

Don't Estimate the Sprint Backlog Using Task Points

by Mike Cohn • 70 Comments

Don't Estimate the Sprint Backlog Using Task Points

Back when I was writing [Agile Estimating and Planning](#) (the 2005 book, but now it's also a video course), I had already been studying and experimenting with estimating approaches for five years. By conducting experiments with various teams, especially those that worked directly for me, I felt I had learned enough to write that book.

But there was quite a bit I didn't know (there still is, of course!), and so I sought out teams that were estimating or planning their projects in ways that differed from the advice I was giving. A few of these teams were using task points to estimate their sprint backlog items in conjunction with story points for the product backlog items.

This was intriguing. By then, I had already become a proponent of story points—I've since become even much more of a proponent because of the advantages that story points offer. But I couldn't see why a team would use task points. However, because I was such a fan of story points, I had to learn more about teams using task points.

To do that, I talked a couple of the teams using task points into letting me visit them so I could see what they were doing and learn more about it.

What I learned confirmed my suspicion that teams should not estimate sprint backlog tasks in task points.

The Main Benefit of Story Points

The [main benefit of story points](#) is that they allow individuals with different skills, experience levels and backgrounds to agree on a relative estimate for work.

As a silly example, suppose that an octopus and I each estimate the effort to wash my car.

The octopus has four times as many arms as I do. So, presumably, the octopus can wash my car four times faster than I can. (I told you it was a silly example. Go with it for a minute.)

But the octopus will also be four times faster than me at washing *any* car. So, the octopus and I can agree that perhaps my two-seat car can be estimated at five story points and a second, larger car can be estimated as 10 story points.

So, then, story points allow the octopus and I to agree on a number of story points, even though the octopus is presumably much faster than I am at washing cars.

Task Points

What I learned from the teams I visited was that a task point was nearly identical to a story point. All the teams I interviewed, though (or have talked to subsequently), wanted to use a more granular unit than their story points.

For example, a team that finished 10 story points per sprint might complete 80 task points per sprint.

Teams simply wanted a smaller, more granular unit to better track progress on things. It would be equivalent to a car's speedometer reporting speed in light years per hour. That would make it very hard for me to know if I'm exceeding the speed limit.

Introducing a more granular unit was a good idea for many teams. After all, it's really hard to gauge daily progress using story points for a team that finishes seven story points in two weeks, for example. For many points, story points are simply too large to be useful for this.

But We Already Have More a More Granular Unit

However, a more granular unit already exists. And it's one that every team member is already familiar with: hours.

When I looked at teams estimating in task points, I could not find an advantage over simply estimating sprint backlog tasks in hours.

There's a [fundamental difference](#) between product backlog items (most commonly, user stories) and sprint backlog tasks: a typical product backlog item will be worked on by three to five people, perhaps a programmer or two, a tester, and maybe a designer, architect, analyst or technical writer and so on. A task on the sprint backlog will usually be worked on by one person.

This fundamental difference means it is not vital for everyone on the team to agree on a task estimate as they should on a product backlog item estimate. Everyone on the team has the right to have input on a task estimate, but those who will do the work should ultimately be the ones who establish the estimate.

If there's only one person who is likely to do a specific task, put that person's estimate on the task. If two or three people might do the task, use a consensus estimate among those individuals or use the estimate of the person most likely to do the work.

If, during the sprint, the work is reallocated to someone else, the worst that will happen is that estimates will swap between tasks. A speedy team member picks up the work of a slower team member and changes an estimate from eight to four, and vice versa.

Why Not Task Points

This leaves task points without a compelling advantage over hours. Yet task points have all the drawbacks of story points:

- A foreign concept to many team members
- The need to establish baseline values against which relative estimating can begin

- A concern that estimates drift over time in comparison to the original relative values

My Advice

I finished my foray into learning about task points unconvinced of their value. I continue to advocate [capacity-driven sprint planning](#) as it's commonly known. But perhaps capacity-driven sprint planning is a better name. I find it to have [significant advantages](#) over [velocity-driven sprint planning](#).

Commitment- or capacity-driven sprint planning in hours works best for most teams. Some teams follow that general process, but throw away the estimates, using them only as a tool during the meeting to figure out what to bring into the sprint.

Other teams don't estimate sprint backlog tasks at all in any unit. Either of these approaches works well once a team has some experience with the approach.

I'm glad I undertook my project to learn more about task points and the teams using them.

In learning about alternative approaches to estimating, I'm always disappointed when a new approach doesn't turn out to be better than an existing one. But, many don't.

Yet, it's by looking at as many options as possible that we find the continuous improvements every agilist seeks.

What Do You Think?

Have you used task points? Did you find advantages to them that I've overlooked? Please share your thoughts in the comments below.

Posted: May 17, 2016

Tagged: estimating, story points, sprint planning, planning, tasks

