

Webinar: Velocity and Story Points Ver 1.

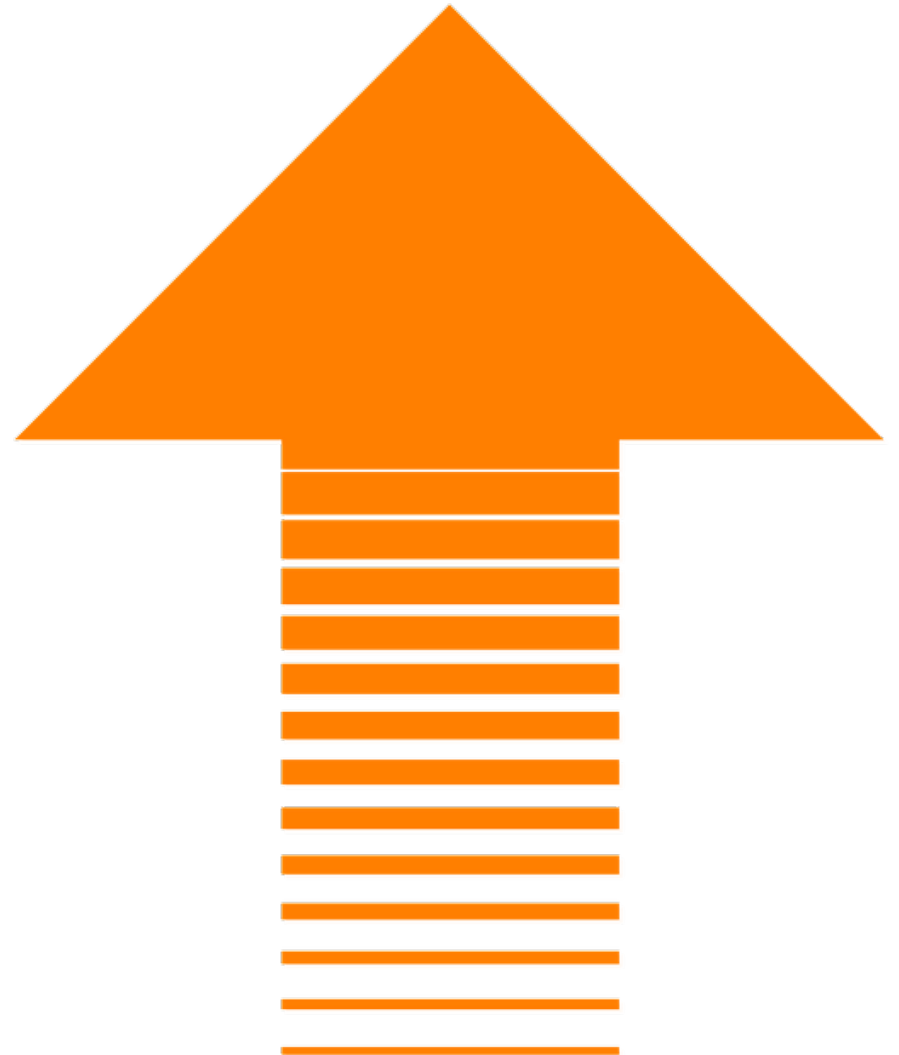
[LeanAgileTraining.com](https://leanagiletraining.com)

July 31, 2024

© 2024 Joseph Little

Scope of this discussion

1. Why?
2. Scrum is a Game
3. What is velocity? Why?
4. What are Story Points? How do we use them?
5. Planning Poker is a Game.
6. Winning & Losing.
7. Best we can cover in 1 hour.



First: A Game

1. Scrum is a game.
2. So important, so often forgotten.
3. If you have a game, you have to keep score.
4. Keeping score helps many ways, but of course you have to be reasonable about it.
5. And you have to help the Team win. Mostly win. (51%?)



Issues, we have issues!

1. Team members are uncomfortable with Estimating
2. Estimates for the Sprint can be notably inaccurate
3. Some people hate Story Points
4. Some people hate Velocity
5. Some people have misused Velocity and Story Points
6. Over-Stress (no fun)
7. Unaware of Learning!



Key Ideas

1. The Scrum Team wins (or loses) together
2. Winning is good! (“We are the Champions!”)
3. Losing is Good! (“You live, you learn!” Alanis.)
4. People are engaged more with Games.
5. Learning is important for Knowledge Workers!



What is Velocity

4.Sprint 1= 18 SP

5.Sprint 2= 22 SP

6.Sprint 3= 20SP

7.AVERAGE = 20SP

8.The average is the (average) Velocity of the Team at the end of Sprint 3.



So what? Why?

- Scrum is a Game (see cover of Scrum Guide 2020).
- Each Sprint is a Game.
- We set a goal for the Game: the number of SPs the team expects to complete.
- Then we measure how we did!
- Then: we win! (Yippee!) or we learn! (Great!)



Velocity enables the benefits of the Game

- If we win, it builds the Team. (We sing “We are the Champions”)
- If we don’t win, then we have an incentive to improve, so we can win next time.
- We get transparency that we need to improve.
- And, we only have to look at the elements in the “losing” Sprint to see what to improve on.



How do we use it?

- To define our expected capacity (to decide how much to commit to)
- We start with the Velocity number
- We (the Team) can adjust it, up or down, based on fair reasons.
- And then we commit (eg, to 20 SP of work). (Committing is more than just the SPs.)



Why might we adjust from the Velocity?

- People are OOO more (or less)
- An impediment was fixed
- We will collaborate more or better
- The PO will answer questions faster or better (or vice versa)
- The SM will get blockers fixed faster or better
- Etc



The Team decides how much to take in the Sprint

1. The Team accepts stories
2. The Team insists that stories have “good enough” details
3. The Team adds up all the SPs, and compared with expected capacity
4. The Team looks at the sum of the hours on the tasks, and compared to expected capacity
5. The Team commits.
Ex: 20SP = 20 SP.



It's a Game

1. We plan and set a Goal (focus now: the 20SP)
2. We work and measure, and see if we can knock in enough runs (ie, SPs)
3. We get a small win! Celebrate.
4. Or we learn. Have an incentive to improve. Become a bit less imperfect. Like every ball club.
5. Recommend: Try to win 50-60% of games / Sprints.



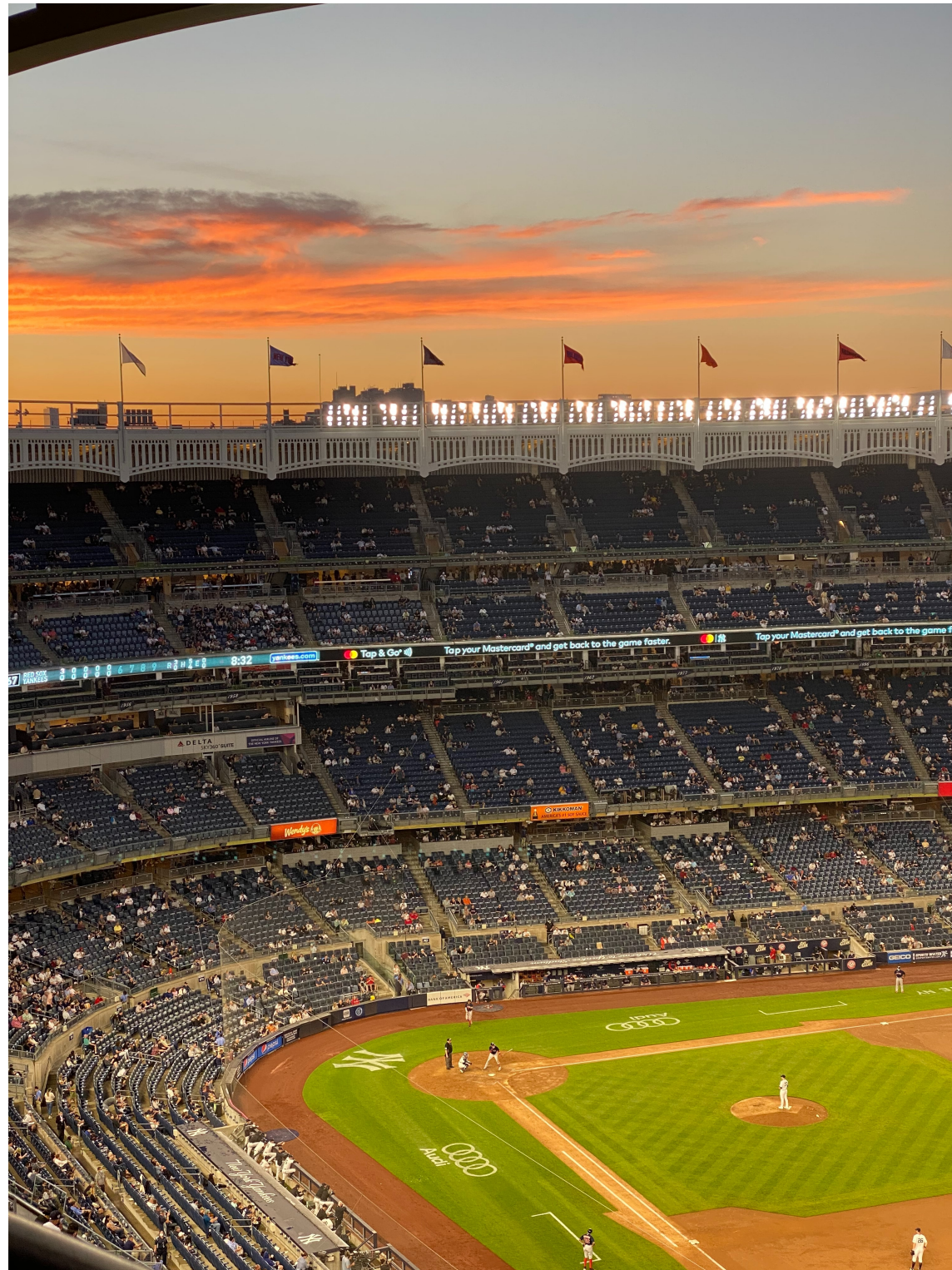
Deviation

- Probs improve if we have 8+ small stories
- We need to identify and mitigate things that cause large Velocity deviations. Ex: over-committing, weak estimating...
- The Developers get to estimate and commit.
- The visibility is better and the climb starts in a better way



Scrum is fun!

1. Commit so you can win
2. Work smarter, not harder
3. Build high motivation (also: calling PO!)
4. Celebrate small wins
5. Enjoy game of Planning Poker
6. Use planning process to attain eu-Stress



About Estimating Stories

How to do Story Pointing

1. The Developers (coders, testers, etc) vote
2. Reference Story = 1 SP
3. **Wide-band Delphi expert estimating**
4. Use Fibonacci cards
5. Often several rounds (3 or 4)
6. Within 3 consecutive Fibonacci cards
7. THEN: Average (to nearest integer)



Example

1. Three voters
2. Votes are 5-3-2. Three consecutive Fibonacci numbers.
3. $5+3+2 = 10$. $10 / 3$ voters = 3.3 SP
4. But we round to 3SP.



Breaking Up is good to do

1. Learn 20 Ways to break up stories (xp123.com). Practice.
2. Feels hard at first. Becomes easy.
3. Estimates are better for small stories
4. Helps that all the stories in a sprint are small, about the same size.
5. Separate win: 80-20 Rule!!!



More Tips

- You never have complete knowledge
- So, estimate the best you can with the knowledge you have
- Coders and testers both estimate for the total “work” by the Team. (Yes, a learning process)
- Estimating and reviewing helps identify knowledge you want.



Planning Poker Game

1. Making it makes it fun.
2. Fun brings engagement
3. Engagement means more learning, more sharing, more retention of knowledge
4. Everyone brings something to the party
5. Small “mistakes” are forgotten (especially if we win often)



Key Win

- The knowledge the Team gains helps estimating and planning.
- And, also, helps build a better product



Human Psychology

I hope you see how this all builds on some basics of how people work



And LEARN

- 1.Re-estimate multiple times
- 2.Re-estimate a given story when you gain new knowledge (more detail, other info)
- 3.Developers get to have “all” the information they need, eventually
- 4.Voting (and doing) helps them learn what info they need most
- 5.“The Season is more important than one game”
- 6.LESS pressure. They do not know at first what the Velocity will be



Two wins from Planning Poker

1. **The number on the card gives visibility to what our estimate is.**
2. This visibility allows us to “prioritize our stupidity”, and learn.
3. We learn about each other (eg, within the Developers). This will help.
4. **We share the most useful knowledge quickly.** In about 60-75 mins. Awesome!

“Everyone knows something.”

“Everyone knows what we all know.”



Eternal problems

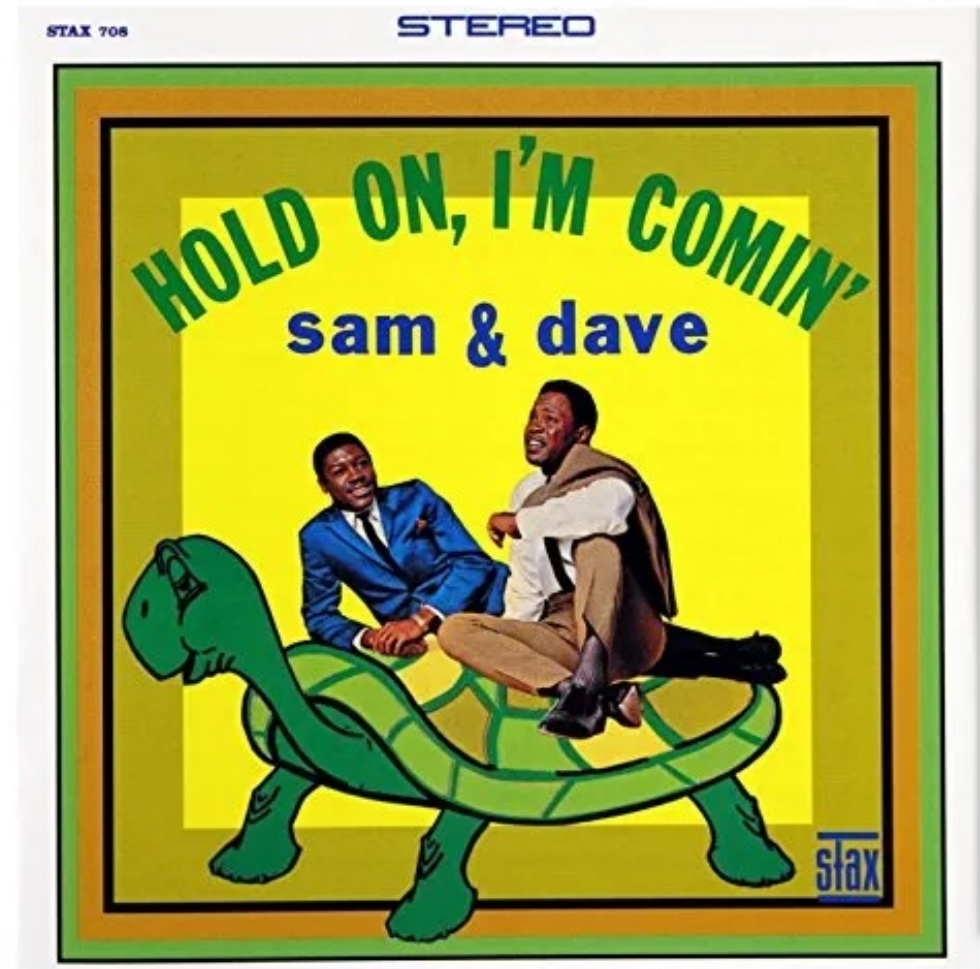
1. Humans avoid the truth
2. Human illusion (I am Superman!)
3. Customer wants and needs are unclear
4. “Impossible to see, the future is”
5. Unclear Req’s
6. Stuff happens
7. “What was it you wanted?”
8. More distractions
9. Dementors
10. We wait for perfection
11. “Oops, I did it again”



“Hold on, I’m Coming”

When in trouble with the Sprint Commitment:

1. A firefighter has a simple plan
2. Collaborate more or better!
3. Fix impediments!
4. Get answers quicker!
5. Do something unexpected! (Get creative team!)
6. Drop the lowest story!
7. Try to fulfill the Sprint Goal, at least.
8. Learn!



Questions

Please ask

Now or later



Other webinars

1. Two types:

- About courses and workshops (30 mins)
- About agile ideas (60 mins)

2. We welcome your ideas!

3. Sign up at our website or at MeetUp (“Lean Agile Training webinars”)

4. Don't forget our courses and workshops!



Contact us

- Please contact us to discuss further. Joseph Little
- info@leanagiletraining.com
- Happy to discuss.
- (704) 376-8881

