
Improving Project Completion Estimations with Story Points

Randy Smith and
Steve Curtis



1 Executive Summary

One of the challenges facing organizations as they plan, develop, test, and field applications or systems is developing accurate cost and time-to-completion estimates. Clearly, these estimates are critical for planning and managing their enterprise, influencing a wide spectrum of business decisions, including:

- Budgets and schedules;
- Staff Requirements (both skillsets and numbers);
- Customer commitments and contractual obligations;
- Marketing, advertising, and sales campaigns; and
- Infrastructure purchases and leases.

The same challenges arise for organizations providing customer support, training, or any work product that is the result of hours and effort being expended by their people. These organizations need to accurately estimate how many hours and how much effort need to be expended by how many people of various skillsets in order to effectively inform those decisions.

This paper proposes an alternative to legacy approaches that look at hours based level of effort (LOE) estimates and, instead, focuses on the Story Points (or velocity) estimates for a team working on a product's backlog.

This approach integrates the LOE (the amount of work to do) with the complexity of the work and any risks or uncertainties that are associated with doing the work. As a result, an organization can make more accurate estimates that better inform their business decisions.



2 Background

Project managers have long viewed the use of LOE in hours as a solid measure for estimating cost and time to completion. But it turns out estimates in hours are notoriously inaccurate. The reasons are varied, ranging from a general tendency to underestimate familiar tasks and overestimating new ones, to over- or under-valuing risks, to trying to work backward from an established, often arbitrary, deadline.

Unless a team has established detailed, robust, and objective processes for collecting metrics for actual hours, any time-to-completion estimate presented in hours should be considered suspect. Further, cost estimates in hours are equally suspect. Estimates between teams are especially problematic because a disruption to the current team will have cascading effects on subsequent teams and their estimates will quickly fall apart.

For sprint capacity planning, Agile teams have adopted Story Points as a tool for estimating the effort needed to implement any item in a product backlog. Devis scales this same Story Points approach for release and project level planning.

Many Agile process adherents argue that Story Points are not the tool to use for cost estimates. At Devis, we agree that using Story Points for cost estimates has potential pitfalls but, in our experience, can be an accurate estimating mechanism. We have been successful using Story Points for cost-to-completion estimates for individual backlog items, within releases, and within projects.

As a project matures and actual velocity can be measured with increasing accuracy, Story Points can become an ever more useful tool for project managers to estimate project completion and cost.

3 Pre-Requisites

To use a Story Points method there are several pre-requisites a team must have in place. In summary, these include:

- Product backlog the team believes is substantially complete.
 - The backlog can be at the Epic level or a mix of Stories and Epics.
 - The backlog must be pointed (have Story Points assigned). Each item in the backlog must be pointed by the team. These point estimates can be developed using a number of different methods, but in each method, we recommend Fibonacci numbers rather than a linear sequence for assigning estimates. We'll explain why below.
- A relatively stable Team with no significant turnover.
- A few sprints having been completed. Ideally one or more releases have been delivered. As your organization matures and remains stable, your ability to develop reasonably accurate estimates even for backlog items in a first sprint will increase, but in all cases, your estimates for any project will improve as the project continues toward completion.
- All prior work and prior sprints having been pointed and velocity calculated.
- A tool capable of recording velocity and graphing burn down.

- Actual cost data from the project.

Let's take a closer look at several of these pre-requisites and what they mean:

Product Backlog

The product backlog is the compilation of all the things your team still has to accomplish in order to complete a project, typically consisting of all the User Stories and Epics that need to be addressed. Stories are the well-defined or understood stakeholders' needs or requirements, and Epics are larger less well-defined needs or requirements. With a complete product backlog, your team can estimate what work still needs to be accomplished because you have it in front of you.

Pointed Product Backlog

In order to build your estimates of time-to-completion and cost, you have to come to a team understanding of how much work each Story and Epic in your product backlog is going to take to complete it. Each Story and Epic in the product Back Log must be pointed. The pointing will be rough in many cases for many of the items, but they must be pointed.

Completion of Prior Pointed Sprints and Calculated Velocity

It may seem obvious, but it is important to note that the more sprints the team has completed

the more accurate the estimates for the remaining Story Points will be.

Stable Teams

If the same people are a) working on the same project and b) working with the same team, estimates will necessarily be stronger. This is because team members will have a common basis for their scores and will have a clearer understanding of what the team can accomplish in any given timeframe.

Tool for Recording and Graphing

Whether it's a home-grown MS-Excel spreadsheet and line chart graph or the work product of a sophisticated workflow management or continuous development tool, a mechanism for tracking and reporting your team's progress is critical. Understanding how many Story Points the team is able to address in each sprint is the foundation for estimating the time needed to resolve the remaining Story Points. Devis is tool agnostic, leveraging a wide range of alternatives to produce actionable information.

Project Cost Data

In order to maximize the accuracy of your cost estimates, you will want to have cost data from the on-going project that has been charted against the Story Points already accomplished.

The goal is to calculate the average cost per story point and, based on the remaining Story Points, arrive at a projection of the cost-to-completion and, thus, a budget for the remainder of the project.

4 A Closer Look at the Process

4.1 Estimating Initial Story Points

As the project begins, the Team will be gathering User Stories and Epics that define the features, capabilities, and work products desired from the system to be developed. Next these Product Backlog Items will be pointed.

At this stage of the project the Story Points are high level estimations and will be rough. Story Points will be refined later as Stories and Epics are refined for releases and sprints. One method for high level Story Point estimates is to have the members of your team each assign T-shirt sizes from small through extra-large for each item in the product back log. Each item in the back log is given a T-Shirt size, later the T-Shirt sizes are converted in to Fibonacci numbers giving a story point value for each item. Keep in mind that these rough Story Points will change as Stories are refined and Epics are broken into component Stories as they are better defined and pulled into Releases and component Sprints. This process is followed until story point estimates are developed for all Epics and Stories in the Product Back Log. This final number is the Product Backlog Story Points.

4.2 Velocity

As the Team executes each sprint, the team tracks the stories completed and uses this information to adjust goals and expectations for future sprints. The measure of Story Points completed or Velocity is calculated. The total Product Backlog Story Points divided by the number of sprints or releases planned, produces the Average Velocity Needed to Complete (AVNC).

4.3 Cost

The team also tracks actual cost data, based on hours expended and hour rates for each member of the team participating in each sprint.

4.4 Adjusting Estimated Story Points

The initial estimation is, generally, the least accurate because there are fewer details and Stories and Epics are unrefined. As the project matures Product Back Log items become more refined. As the team goes through Release Planning and Sprint Planning the Product Backlog, they may rescore the user stories and Story Points will be updated. This refinement means that the estimation process should be recalculated with each Sprint Planning session. This recalculation milestone, called a project checkpoint, occurs at this time.

The table in Figure 1 depicts a project checkpoint after four of 10 sprints.

Project Checkpoint						Releases Remaining 6		
	Burndown Month	Product Back Log Story Points	Velocity	Points completed to date	Story Point Balance	Project % Complete	Average Velocity Need to Complete	Average Velocity
Start	Start	810	0	0	810	0%	81	0
Release 1	Apr - 17	810	70	70	740	9%	82	70
Release 2	May - 17	810	80	150	660	19%	83	75
Release 3	Jun -17	810	84	234	576	29%	82	78
Release 4	Jul - 17	810	91	325	485	40%	81	81
Release 5	Aug - 17	810		325	485	40%		
Release 6	Sep - 17	810		325	485	40%		
Release 7	Oct - 17	810		325	485	40%		
Release 8	Nov - 17	810		325	485	40%		
Release 9	Dec - 17	810		325	485	40%		
Release 10	Jan - 18	810		325	485	40%		

Figure 1. Project Checkpoint Chart. Summary data describing how much of the project has been completed at the end of each sprint.

As Figure 1 illustrates, the team came up with a total backlog of 810 points and, with 10 monthly development sprints, the AVNC is 81. In this example, the velocity has increased over the four months from 70 to 80, 84, and 91 Story Points. As a result, the project is on target at 40% completion after four months. A simple way to track this is to compare the AVNC and the actual Average Velocity (AV, a simple calculation of the total points completed divided by the number of sprints). So long as the AV trends at or above the AVNC, the project is on schedule.

Figure 2 presents a line chart that demonstrates the comparative arcs of AVNC and AV.

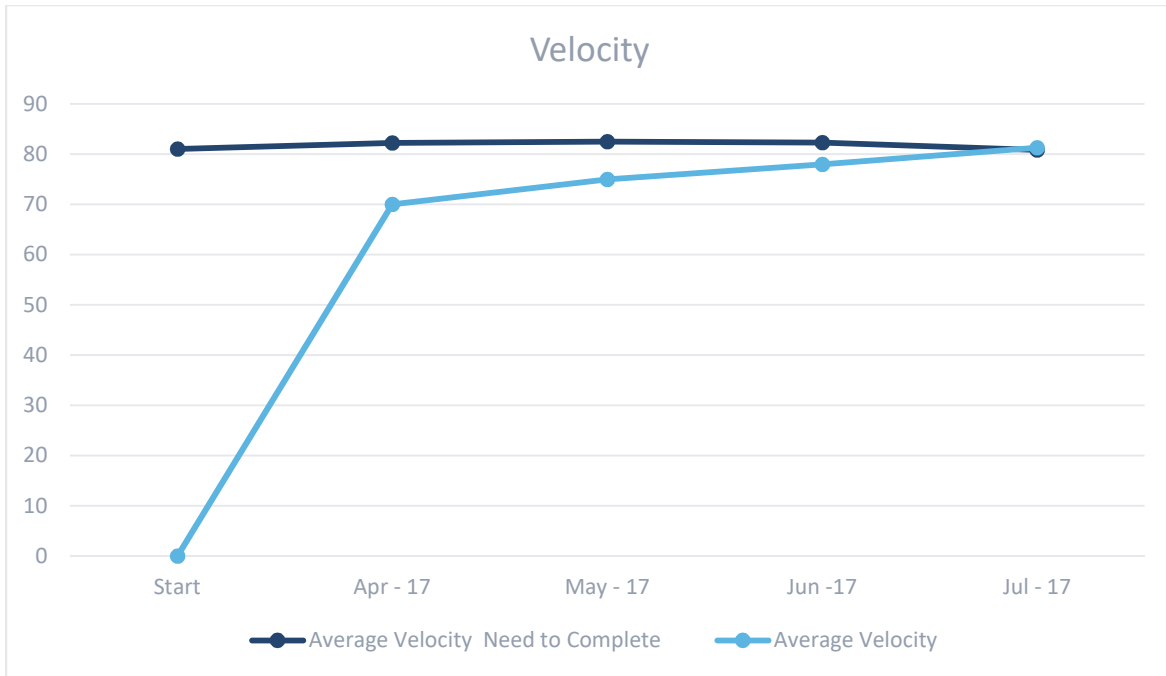


Figure 2. Velocity Comparisons. Comparing AV to AVNC quickly and visually identifies progress on the project.

If the project was well below that 40% mark, the team would need to re-evaluate the project to identify reasons for the shortfall and solutions. These solutions might include additional resources (personnel) or changes in the processes or tools being used. It might also result in a change in the delivery schedule, but the advantage of this estimation approach is that such shortfalls are identified early and solutions that avoid schedule creep are still possible.

Another way to look at performance is to example "Product Burndown." This is a straightforward look at the number of Story Points remaining to be completed. In a Burndown chart, the more uniform the line, the more accurate and stable the estimates-at-completion are, and this is a goal.

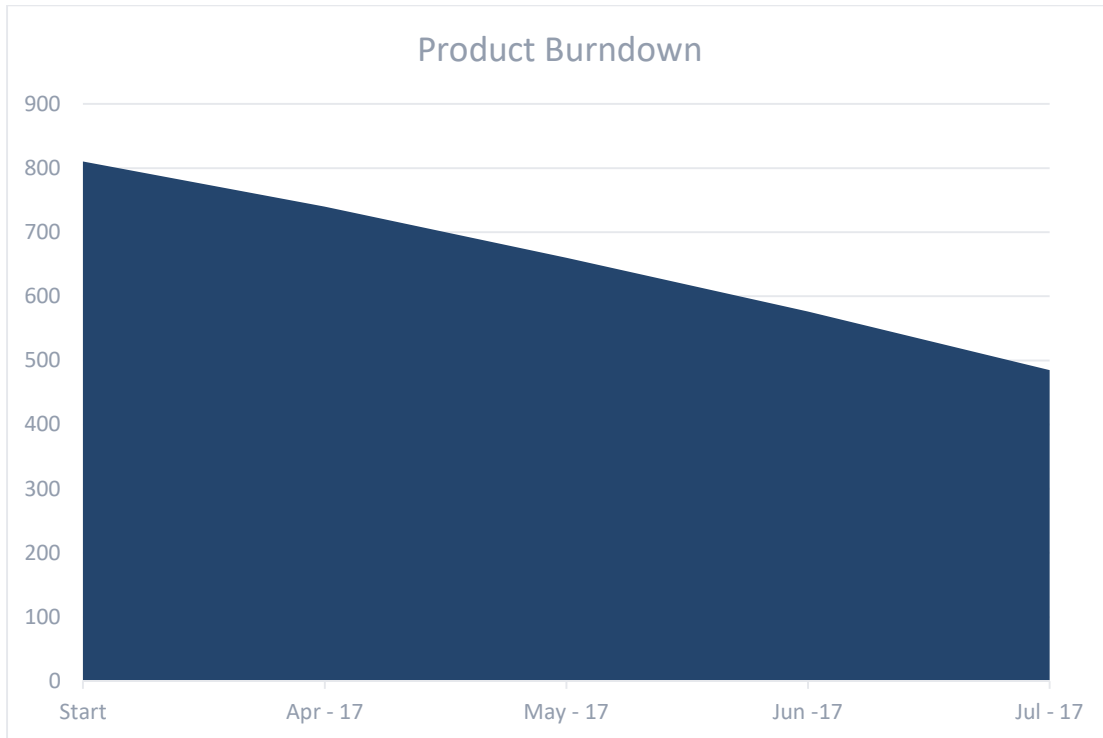


Figure 3. Product Burndown. Showing the steady progress against the product backlog.

4.5 Calculating the Average Cost Per Story Point Completed (ACPSPC)

It is incumbent upon the project manager to manage cost as well as scope and schedule. Based on the velocity calculations and the product Burndown our sample project looks to be on track, but cost must be accounted for.

The following chart calculates the Average Cost Per Story Point Completed (ACPSPC) using the Budget vs Actual financial data, which is a straightforward calculation, as the name suggests. Similarly, we calculate the Cost of Remaining Points (CRP).

Budget vs. Actual							
	Total Budget						
	\$ 500,000						
Burndown Month	Actual Costs	Total Costs	Remaining Budget	% Of Budget	Average Cost Per Story Point Completed	Cost of Remaining Points	
Apr - 17	\$ 45,000	\$ 45,000	\$ 455,000	9%	\$ 643	\$ 475,714	
May - 17	\$ 38,000	\$ 83,000	\$ 417,000	17%	\$ 553	\$ 365,200	
Jun -17	\$ 40,000	\$ 123,000	\$ 377,000	25%	\$ 526	\$ 302,769	
Jul - 17	\$ 57,000	\$ 180,000	\$ 320,000	36%	\$ 554	\$ 268,615	

Figure 4. Budget Compared to Actual. Determining performance against the budget as well as the schedule.

As Figure 4 describes, the project was initially over budget, but as the team found its rhythm and the velocity increased, the CRP fell below the remaining budget. Figure 5 presents this same information in a graphical view. This graph shows based on the calculation of cost per points and the remaining

budget the project is on track for cost. As the project progresses, the CRP should continue to trend below the Remaining Budget.

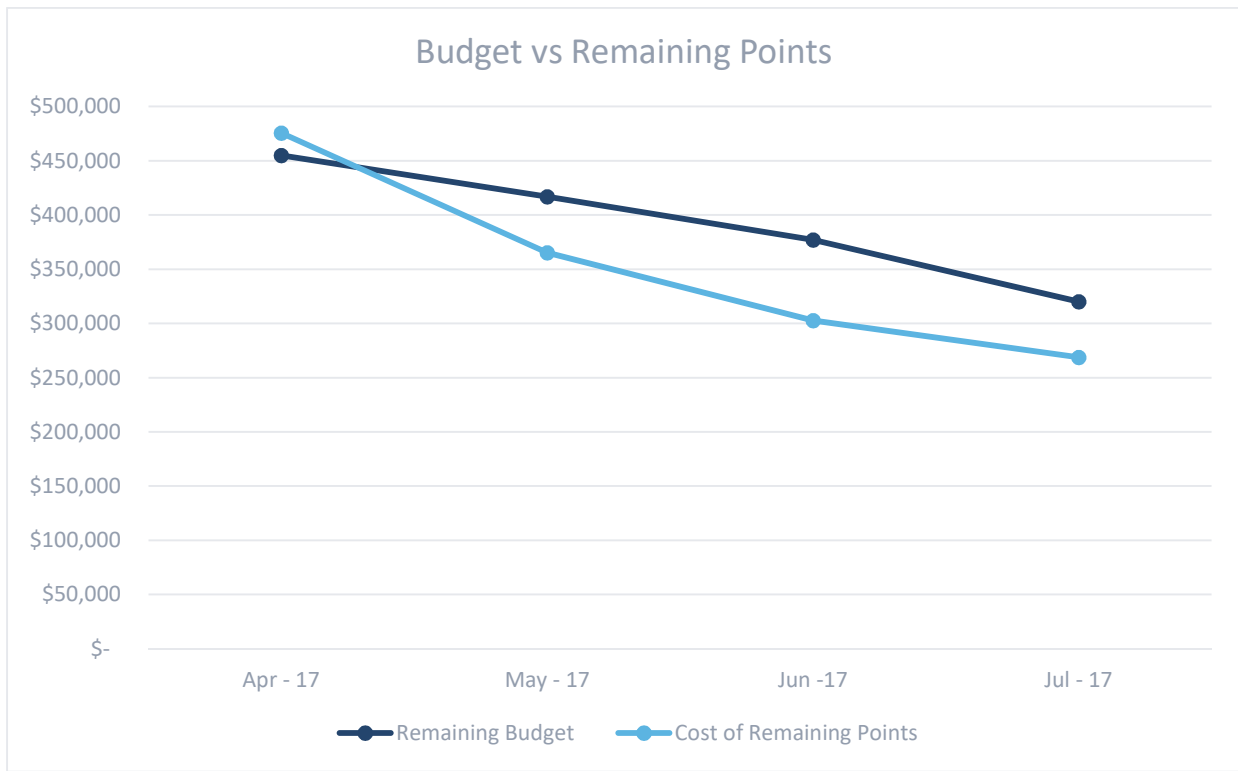


Figure 5. Comparing Budget to Remaining Estimate. Comparing the estimated cost of remaining Story Points against the remaining budget keeps the project on target.

As the project progresses and the positive delta between actual and budget remains stable or grows, the organization can begin to consider shift budget from this project to another.



5 Conclusion

Velocity and ACPSPC are a viable solution to a longstanding challenge: How best to estimate time-to-completion and actual cost for a project. While the basic calculations are similar to those used in traditional estimating methods, Story Points and velocity provide a significant upgrade in setting the basis for these calculations. They draw consensus from the Team (development or otherwise); weigh the requirements based on thoughtful consideration of their complexity, size, and risk; and minimize typical estimation pitfalls to produce a solid foundation for developing estimates.

Over time, especially in an organization with a highly stable Team, stronger and stronger estimates can be developed earlier and earlier in the project.

Devis has a long and successful history of implementing this estimation process across a wide range of use cases and is able to mentor your organization in leveraging your maturing DevOps process to improve your project schedule and cost estimates.

About Randy Smith

Randy Smith is Devis's Chief Operating Officer. His many year of experience in project and program management in addition to his working knowledge and practical experience with Agile processes and tools is a valuable asset for Devis and its clients.

About Steve Curtis

Steve Curtis is Devis's Vice President of Development Services. His experience in the management, delivery and optimization of delivery teams is an invaluable resource to Devis and our clients.

Mr. Curtis' experience, coupled with his PMP, ACP, CSM, and CSPO certifications make him a well-rounded expert in management and delivery in Agile organizations.

About Devis

Devis is a minority, woman-owned small business (WOSB) with more than 25 years as a leading provider of IT solutions to the Federal Government and international development community. We have built our practice focused on solving the information sharing problems faced by public and private organizations with dispersed stakeholders. Our core areas of expertise include:

- *Agile Application Development*
- *Tier 2 & 3 Help Desk Support*
- *Secure, Cloud-Based Managed Services*
- *Worldwide IT Deployment*
- *Systems Integration*
- *Knowledge Management*
- *IT Consulting*
- *Section 508 Accessibility*
- *Software and Business Process Training*

Our accomplished staff has an average of more than 15 years of experience in the IT field, and more than six at Devis. They have travelled on hundreds of TDYs to over 70 countries. Our proven success is a direct result of our certified staff and our ability to develop, deploy, and maintain systems while understanding and supporting our clients' goals.

For More Information

Chris Kagy

VP Business Development

ckagy@devis.com

703-525-6485 ext 126