# Table of Contents

**How Do Small Changes Lead to Big Improvements?** ..... 3
**Want to Know More?** ..... 5

# How Do Small Changes Lead to Big Improvements?

“Improving daily work is more important than doing daily work” - Gene Kim, The Phoenix Project

One thing that Agile encourages is a more iterative and incremental approach to delivering value. This thinking also applies to how we do the work. Agile expects that you will identify improvements in an incremental way, continuously and relentlessly improving the ability of your Team to deliver value. The expectation is that these incremental improvements build on each other. Since they are small changes, there is potentially a lot less risk in trying something out.

This approach is usually contrasted with the more traditional approach to improvement where every now and again you get everyone in a room and figure out the next improvement in the business process under consideration.

The question becomes “can we rely on these incremental improvements to produce dramatic changes in our ability to deliver value?”

The short answer is “yes”, and the short reason is “the effect of compounding improvements.” Let’s look at how this works.

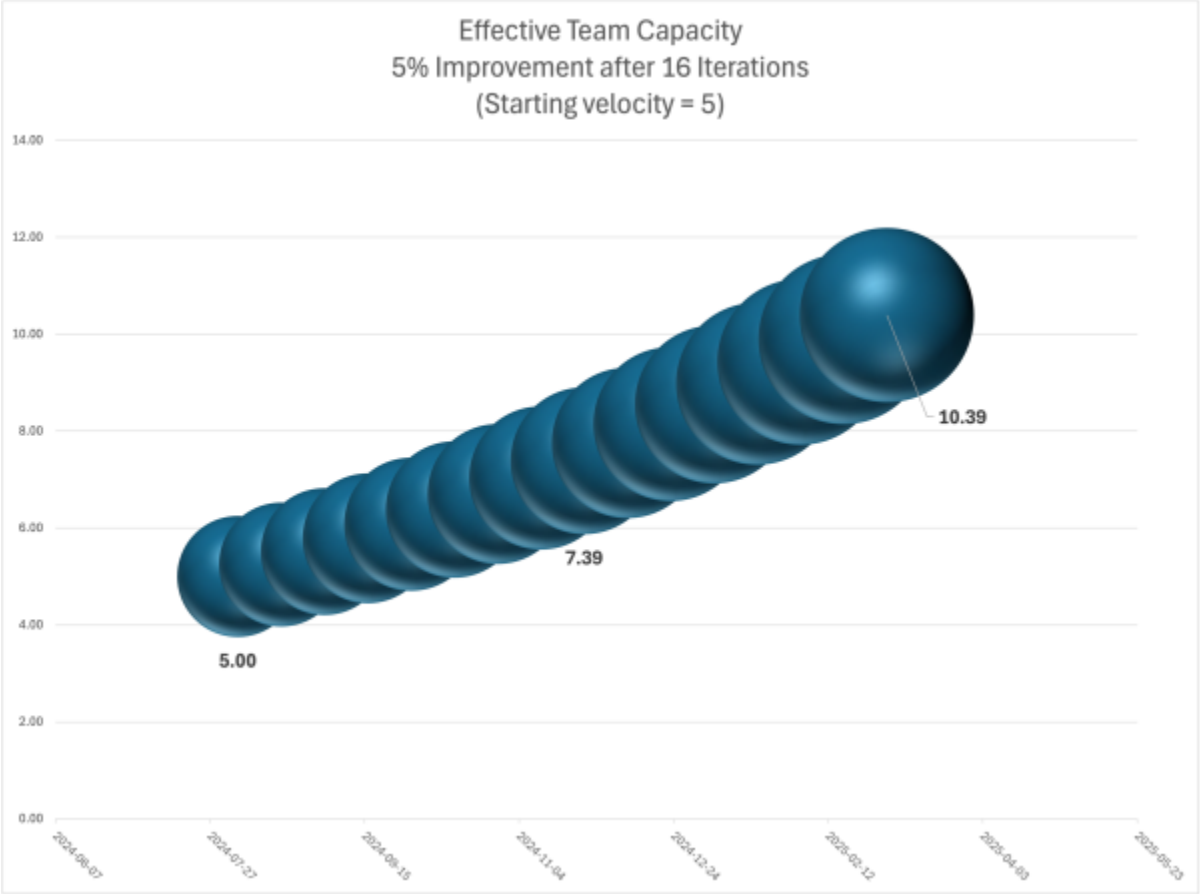
If we assume baseline Team performance is 1 unit of throughput as they form and, as a result of a retrospective we have a (mere) 1% increase in throughput after each and every retrospective, then after a year they will be 29% more throughput each and every iteration (1.01 ^ 26 weeks). If they do improvements weekly the number is 67% (1.01 ^ 52 weeks).

A couple of notes:

- The rate of change is increasing. If it were linear, we should only get 58% (29 x 2) increase, not 67%. This is the effect of compounding improvements.
- The compounding effect is a great argument to do more and more improvement work, not just wait to the next Retrospective.
- This assumes that every improvement identified actually produces that 1% improvement. Since the people that are doing the work are identifying the improvements, my expectation is that the majority of changes should result in improvements. But even if everything the Team tried was a failure the risk is pretty low. You’d end up with a throughput for that Team of .77 (1.01 ^ -26 weeks). And this assumes the Team never goes back to fix their problems.

One interesting side effect of this is the natural concern when people start an Agile transformation that they “lose capacity” as a result of identifying dedicated Scrum Master (SM) and Product Owner (PO) roles. Because of the compounding improvements this is not a concern. Lets say we have typical Team of 7 people, one of which is SM, another is PO. With this improvement model, after a year with 2 week iterations, the Team is functioning at the level of 6.45 people (5 x 1.29 as a %). The 1 week iterations would produce a Team operating 8.35 people.

I realize that 6.45 is still less than the 7 people we started with but the reality is that 1% improvement every couple of weeks is a very low hurdle. Most Teams are able to identify far larger improvements. Newly formed Teams can expect to see significant improvement percentages. Simple tools such as Value Stream Mapping, improvements in collaboration and communication, reduction in impediments required to deliver value and so on are usually easy to find and offer large improvements. All it requires is the capacity and discipline to act on findings. The chart below shows an doubling of capacity if we do a 5% improvement for 16 iterations:




Perhaps the Team can end up somewhere in the realm of 1% improvement per (work) day over a year. The improvement for this team is 1342% (1.01 ^ 261 days)! This is not a typo. That is a 13X improvement in throughput, or a Team operating like it is a 67 people. Fanciful? Probably. But kind of interesting. My personal experience showed a Team with 9X improvement after the first year.

This is all throughput / capacity / utilization type thinking - not exactly Agile where we expect people to focus on (the flow of) value delivered. The assumption we are making is that the Teams are doing the right work based on business priorities. There are thus two potential improvements our customers could see - the right thing faster as a result of good prioritization, smaller batch size and so on, and more of it as well as a result of improvements in our delivery process.

Note: A condition for making these improvements possible is built in Agile:

- There is someone operating in the role of a coach (the SM)

- There is capacity to focus on improvements (at a minimum, the retrospective ceremony).

# Want to Know More?

- How Do We Get Away from “Business as Usual” Thinking On Teams? for ideas on improvement.
- How do we run our first retrospective ceremony)
- What Does a Scrum Master Do All Day?
- British Cycling approach to improving through “marginal gains”

FAQ, retrospective, incremental, compounding, improvement

From:
https://www.hanssamios.com/dokuwiki/ - **Hans Samios' Personal Lean-Agile Knowledge Base**

Permanent link:
**https://www.hanssamios.com/dokuwiki/how_do_small_changes_lead_to_big_improvements?rev=1591132944**

Last update: **2020/06/02 14:22**