Improving Project Completion Estimations with Story Points

Randy Smith and Steve Curtis

devis.

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. These estimates are critical for planning and managing enterprises and 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 and/or training or producing any work product consuming organizational resources. These organizations need to accurately estimate how much effort is needed by x number of people of y number of skillsets to effectively inform business decisions.

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

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

2 Background

Project managers have long used LOE (or hours) to estimate cost and time-to-completion, but hour estimates are notoriously inaccurate, though the reasons vary. There is a general tendency to underestimate familiar tasks and overestimate new ones, for example. Risks are also often over- or under-valued and estimated deadlines tend to be arbitrary.

Unless a team has established detailed, robust, and objective processes for collecting time-on-task metrics, any time-to-completion estimate presented in hours should be considered suspect. Further, cost estimates in hours are equally suspect. Estimating time-to-completion and cost for work to be completed by two or more teams proves especially problematic because a disruption to the one team will have cascading effects on subsequent teams.

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

As a project matures and actual velocity is 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 the Story Points method there are several pre-requisites a team must have in place. These include:

- A 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 (or 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 go into further detail below.)
- A relatively stable Team with no significant turnover
- A few completed sprints.
  - 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 were pointed and velocity calculated.
- A tool capable of recording velocity and graphing burndown
- 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 tasks your team needs to accomplish in order to complete a project. This typically consists of all User Stories and Epics that need to be addressed. (Stories are the well-defined or understood stakeholders' needs or requirements; Epics are larger less well-defined needs or requirements). With a complete product backlog, your team can accurately estimate what work still needs to be accomplished.

Pointed Product Backlog

In order to estimate time-to-completion and cost, there needs to be a team understanding of how much work each Story and Epic in your product backlog is going to take. Each Story and Epic in the product backlog must be pointed. The pointing will involve rough estimates for many of the items, but they must be pointed.

Completion of Prior Pointed Sprints and Calculated Velocity

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

Tool for Recording and Graphing

Whether it's an Excel spreadsheet using line chart graphs 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 and leverages a wide range of alternatives to produce actionable information.

Project Cost Data

To maximize the accuracy of your cost estimates, chart cost data from the on-going project against the Story Points already accomplished.

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

4 A Closer Look at the Process

4.1 Estimating Initial Story Points

When the project begins, the Team will gather User Stories and Epics that define the features, capabilities, and work products needed for the system to be developed. Next, the Product Backlog Items should be pointed.

At this stage of the project, the Story Points are high-level estimations. Story Points will be refined later as Stories and Epics are clarified for releases and sprints. One method to create high-level Story Point estimates is to have the members of your team assign T-shirt sizes (small through extra-large) to each item in the product backlog. Later, the T-Shirt sizes are converted to Fibonacci numbers, giving each story point a value. Keep in mind that Story Points will change as Epics are broken into component Stories as they are pulled into Releases and component Sprints. This process is repeated until story point estimates are developed for all Epics and Stories in the product backlog. 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 number of 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, which lead to unrefined Stories and Epics. As the team goes through Release Planning and Sprint Planning the Product Backlog, they may rescore the user stories, thus updating Story Points. This refinement means that the estimation process should be recalculated with each Sprint Planning session. A 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	Burndown Month	Product Back Log Story Points	Velocity	Points completed to date	Story Point Balance	Project % Complete	Average Velocity Need to Complete	Average Velocity
						Releases Remaining 6		
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 created a backlog of 810 points. With ten monthly development sprints, the AVNC is 81. In this example, the velocity has increased over the four-month period 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). As long as the AV trends at or above the AVNC, the project is on schedule.

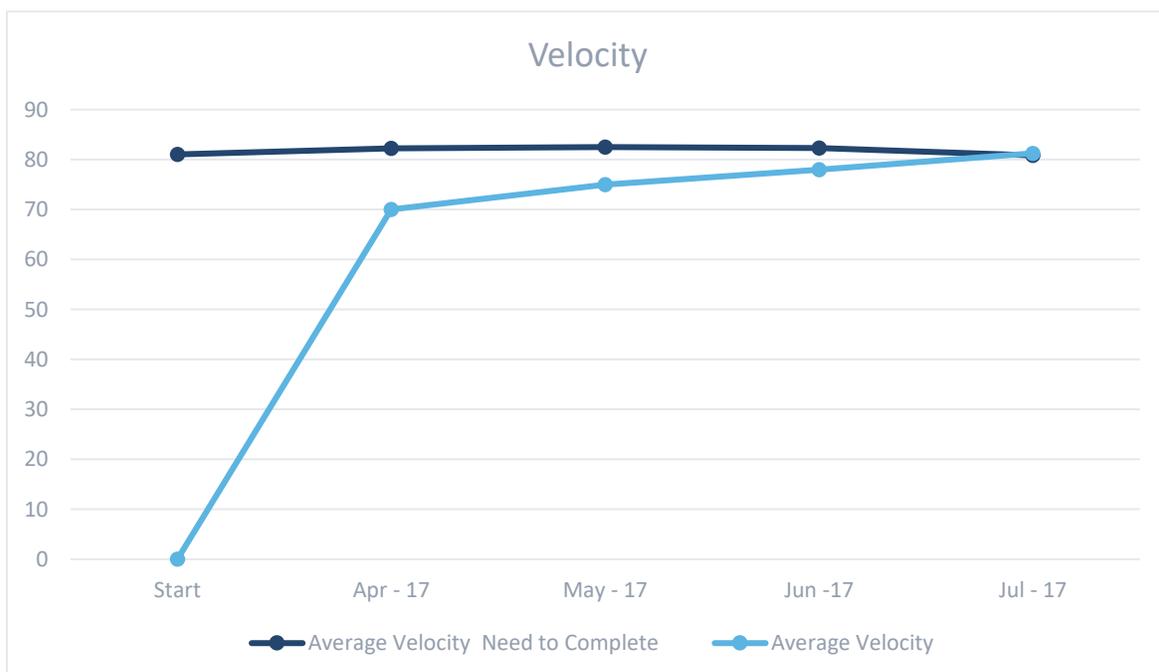Figure 2 presents a line chart that demonstrates the comparative arcs of AVNC and AV.

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

The material in this document is proprietary to Devis.

If the project was well below the 40% mark at four months, the team would need to re-evaluate the project to identify reasons for the shortfall as well as 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 graph "Product Burndown." Product Burndown 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. This is the goal.
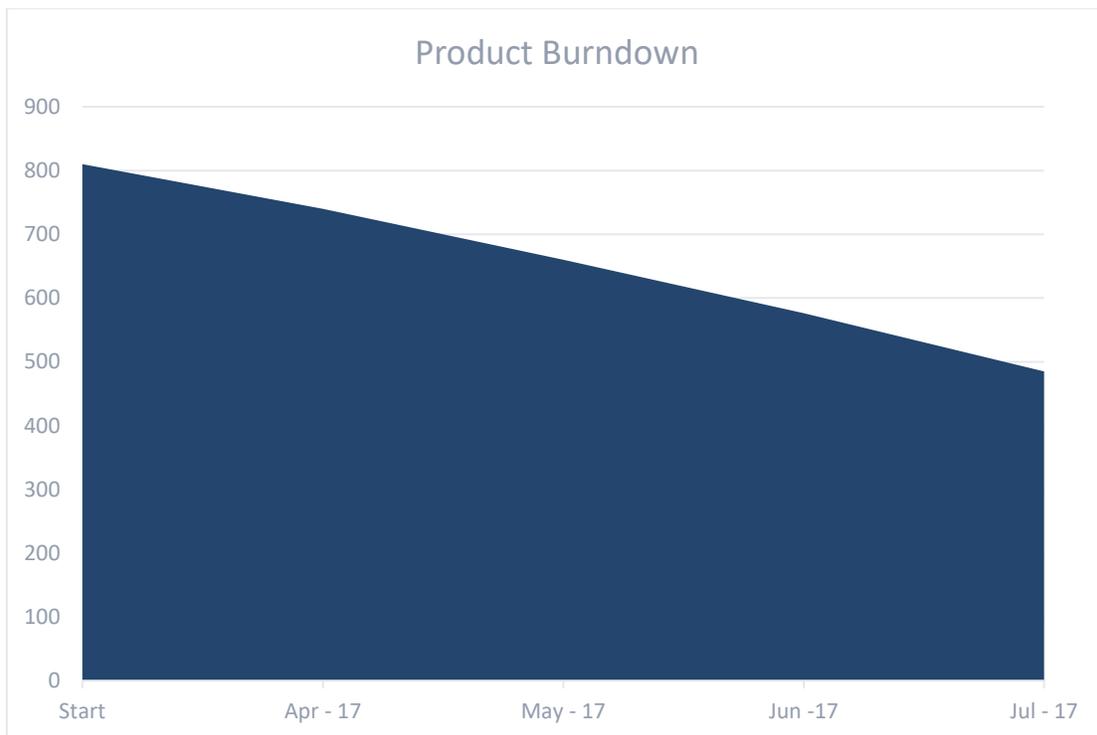

Figure 3. Product Burndown. *Showing the steady progress against the product backlog.*

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

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

The following chart calculates the Average Cost Per Story Point Completed (ACPSPC) using Budget vs Actual financial data. 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 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 that, 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.
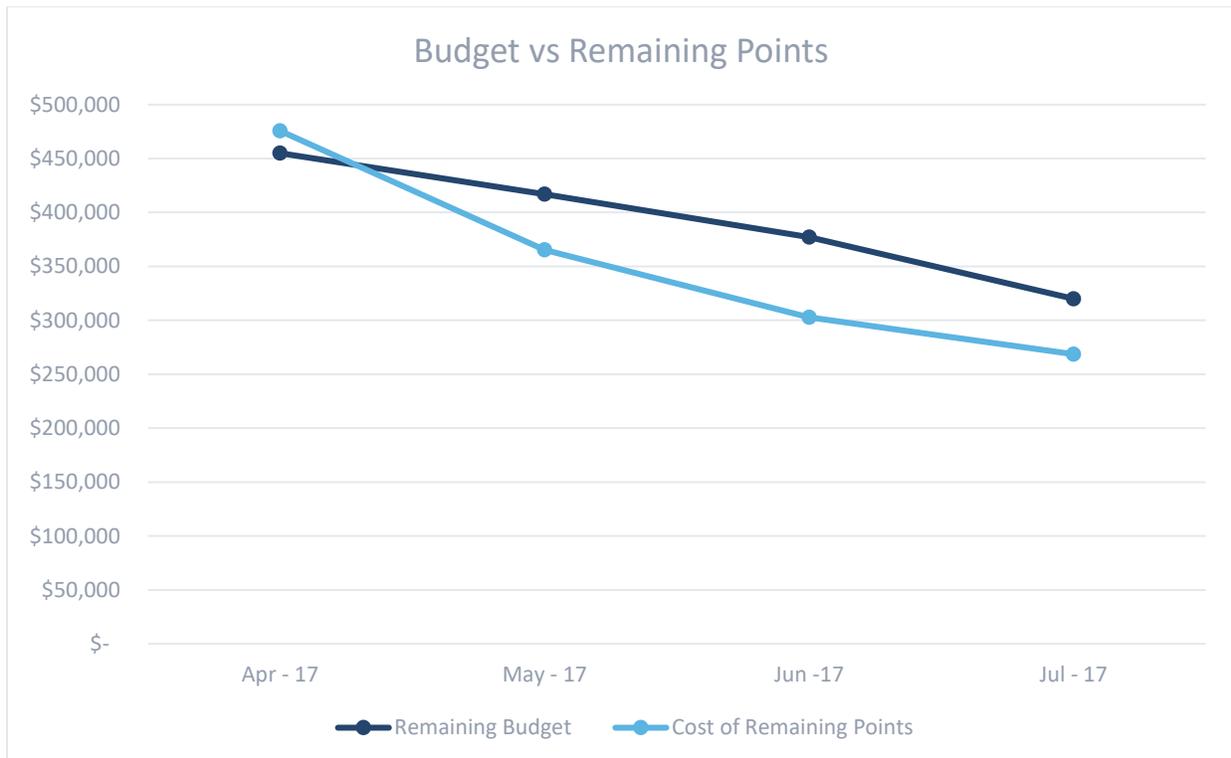

Figure 5. Comparing Budget to Remaining Estimate. *Comparing the estimated cost of remaining Story Points against the remaining budget keeps the project on target.*

As the project progresses and the positive delta between actual and budget remains stable (or grows), the organization can begin to consider shifting 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 estimates can be developed earlier in the project.

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

About Randy Smith

Randy Smith is Devis' Chief Operating Officer. For the last 25 years, he has supported the development and management of information systems, most recently as the Project Manager for USAID's Global Acquisition and Assistance System (GLAAS). Mr. Smith is a Certified Project Management Professional (PMP), Certified Scrum Master, and is ITIL v3 Certified.

About Steve Curtis

Steve Curtis is Devis' Vice President of Development Services. He is an industry veteran, with over 20 years of experience in the information technology field. He specializes in the design, implementation, and management of data-centric solutions – combining a background in database design and object-oriented programming with the industry-leading standards of the Project Management Institute.

Mr. Curtis is a Certified Project Management Professional (PMP), a certified ScrumMaster (CSM), an Agile Certified Practitioner (ACP), a Certified Scrum Product Owner (CSPO), and has completed graduate work in Software Engineering at Virginia Polytechnic Institute.

About Devis

Devis is a minority, woman-owned small business (WOSB) with more than 25 years of experience as a leading provider of IT solutions to the Federal Government and international development community. We have built our practice by solving 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 years of experience at Devis. They have collectively travelled on hundreds of TDYs to over 70 counties. 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