

To Re-estimate or not; that is the question

by Mike Cohn •

108 Comments

Should a team that is estimating in story points ever re-estimate? This is a question I'm commonly asked and would like to address here. Most people have a natural feeling that re-estimating is somehow wrong but they can't quite say why. I'll encourage those individuals to stick to their hunches, and hopefully I can provide of the reasoning that supports your natural inclination that most re-estimating is wrong. Philosophers talk about two types of knowledge.

The first is *a priori* knowledge, which is knowledge before you experience something. Let's call this *knowledge-before-the-fact*. This is the type of knowledge we have when we estimate something. Before estimating development of the new search screen I think it's about 8 story points, because it seems to be about the same total effort as some other 8 point story. The other type of knowledge is called *a posteriori* knowledge by the philosophers. This is knowledge *after the fact*. When we estimate it is important that we not mix *knowledge-before-the-fact* with *knowledge-after-the-fact*.

Suppose you are looking at a Scrum product backlog that has just been estimated with none of the work started. Each of those estimates was given before-the-fact (a priori). Now suppose you are looking at the same project a few months later. You've got a list of completed work, some of the items on that list still show their original, before-the-fact estimates but some have been re-estimated with after-the-fact estimates. The product backlog is similarly mixed: mostly the initial, before-the-fact estimates but some estimates that have been revised after-the-fact because of what was learned by developing previous user stories off the backlog. Having both before-the-fact and after-the-fact estimates on your product backlog and list of finished work can cause a lot of confusion for the project. When all estimates are given in before-the-fact numbers we can reason about them and compare them.

Suppose the team is estimating a new item and want to say its equivalent to 20 story points because it's similar to another item that has been estimated at 20 story points. That logic makes sense if the original item has not been re-estimated. If the old item was given an estimate of 10 before the fact and re-estimated to 20 after the fact then it is harder to know if the new item should get a 10 or a 20. With the re-estimation having occurred we're in the position of saying "Before I start this one I think it's a 20 because the other one felt like a 20 after I did it." That's

weaker than “Before I do either of these they seem the same size.”

So, does this mean you should never re-estimate? Absolutely not. There are times when you want to re-estimate. Generally re-estimating is useful when you completely blew it on the original estimate and can see that the mistake was a rare occurrence. (That is, if every estimate is systematically off by half I wouldn't re-estimate.)

Second, you should re-estimate when there has been a change in relative size. For example, the team has discovered that learning AJAX will be about half as hard as they thought. We'd want to fix that because the new knowledge tells us that our relative estimates are off-kilter for the AJAX-heavy stories.

Posted: September 2, 2007

Tagged: user stories, product backlog, estimating, story points

About the Author

Mike Cohn specializes in helping companies adopt and improve their use of agile processes and techniques to build extremely high-performance teams. He is the author of User Stories Applied for Agile Software Development, Agile Estimating and Planning, and Succeeding with Agile as well as the [Better User Stories](#) video course. Mike is a founding member of the Agile Alliance and Scrum Alliance and can be reached at hello@mountaingoatsoftware.com. If you want to succeed with agile, you can also have Mike email you a short tip each week.
