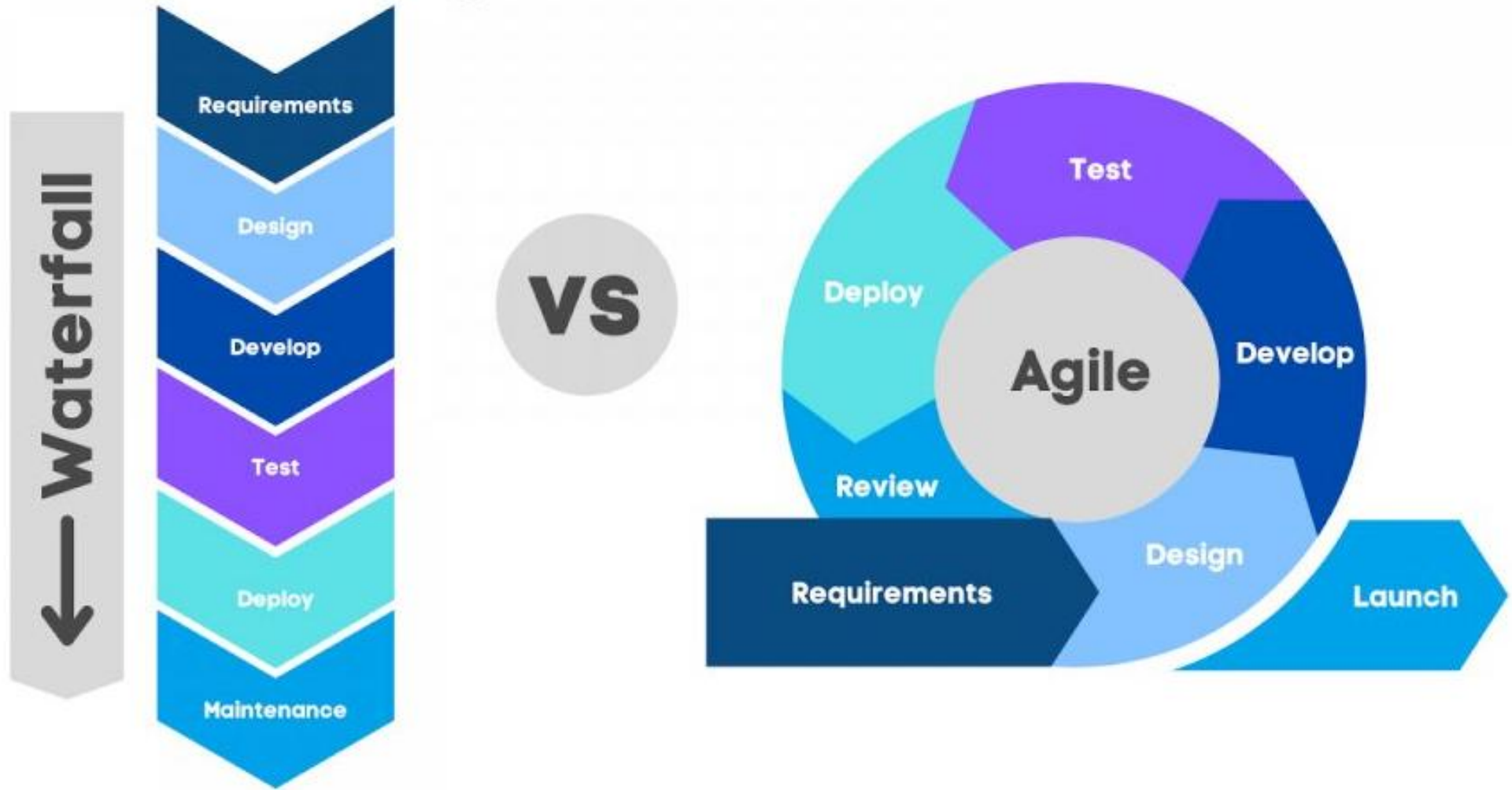
𝕏 @scottsauber    🦋 @scottsauber.com

# Before agile

- Waterfall
- Batch everything
- Do all work up front before next steps
- Problem is software changes
- People and companies change their mind, environments change, etc
- Changing anything blows up the entire plan
- Projects would take years without delivering value



scottsauber

# What is agile?

- The ability to respond to change quickly and safely

- Software is uncertain, let's make small bets instead of large ones

- A set of principles to guide to the ultimate goal of satisfying the customer through early and continuous delivery of valuable software

scottsauber

# Agile vs Waterfall

**Waterfall**

- Requirements
- Design
- Develop
- Test
- Deploy
- Maintenance

**VS**

**Agile**

- Test
- Develop
- Design
- Launch
- Requirements
- Review
- Deploy

scottsauber

# Agile Manifesto

- 2001 a group of 17 individuals met to talk in Snowbird, Utah
- Came up with a set of principles to guide to the ultimate goal of satisfying the customer through early and continuous delivery of valuable software
- https://agilemanifesto.org

scottsauber
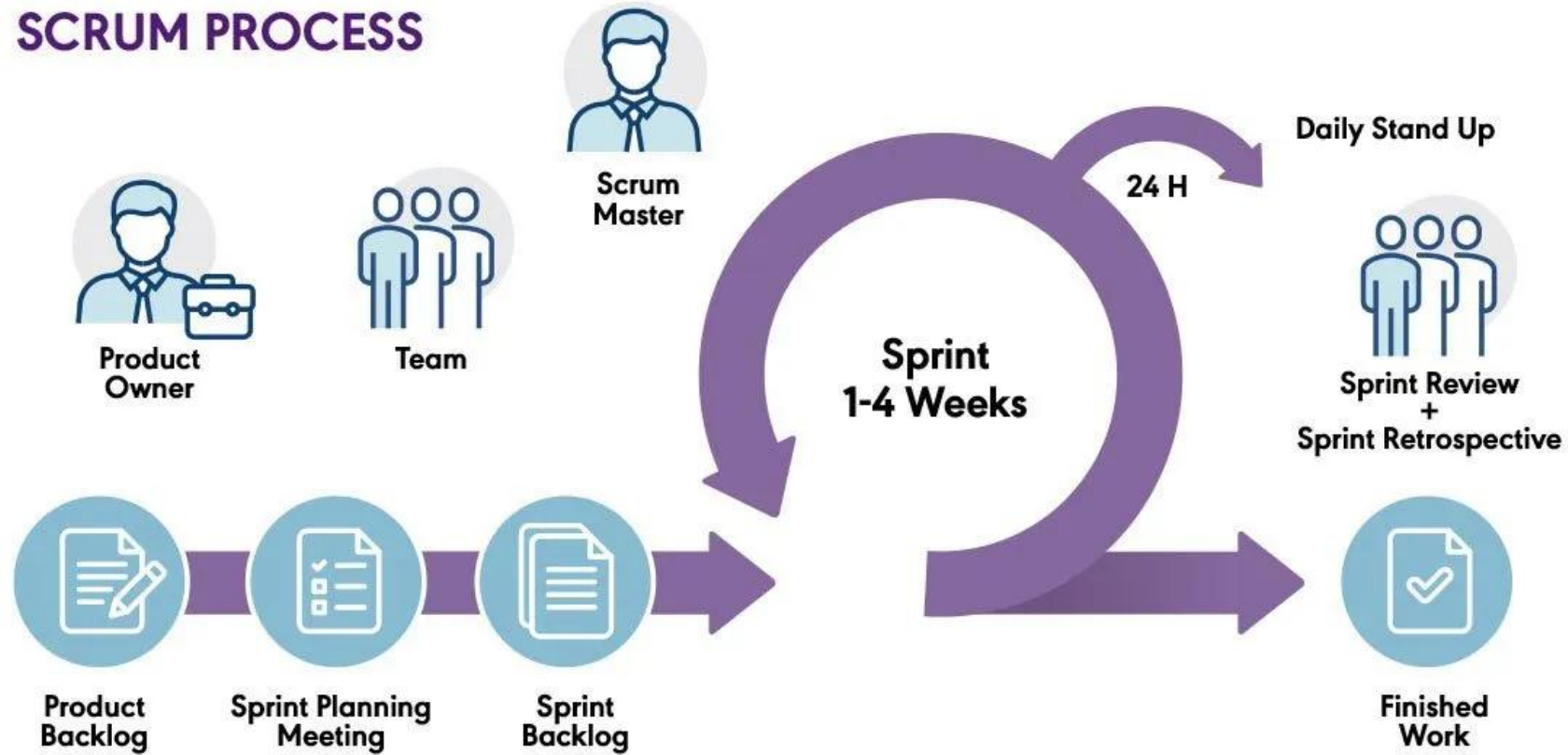
That's it.

Or it should've been...

# Framework Review

# Frameworks emerged

- Scrum
- Kanban
- Scrumban
- SAFe

# Scrum at a high level



**SCRUM PROCESS**

Product Owner

Team

Scrum Master

Sprint 1-4 Weeks

24 H

Daily Stand Up

Sprint Review + Sprint Retrospective

Product Backlog

Sprint Planning Meeting

Sprint Backlog

Finished Work

scottsauber

# Story Points Refresher

- Created by Ron Jeffries

- Unit of measure to represent effort and risk

- Usually Fibonacci sequenced (1, 2, 3, 5, 8, 13, etc)

- Usually estimated during Planning Poker done during Refinement

- Can't be compared across teams reliably (plz don't try)

- Ron Jeffries in 2019: "Well, if I did invent story points, I'm probably a little sorry now... If they're not providing great value to your team or company, I'd advise dropping them on the grounds that they are waste. If, on the other hand, you just love them, well, carry on!"

scottsauber

# Kanban at a high level

- No Sprints

- Visualize work

- Organize work by priority

- Pull the next most important thing off the backlog
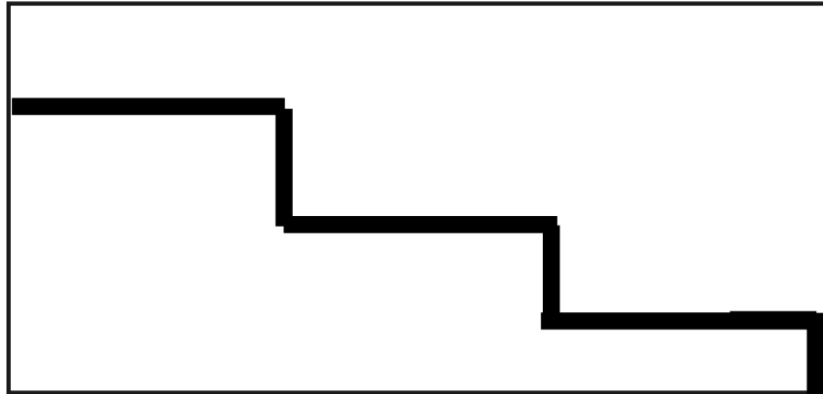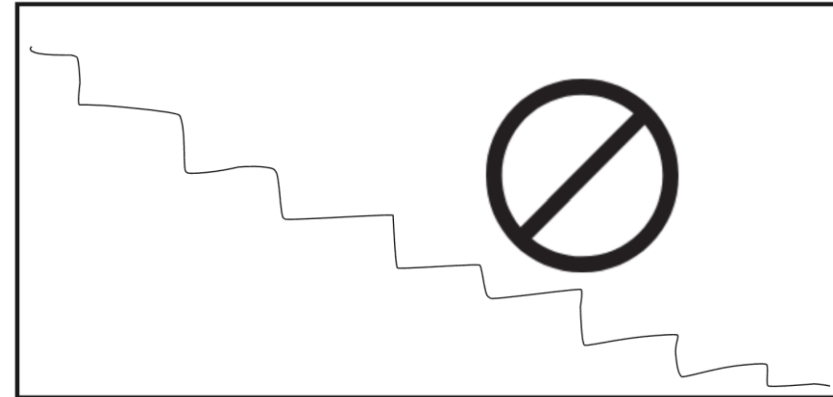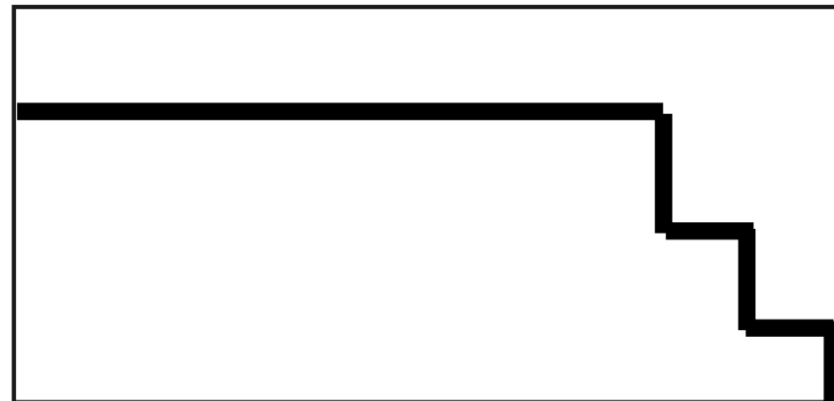
- Optimize flow and limit WIP

scottsauber

WIP = 1

A  B  C

WIP = 3

A B C A B C A B C

BURNDOWN CHART

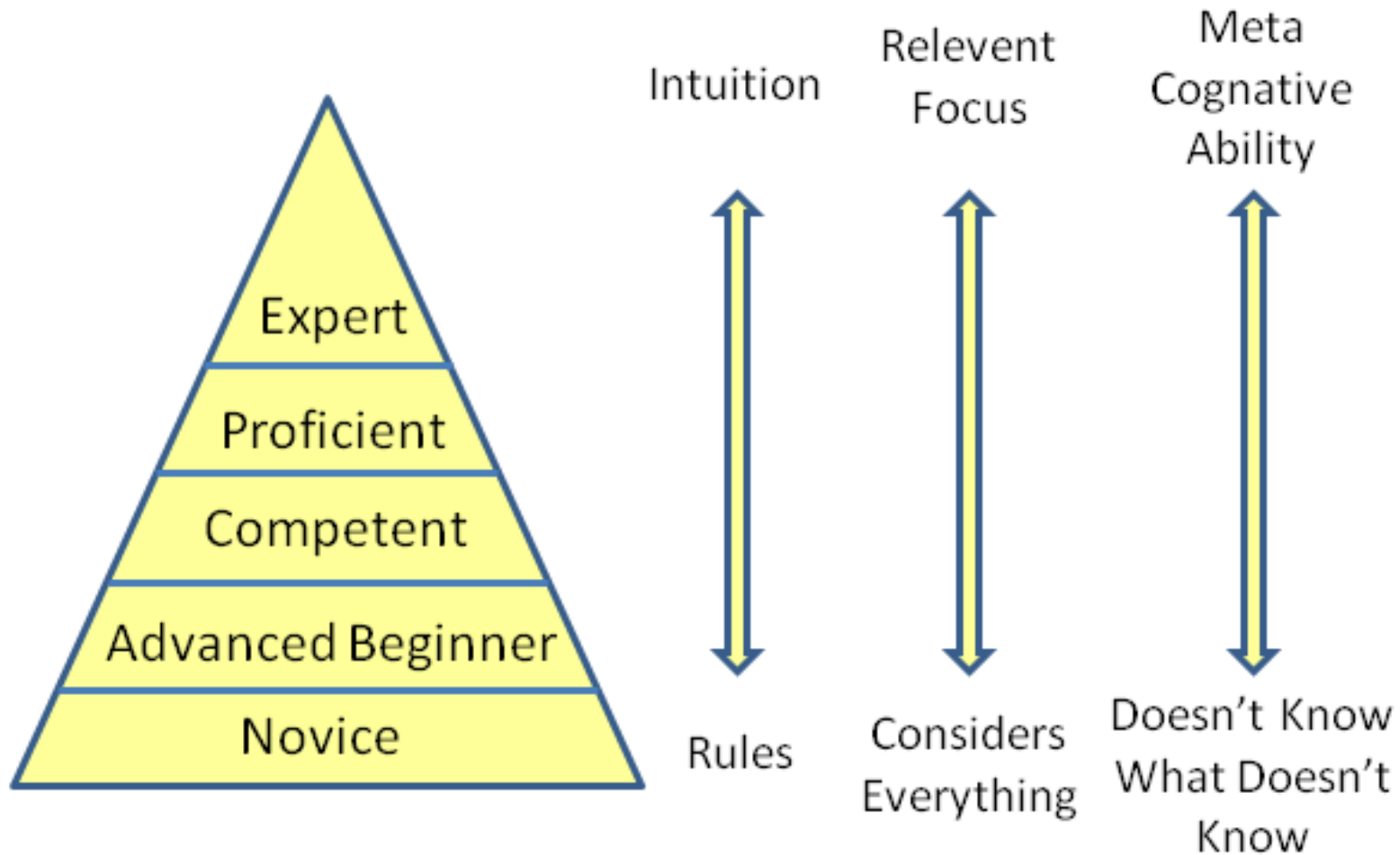BURNDOWN CHART

BURNDOWN CHART

scottsauber

# Scrumban at a high level

- Use Scrum for planning/alignment, Kanban for execution/flow
- Planning/Refinement
- Daily Standups
- Sprint Review/Demos
- Retrospective
- Sprint goes away

scottsauber

# Frameworks

- Meant to encourage agility
- Go through the motions of the same old process
- Forget to think about why
- Frameworks are useful as a starting point if you don't know what else to do

scottsauber

# Dreyfus Model



Expert
Proficient
Competent
Advanced Beginner
Novice

Intuition

Relevent
Focus

Meta
Cognative
Ability

Rules

Considers
Everything

Doesn't Know
What Doesn't
Know

scottsauber

# That's Agile, what about agile?

# What is agile?

- The ability to respond to change quickly and safely
- A set of principles to guide to the ultimate goal of satisfying the customer through early and continuous delivery of valuable software
- https://agilemanifesto.org

scottsauber

# Story time



scottsauber

# What Agile traps do organizations fall into?

# Problem with Sprints

# Problems with Sprints

- Sprints are a good first step if you're doing waterfall
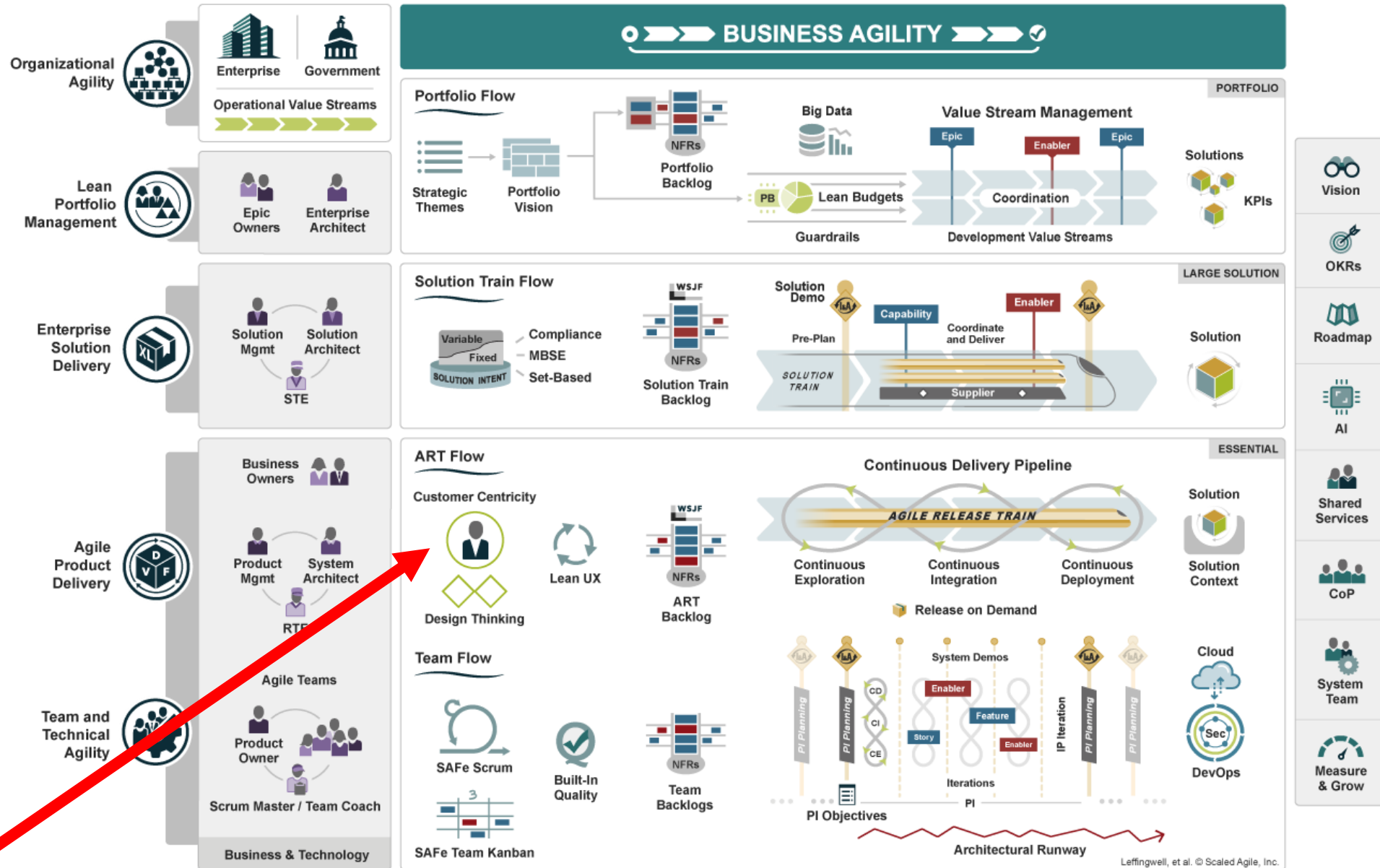- The best teams graduate out of sprints, in my experience

scottsauber

# Problems with Sprints

- Commit to work 1-2 weeks ahead of time
- Inevitably, new work comes up messing up your plans
- Not completing committed work drags the team down
- 18 points in refinement but 20 points of capacity… what happens?
- Working on a 5 point story and it rolls into next sprint… what happens?
- "I can't address that issue right now, I need to merge this PR because the sprint ends soon. I'll get to it later."
- "We're done 1 day early with the sprint… I don't want to pull in anything new"

scottsauber

# Stop doing Sprints

- Sprints are an arbitrary and make you do "weird" things

- Instead, organize work in priority order

- Do Refinement, Retros, etc every 1-2 weeks

- Lots of conversations "vanish"

scottsauber

# Problems with SAFe
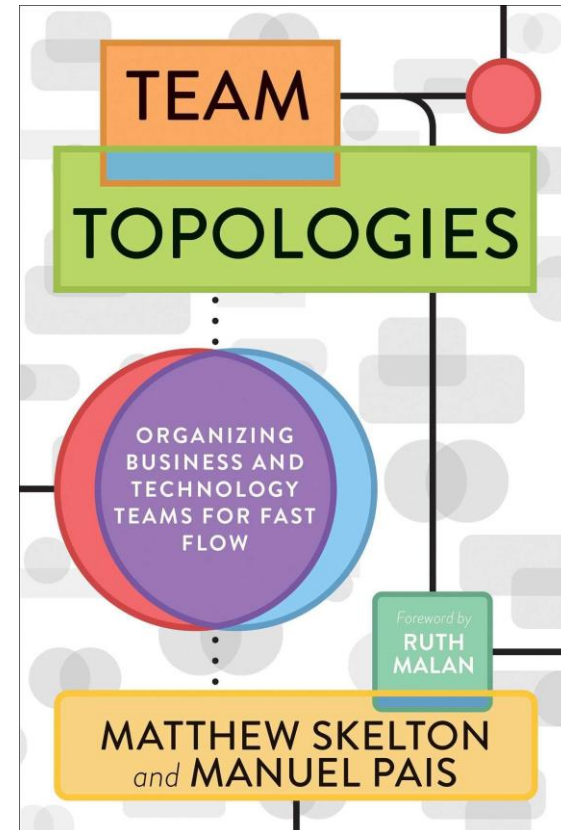
The SAFe 6.0 Big Picture

# Stop doing SAFe

- SAFe optimizes for certainty

- Antithesis of agile which optimizes for change

- SAFe does not optimize for change, autonomy, removing dependencies (manages them instead), etc.

- SAFe *can* be effective as a transitional framework, not a destination

- [Chet Hendrickson](): "(SAFe) optimizes for risk averse people, I'm not optimizing for security and safety, I'm optimizing to be able to react quickly to change. SAFe can work well for alignment, if you're going the right place, but if the world is changing..."

scottsauber

# Instead…

- Conway's law says our software mirrors our org structure
- Management are the real architects
- Change the org structure
- Align teams to value streams and remove dependencies
- Scales via architecture and boundaries, not meetings
- Reduces coordination load, not manage it harder
- BTW - you shouldn't have frontend and backend teams
- Team Topologies book for more



TEAM TOPOLOGIES

ORGANIZING BUSINESS AND TECHNOLOGY TEAMS FOR FAST FLOW

Foreword by RUTH MALAN

MATTHEW SKELTON and MANUEL PAIS

scottsauber

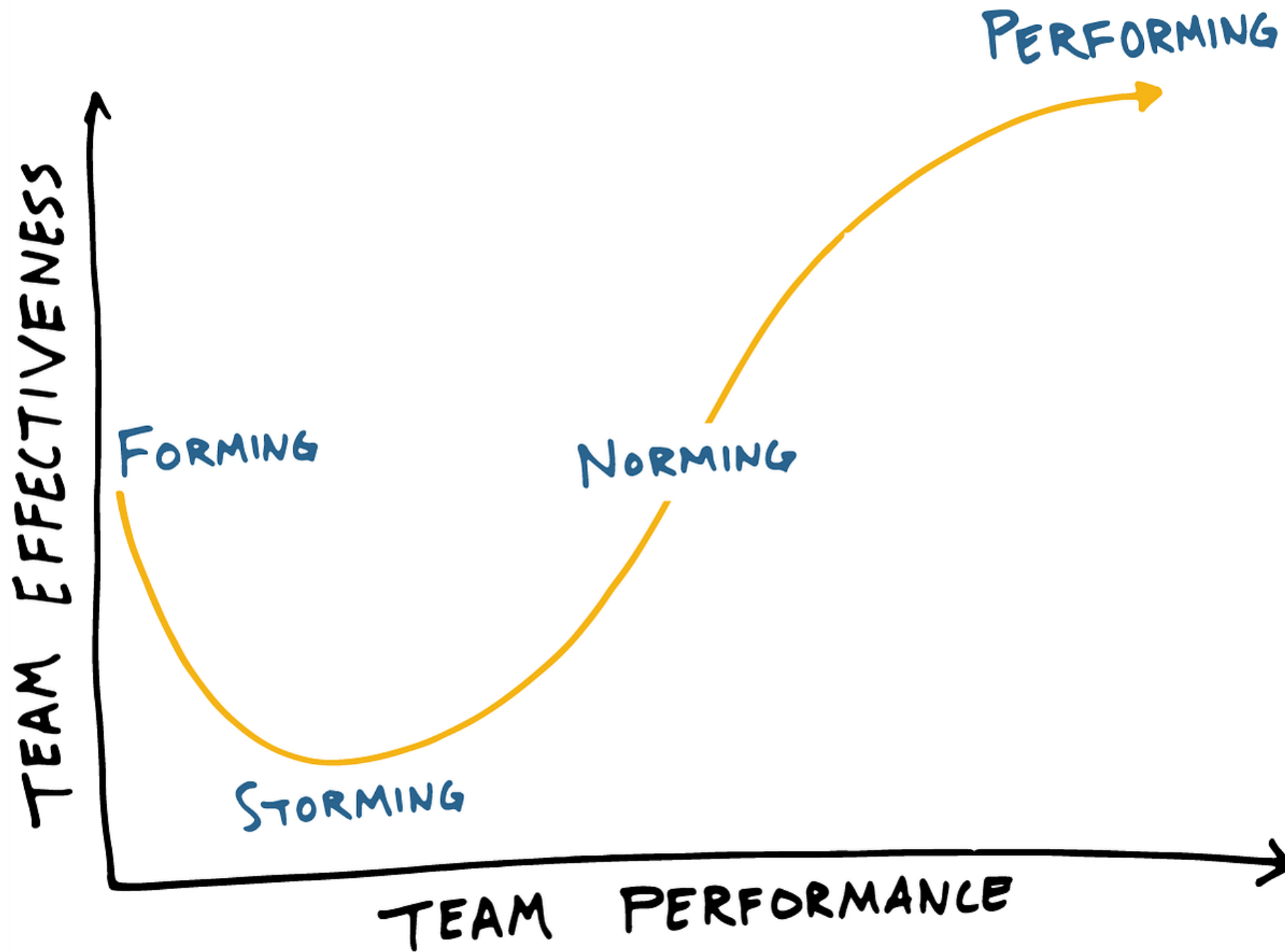# Problems with Story Points

# Problems with Story Points

- Meant to remove time from equation – in practice… "1 point is like 1 day, right?"

- Velocity becomes performance metric – bloat story point estimating

- Management uses it to compare teams – DON'T DO THIS

- Estimation meetings become expensive – way less valuable if team has durability.

- "How many points we delivered" isn't relevant – "How much value we delivered" is what really matters

scottsauber

# Do "No Estimates" instead

- Break down work into relatively the same size

- Stories should still add value when completed – don't break down too much

- Still can understand velocity and prediction – ie 20 stories completed a week

- At a minimum, if using story points, never have something bigger than a 5 – break it down

- Downside – if it's a new team, sometimes Planning Poker Discussions can be valuable, but Pairing overcomes this

- "Is the value of estimating higher than the cost of estimating?"

scottsauber

# Random tips

- If you're a shared services team that intakes support tickets – don't do Scrum, do Kanban

- If you're a new team, start with weekly ceremonies, not bi-weekly.

- Graduate to bi-weekly as you start Norming and Performing

scottsauber

# How do I know if I'm doing agile right?

# What was the point of agile again?

- The ability to respond to change quickly and safely
- A set of principles to guide to the ultimate goal of satisfying the customer through early and continuous delivery of valuable software
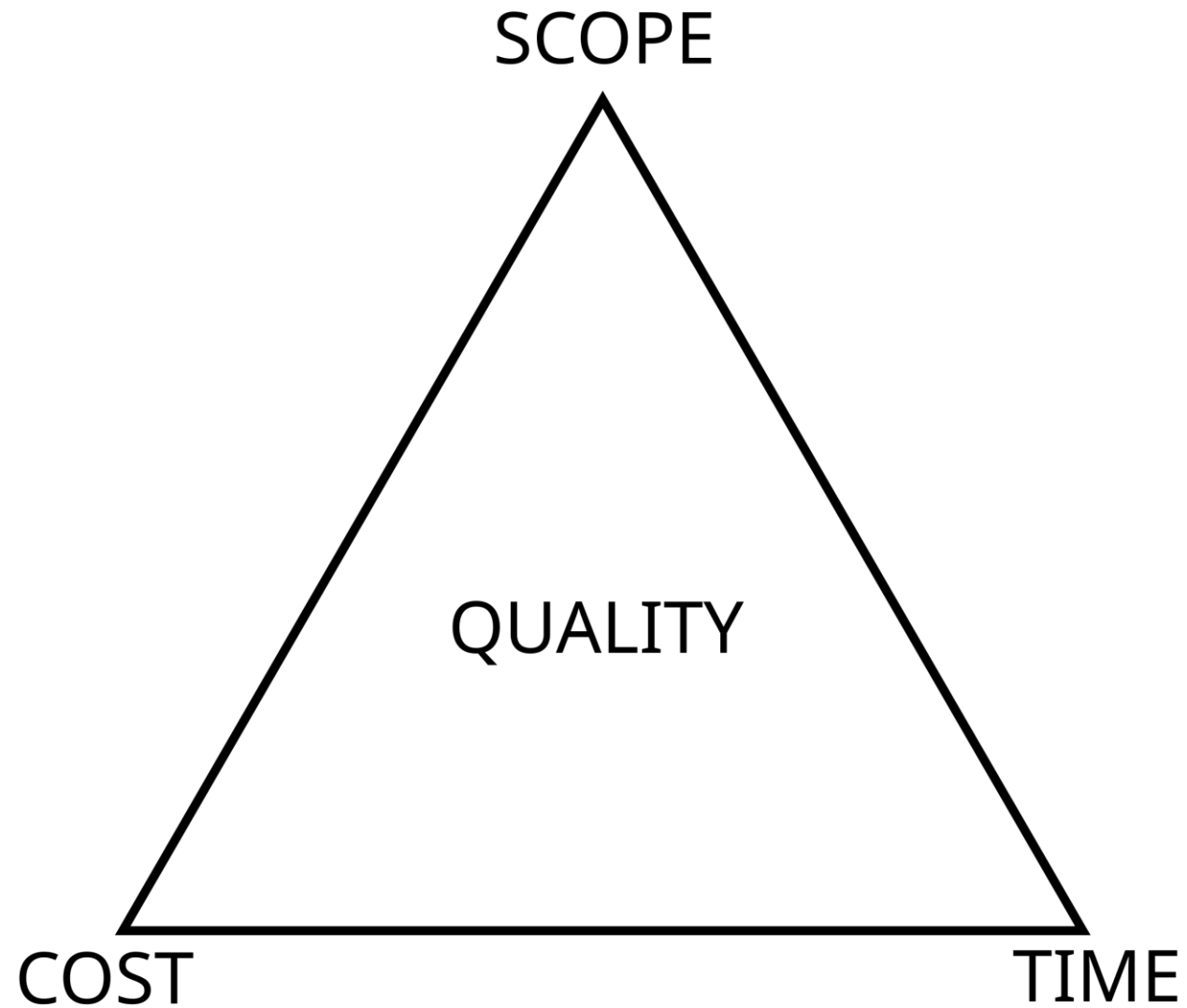
scottsauber

# Build It Right with DORA

- DevOps Research Assessment interviews 10s of 1000s of teams
- Found it came down to 4 metrics and 34 capabilities
- https://dora.dev
- Also: SPACE and tools like DX, but start with DORA
- DORA's metrics prove that you can respond to change quickly and safely
- The Quick Check compares you against your peers
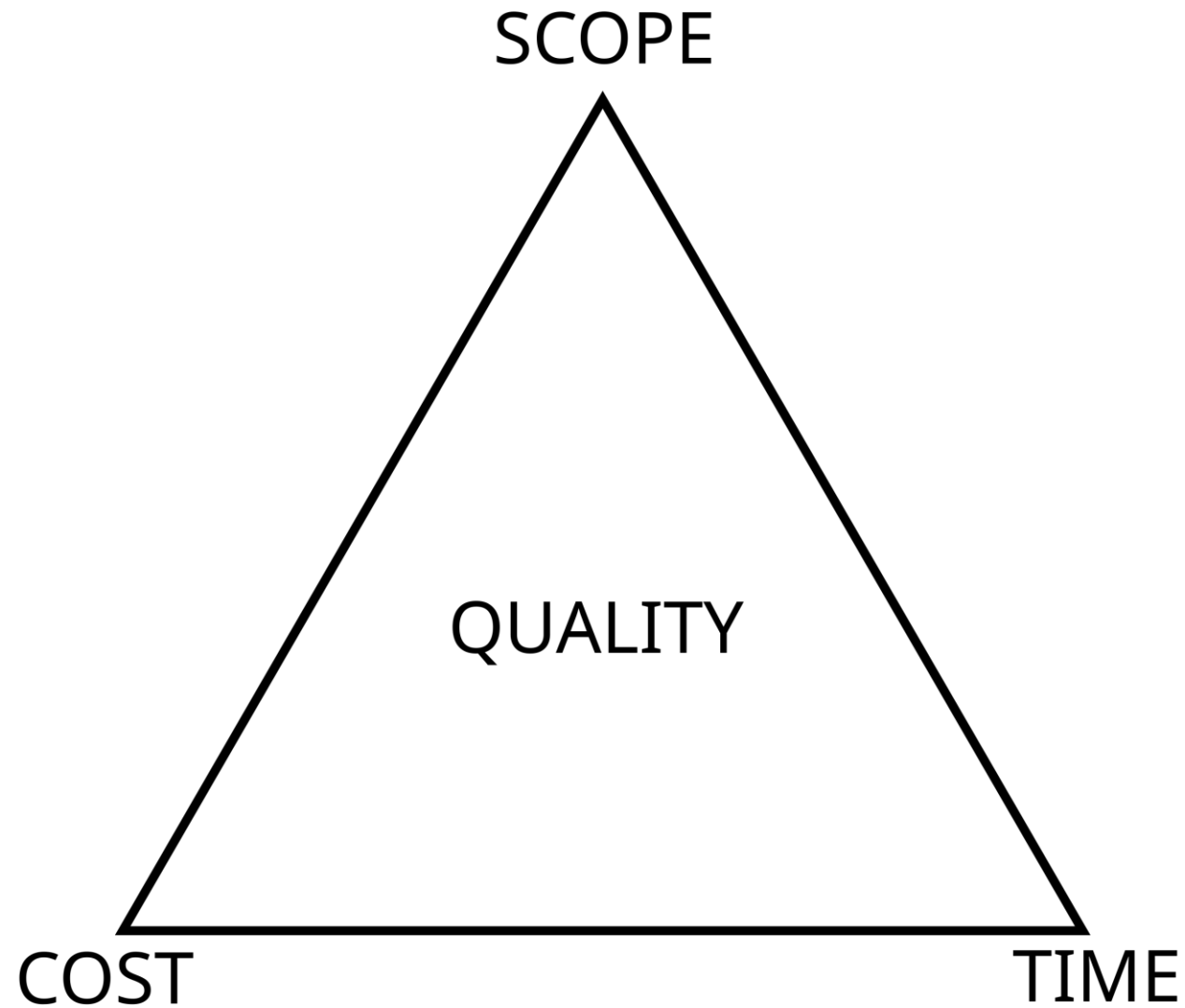
scottsauber

# Build The Right Thing

- But it doesn't matter if we can respond to change, if we're responding to the wrong change

- Too often we measure Efficiency and not Effectiveness

- Define success criteria for a story, retro back on it

- Define North Star metrics and track them

- Usage is always a good metric

- Jeff Patton video on [Output vs Outcome vs Impact](#)

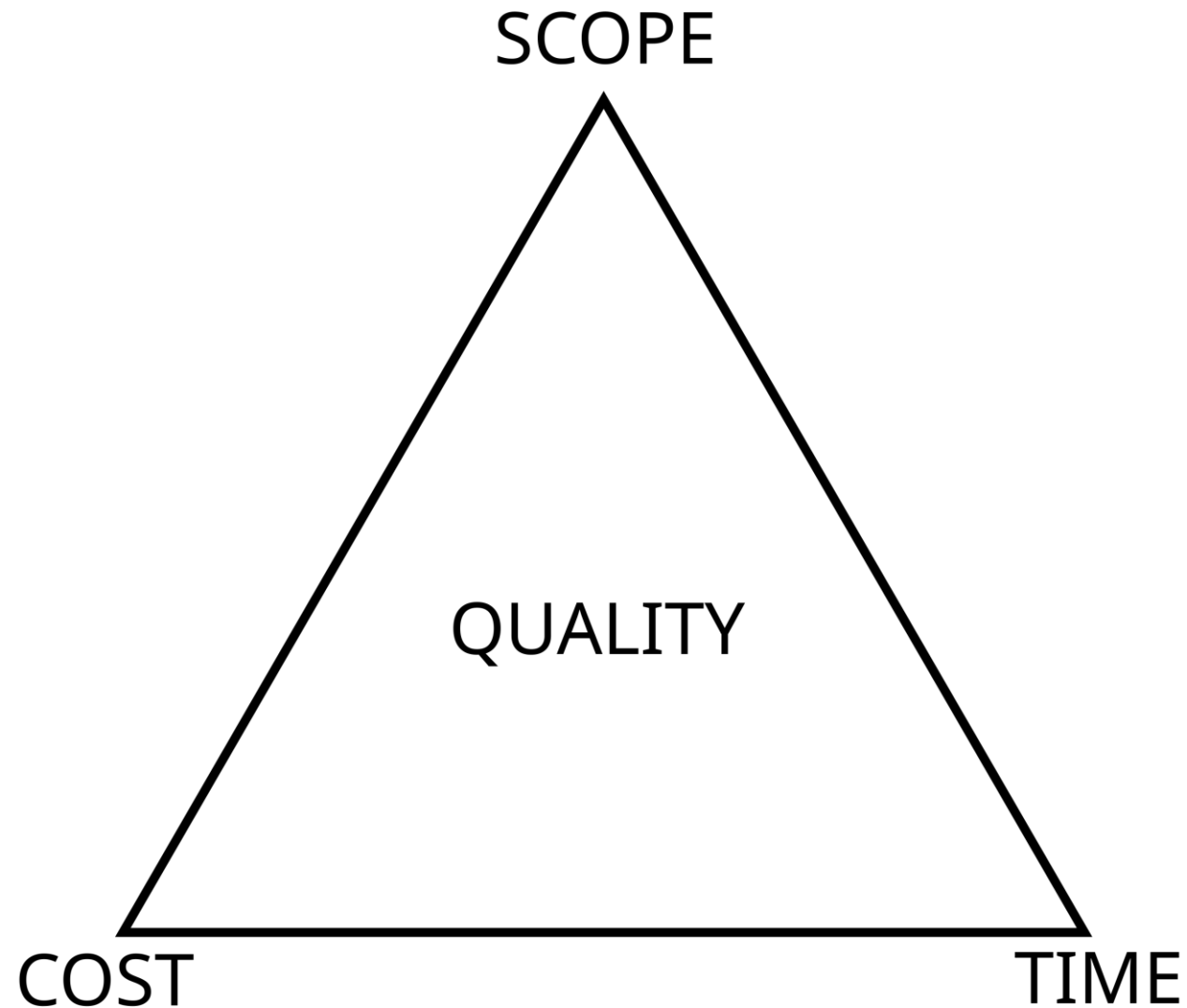- My number 1 beef with Scrum is there's no ceremony that emphasizes "did what we build actually matter"

scottsauber

# The Iron Triangle



SCOPE

QUALITY

COST                    TIME

scottsauber

# What's the problem?



SCOPE

QUALITY

COST          TIME

scottsauber

# It doesn't measure Outcomes and Impact



SCOPE

QUALITY

COST

TIME

scottsauber

# Takeaways

- Refresher on what agile is supposed to be about
- Ideas on how to improve your agile processes

scottsauber

# Resources

- This slide deck at https://scottsauber.com

scottsauber

# Questions?

ssauber@leantechniques.com

𝕏  @scottsauber

🦋  @scottsauber.com

Add me on LinkedIn:

Slides at scottsauber.com

# Thanks!

Slides at scottsauber.com