

ScrumWorks® Pro Tutorial: Tracking and Reporting Release Progress

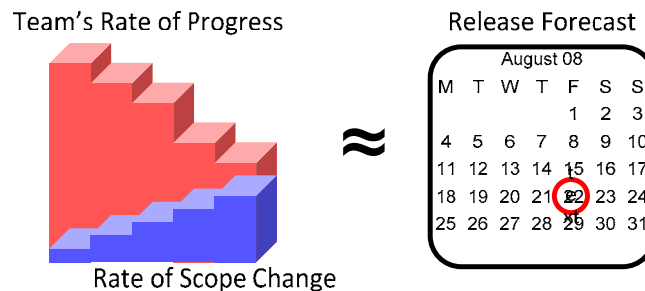
Question:

“How can I use ScrumWorks Pro to establish milestones, track progress, and forecast completion?”

This tutorial addresses the topic by providing a step-by-step guide for using ScrumWorks Pro to forecast Release schedules. Common questions are covered at the end of this tutorial.

The Big Picture: Tracking Team Progress and Rate of Change

At its core, Scrum and other agile processes employ “empirical” metrics. That is, they measure what actually happened in the past, extrapolate that data, and apply it to a forecast. In ScrumWorks Pro, “Releases” represent schedule-driven or feature-driven milestones. ScrumWorks Pro tracks the rate at which Teams finish work in Sprints as well as the rate at which the Release Backlog changes due to scope increases, decreases, and other factors such as re-estimation of existing items. ScrumWorks Pro’s “**Enhanced Product Burndown Chart**” generates a Release schedule forecast by extrapolating this historical rate of progress and historical rate of change. It’s then up to the Product Owner and Sprint Team(s) to make whatever adjustments necessary to meet Release dates or to fulfill feature-driven plans.



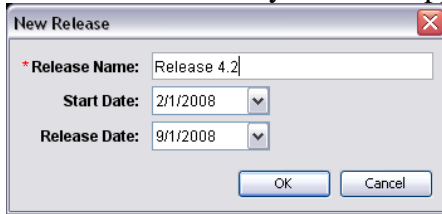
Step 1: Create a Release

The first step in tracking Release progress is to create a Release in your Product Backlog. While the term “Release” may mean something specific at your organization (such as an external customer-facing Release), in ScrumWorks Pro, Releases can also be used to represent internal milestones or any other schedule- or feature-oriented goal. In a broad sense, a “Release” can be used as a subdivision of the backlog for grouping Product Backlog Items.

Releases are created using the menu system: File > Create > New Release

Provide a name for your Release. The name identifies and differentiates the Release, so be as specific as possible.

If your Release is schedule-driven, provide a start and release date. This is not required and for feature-driven Releases (keep working until XYZ feature goals are met), a start and Release date may not even apply.



The 'New Release' dialog box is shown with the following details:

- Release Name:** Release 4.2
- Start Date:** 2/1/2008
- Release Date:** 9/1/2008

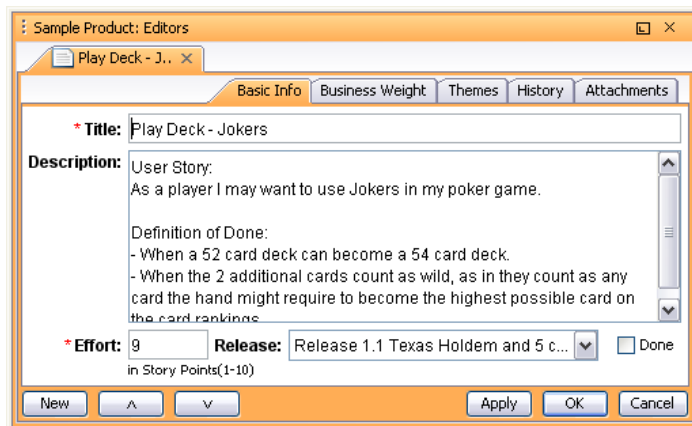
Step 2: Create and Estimate Product Backlog Items

The next step is to create Product Backlog Items or move them into the Release using drag-and-drop. Product Backlog Items represent work items to be completed during the Release in question. The list of Product Backlog Items associated with a specific Release is often termed “Release Backlog.”

Product Backlog Items are most often feature-oriented and usually written in “User Story” format to express some product goal from an end-user’s perspective. For example:

“As a banking customer, I want to withdraw money from an ATM machine using my debit card and my personal identification number, so that I can access money conveniently from many locations on any day.”

Product Backlog Items are prioritized by the Product Owner and estimated by Sprint Teams. Estimation of work at the Product Backlog Item level does not need to be overly precise. Ballpark estimates are sufficient since very little analysis has taken place to date. Estimation units are left to the discretion of each organization, but most Scrum Teams use amorphous, relative estimation units like “story points” as opposed to chronology-based units like “days.” Later in this tutorial, we’ll examine how these rough, high-level estimates can be used to forecast Release schedule.



The 'Sample Product: Editors' window displays the following information for the 'Play Deck - Jokers' item:

- Title:** Play Deck - Jokers
- Description:**
 - User Story: As a player I may want to use Jokers in my poker game.
 - Definition of Done:
 - When a 52 card deck can become a 54 card deck.
 - When the 2 additional cards count as wild, as in they count as any card the hand might require to become the highest possible card on the card rankings.
- Effort:** 9
- Release:** Release 1.1 Texas Holdem and 5 c...
- Done

Step 3: Marking Product Backlog Items “Done”

Product Backlog Items (PBIs) are committed to Sprints during a Sprint Planning Meeting. Before the Team commits to work represented in a User Story, the Product Owner and Team agree on a “definition of done” to clarify expectations around the feature maturity of the PBI. The Sprint Team works throughout the Sprint to finish the PBIs. At the end of the Sprint, the PBI is reviewed by the Product Owner and the product increment is inspected for completeness. If the “definition of done” for any given PBI is adequately met, the PBI should be marked “done”.

Marking a PBI “done” signals that the work is finished and the team receives credit for it in terms of “Velocity” in metrics and reports throughout ScrumWorks Pro. For example, if the Team completes 40 Backlog Effort points out of a total of 50 committed to a Sprint, the Team’s Velocity for that Sprint is 40 points.

To mark a PBI “done,” right-click the PBI and select “Mark as Done.” Alternately, open the PBI editor and click the “Done” checkbox.

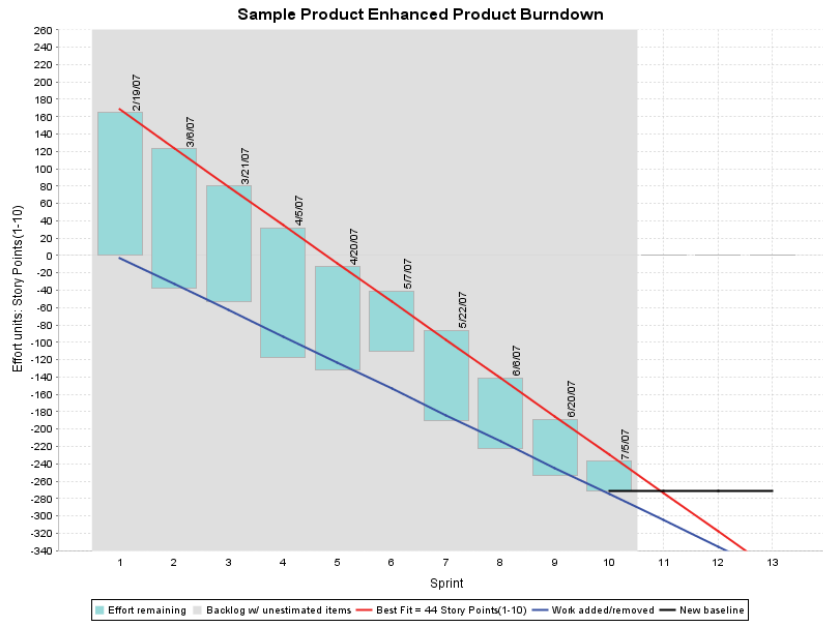
Step 4: “Enhanced Product Burndown Chart” to Forecast Completion

As mentioned in Step 3, when PBIs are marked “done,” the estimation value associated with that PBI counts toward a team’s Velocity. The next step is to examine the historical rate at which the team progresses (their Velocity) and compare it to the rate at which the Release Backlog is changing.

The Enhanced Product Burndown Chart uses the team’s rate of progress and the rate at which requirements change to forecast a release’s completion date.

Access the Enhanced Burndown Chart using the main menu: Reports > Enhanced Product Burndown Chart

Since we are usually trying to forecast completion of a release, on the left side of the dialog select the target release. Once selected, click the “Build Chart” button to refresh the chart.



A new data point is drawn for each Sprint of the Release on the first day of the Sprint. The x-axis represents the Sprints in the Release(s). The y-axis represents the total backlog effort outstanding in the Release Backlog. In combination, the data point bars represent the total Backlog effort outstanding on the first day of the sprint in question.

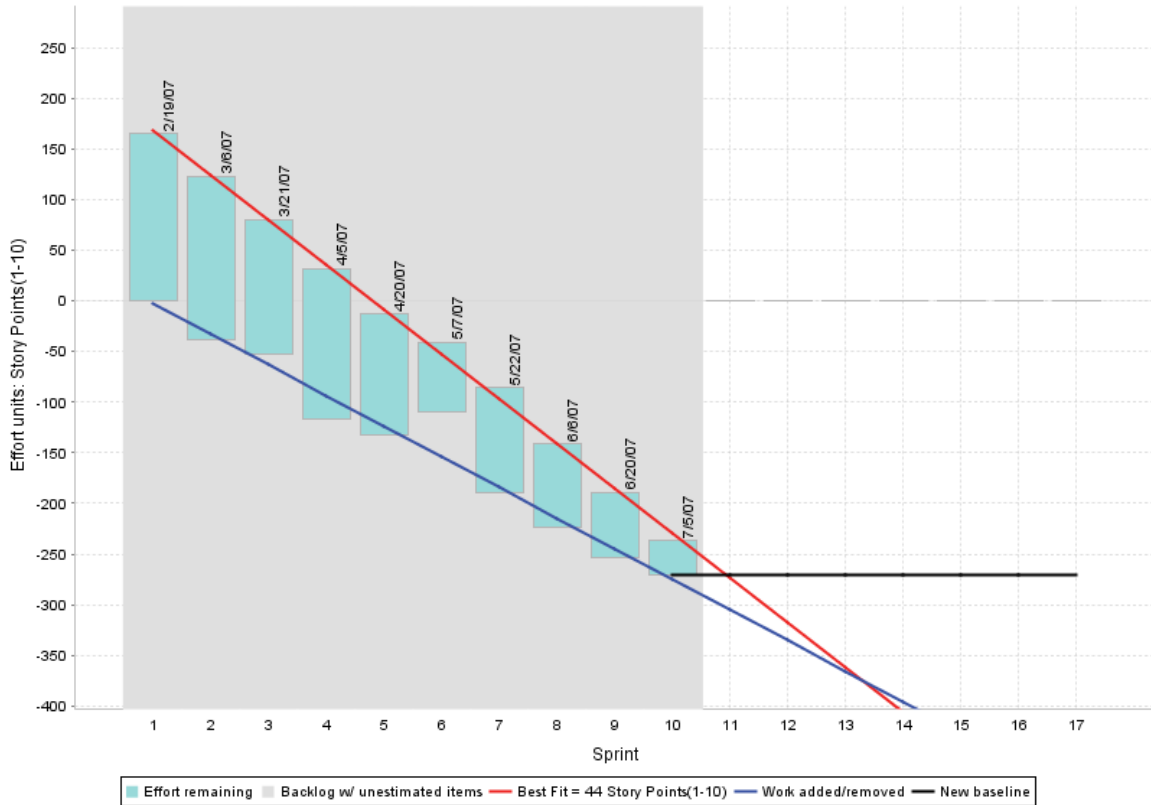
As the team makes progress (by marking PBIs done), the total bar height is lowered by removing effort from the *top* of the bar. A trend line is applied to the team's rate of progress over multiple Sprints, which represents the team's Velocity.

At the same time, if any other change happens in the Release Backlog (e.g., scope change, re-estimation, etc.) the *bottom* of the line is impacted. For instance, if a new PBI is added to the Release Backlog, the bar height is lowered from the previous Sprint's baseline to reflect the increase in overall effort remaining. A trend line mapping to the rate of change affecting the Release is added to the bottom of the chart (blue line).

The Release completion date can now be forecasted by looking for the intersection of the Velocity trend line (red) and trend line reflecting the rate of change (blue).

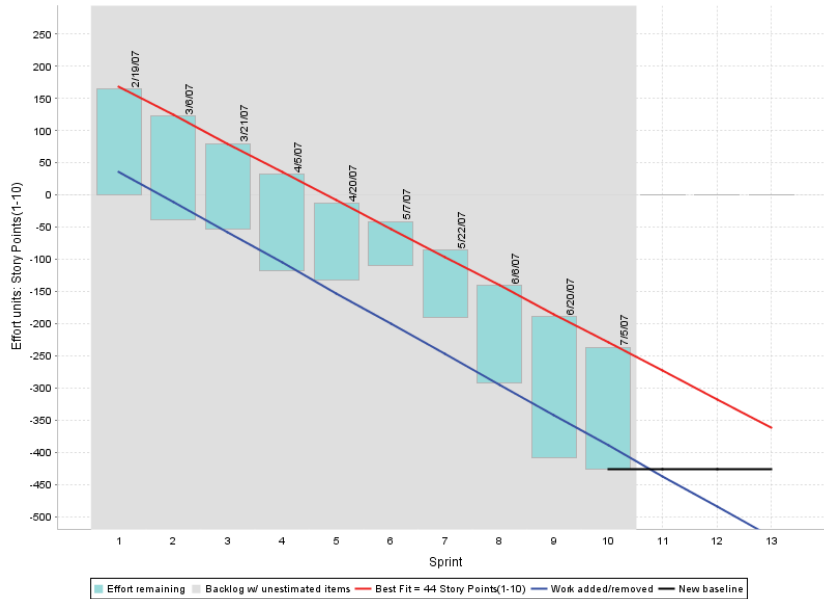
To enable forecast completion, click the "Enhanced Chart Options" tab on the left and select "Forecast completion."

Sample Product Enhanced Product Burndown
Projected completion in 1 - 3 sprints



A Release forecast is only available if the two trend lines have an intersection point — that is, if they converge. A forecast is not possible if the rate at which the Team is progressing is lower than the rate at which the Backlog is growing.

Sample Product Enhanced Product Burndown



Because this chart is based on empirical data, it is more accurate over time as more data points are recorded. During the first few Sprints, a new Team's Velocity as well as the rate of change on the Backlog are seldom stable. Select the current Release plus the previous Release to benefit from an established Velocity derived from a previous Release cycle.

Frequently Asked Questions:

1) What's the difference between Product Backlog Items and Tasks?

ScrumWorks Pro employs two primary levels of actionable work items: Product Backlog Items (or PBIs) and Tasks. PBIs are often feature-oriented and usually written in "User Story" format to express some product goal. For example:

"As a banking customer, I want to withdraw money from an ATM machine using my debit card and my personal identification number, so that I can access my money conveniently at any time."

PBIs are prioritized by the Product Owner and estimated by Sprint Teams. Ultimately, PBIs are committed for work by Sprint Teams. Once committed to a Sprint, the Sprint Team decomposes the PBI into "Tasks" that describe "how" the work will be done. Typical tasks include analysis, design, coding, testing, etc.

In summary, PBIs describe "what" we're building, while Tasks are used to communicate "how" we're building it.

2) How do I forecast the schedule if I don't have a lot of data points yet?

The short answer is that you cannot forecast a schedule using the Enhanced Product Burndown chart with any degree of confidence without at least three or four empirical data points. The point of this chart is to analyze empirical reality, extrapolate Velocity trends, and use the findings to make informed predictions. It makes sense that no real estimate is available until some Velocity and rate of change trends are established.

If you have a new Release, but have previous Velocity and rate of change data points from past Releases, one can include the past Release in the Release selection step prior to generating the chart.

If this is a brand new project or Team with no experience working together, this chart will not be meaningful until the Team has a few Sprints under its belts.