

Table of Contents

How Do We Do Software Capitalization When we Go to Agile?	1
Premise	1
The Basics	1
Want to Know More?	2
<i>Can We Trust Story Points as a Measure of Effort?</i>	2

How Do We Do Software Capitalization When we Go to Agile?

Premise

In accounting, money is not just green, but also comes in two flavors: it comes out of the capital budget or it comes out of the operating budget. If it comes out of the capital budget, then we can defer the recognition of these costs until we actually start selling the result of the effort. If it comes out of the operating budget, the costs drop straight to the bottom line affecting your profitability immediately.

But here is a larger issue. Most of the accounting advice for capitalization was written assuming a waterfall approach. What does this mean? If you fail to take notice of capitalization issues you could end up costing your company millions, or the other way, stopping your company from improving their bottom line. And, if left unchecked this could become an important impediment to your continued use of an agile approach as you try to adapt waterfall reporting requirements to your agile implementation.

The Basics

The good news is that once you have worked through the issue you will find the agile implementation actually makes it easier to capitalize work, with more transparency and consistency. And it takes very little effort from the people doing the agile work.

In software development, we have established rules to determine if it is capital or operating work. A simple distinction is that if the work is associated with a future release, then it can be capitalized; whereas, if it is associated with maintenance work, it cannot.

A release is a candidate for capitalization if:

- It is a major new release, containing significant amounts of new functionality.
- It is a minor release or service pack containing new functionality along with fixes or re-design of existing functionality that will significantly improve the quality of the product.
- It is a major or minor release that contains a significant platform or technology upgrade that will extend the life of the product in the market.

Releases that meet one or all of these criteria can have their costs capitalized starting at concept approval. Capitalization for qualified projects formally begins with approval of the Project Concept document.

Want to Know More?

Perhaps the best description of the approach is by [Dan Greening - Why Agilists Should Care About Capitalization](#)". The approach and results described here exactly mirror my experiences (so I won't write it up again:-))

Can We Trust Story Points as a Measure of Effort?

Many people worry that Story Points do not help us estimate effort. They are after "estimates" - how does that work? The reality is that point based estimates provide a more realistic view than time tracking systems.

It turns out that a story point view of estimated effort is highly correlated with actual work time. Agile practices help teams drive toward high correlation between estimated points and actual time. One goal of agile is to improve overall forecasting starting with the commitment made during sprint planning. Teams that embrace agile estimating principles examine their estimation points and outcomes, trying to ensure that their sprint forecasts are roughly met by the sprint result. If there is wide variation, teams would discuss improvements at a retrospective (see [What Can We Do To Improve Our Point Based Estimates?](#) for specific ideas on this).

In many ways the approach leads to better tracking of effort. When we track stories, estimation points and completion dates (sprint end dates), we know exactly which team did the work, we usually have a day-by-day task burndown and we have a proportional allocation (see [How Do I Convert Points and Velocity to Dollars?](#) for more information). The stories are well documented and understandable (thanks to the user voice form). Team members say the same thing executives say when asked about work as there is a common language and understanding. This means when someone sees something a report and want to ask more detailed questions, no matter where that query leads, there will be a common understanding.

Interestingly, if you do the analysis, and create frequency charts (e.g. How long does it take to complete a 2 point story) the data reflects this consistency. (A version of this kind of analysis can be seen at [Our Estimates are Terrible!](#))

2016/10/03 10:02 · Hans Samios · [0 Comments](#) · [2 Linkbacks](#)
[team, estimates, forecast, faq, points, estimationapproach, capitalization](#)
[FAQ, estimates, forecast, capitalization, executive, finance](#)

From: <http://www.hanssamios.com/dokuwiki/> - **Hans Samios Wiki**

Permanent link: http://www.hanssamios.com/dokuwiki/how_do_we_do_software_capitalization_when_we_go_to_agile

Last update: **2016/10/03 10:36**



