# Lab: View and understand the release burndown report

The Release Burndown report shows you how your team is progressing against the work for a release. In Jira Software, there is no 'release' entity—a version is equivalent to a release (hence, the term 'version' will be used instead of 'release' in this document). The report will show data based on the estimation statistic that your board is using.

Here are some of the ways that you could use a Release Burndown report:

- See how quickly your team is working through the backlog,
- See how work added and removed during the sprint has affected your team's overall progress
- Predict how many sprints it will take to complete the work for a version, based on past sprints and changes during the sprints.

If you have used the Version Report before, you will notice some similarities. However, the Release Burndown report is optimized for scrum teams that work in sprints — which makes tracking much easier.

## Viewing the Release Burndown report

1. Click **Projects** in the navigation bar and select the relevant project
2. Click **Reports** then select **Release Burndown**
3. Select the relevant version from the Release Burndown drop-down. You will be able to choose from versions that are in projects configured for your board (via the board's filter)

If you are using Internet Explorer 8, the Release Burndown will not work.

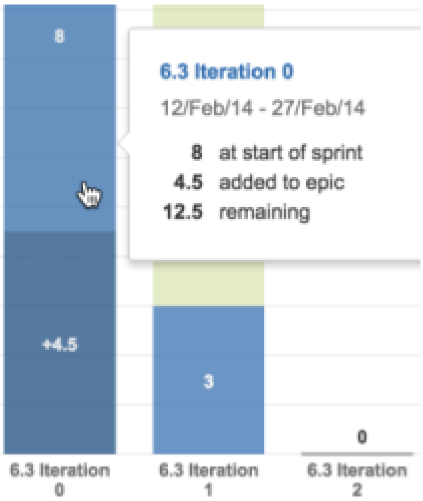
## Printing the Release Burndown report

To print the report, view the report and use the print functionality for your browser. The report will fit on either A4- or Letter-sized pages in both portrait and landscape modes (note, there is a known issue printing in landscape using Chrome).

# Understanding the Release Burndown report

Before you start using the Release Burndown report, you should get to know how it works.

## The sprint bar



6.3 Iteration 0
12/Feb/14 - 27/Feb/14

8 at start of sprint
4.5 added to epic
12.5 remaining

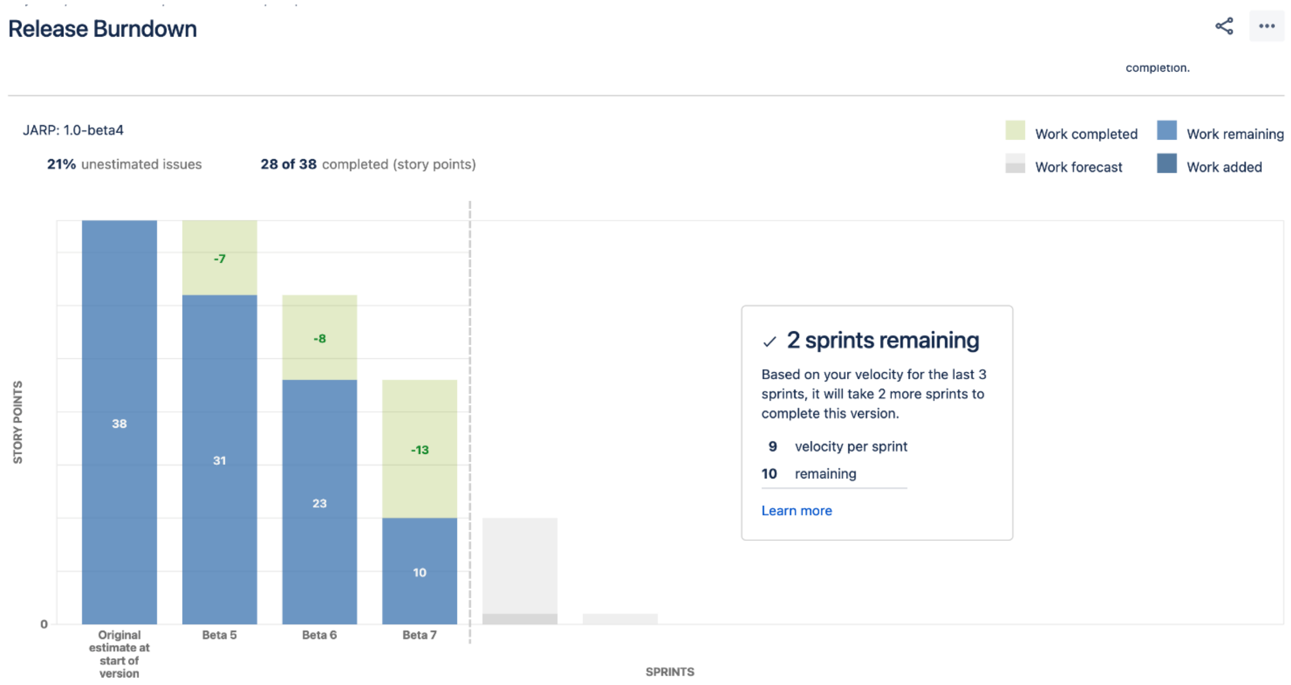
- **Light green section** = work completed during the sprint. If a bar is completely light green, you won't be able to tell how much of the work completed was originally estimated or not. To find out this information, click the bar to view the details.
- **Light blue section** = work that is remaining in the version, out of the total work estimated for the version at the start of the sprint.
- **Dark blue section** = work that was added during the sprint, but not originally included (i.e. scope change).
- **Light green section + light blue section** = total work in the version, that was originally estimated at the start of the sprint.
- **Light blue section + dark blue section** = total work in the version, that is remaining at the end of the sprint.
- **Bars with grey sections** = predicted sprints (see below).

## Predicted sprints

Predicted sprints are calculated based on your team's velocity* (amount of work completed in the last three sprints), and the total work remaining in your backlog. Scope change is not considered when calculating the velocity*, but is included in the total work remaining.

**not the same as the velocity described in the Velocity Chart.*

Consider the following example:



Release Burndown

completion.

JARP: 1.0-beta4

**21%** unestimated issues

**28 of 38** completed (story points)

✓ 2 sprints remaining

Based on your velocity for the last 3 sprints, it will take 2 more sprints to complete this version.

| | |
|---|---|
| **9** | velocity per sprint |
| **10** | remaining |

Learn more

1. **Assessing the outstanding work:** In this example, 10 story points remain for the version at the start of the current sprint.
2. **Calculating the velocity:** 28 story points were completed in the last three sprints (Beta 5 , Beta 6 and Beta 7). This averages out to a velocity of *9 story points per sprint*, rounding to nearest story point.
3. **Predicting the remaining sprints:** At a velocity of 9 story points per sprint, it will take *2 more sprints (including the current sprint)* to complete the work for the version: 10 story points. That is, 1 sprint of 9 story points, plus one final sprint of 1 story point.

### Is my current sprint counted when calculating my team's velocity?

The current sprint is usually not counted when calculating the team's velocity. In the example above, the current sprint bar shows grey sections (like the bars for the predicted sprints) to represent this. The exception is when you have already *completed more work in the current sprint than the work that was predicted to be completed*. In this case, the current sprint (and the actual work completed) is used as one of the three sprints used to calculate the velocity. Also, the sprint bar will show green/blue sections, like the bars for completed sprints.

For example, in the chart above, if your team had already completed more than 9 story points in the current sprint, then the work completed in Beta 6, Beta 7, and the current sprint (rather than Beta 4) would be used to calculate the velocity.

## Other functionalities

The following questions and answers cover the other key functions of the Release Burndown report:

### Why don't the dates for the first/last sprints on my report match the Start date/Release date configured for my version?

The **Start date** and **Release date** configured for the version are shown on the bottom of the report as the **Planned start date** and **Planned end date**. However, these are *planned* dates, and they do not determine the first and last sprints shown on the report.

- The **first sprint** shown is the one that contains the first issue (in the version) that transitions out of the 'To Do' status, i.e. work is started on the version.
- The **last sprint** shown is the one when all work is completed for the version; or if work remains, then the predicted sprint when work will be finished.

The mapping of statuses to your board determines when an issue is considered 'To Do' or 'Done'.

### How does the percentage of unestimated issues affect the report?

The Release Burndown report can only make predictions based on the estimated issues in your version. If you have a high percentage of unestimated issues, then the predictions in the report will not be reliable (the **% unestimated issues** label is colored red when the percentage is above 30%, to help make you aware of this).

For example, if you have only estimated 10% of the issues in the version, then the report predicts the completion of work for the version based on the 10% of the total issues. In reality, your team probably has much more work left to complete.

### What changes affect the original estimate and what changes affect the scope (work added)?

The following changes affect the original estimate of a sprint:

- An issue in a version (before it started) is estimated (estimate is added)
- An issue in a version (before it started) is re-estimated (estimate changes)

The following changes affect the scope of a sprint:

- An issue is added to a version (after it was started) with an existing estimate
- An issue that was added to a version (after was it started) is estimated (estimate is added)

- An issue that was added to a version (after was it started) is re-estimated (estimate changes). Note, if the issue is re-estimated in a later sprint, the scope is retroactively adjusted in the sprint that the issue was originally added to.

**If work is completed outside of a sprint, how is it represented?**

Any change (burndown or scope) that occurs outside a sprint will be shown as part of the sprint with the latest start date before the change date.

**If a completed issue is reopened or added/removed from a version, how is it represented?**

Issue completed in a sprint, then reopened:

- The issue will not be shown in the earlier sprint.

Issue completed in a version, but removed from the version afterwards:

- The scope will remain unchanged and the work completed is still shown.

Issue completed in another version or without a version, but later included in the version (shown on the report):

- The scope will remain unchanged.

Issue completed in a sprint, but only added to the version afterwards:

- The issue will be shown on the report, as if it was always part of the version.

**What if my issue is in an unmapped status?**

If your issue is in an unmapped status (i.e. status not mapped to a column), it will not be considered in the Release Burndown chart. That is, it won't be included in the sprint bars, the % unestimated issues, remaining story points, etc.