In this article, we'll go over how squashing commits affects your Flow data and why you cannot view pre-squashed commits.

In this article

Does squashing commits change the stats in Flow?

Can I see my pre-squashed commits?

Who can use this?

- Core
- Plus

Does squashing commits change the stats in Flow?

Yes, you'll see some change in the reports when squashing.

From Git's perspective, squashing is a form of historical 'revisionism' and alters the Git record. Flow stays true to the Git history, so you'll see some change in the reports when the Git history changes.

Squashing is the process of bundling multiple commits into a single commit. This can be used to combine less important commits into a single commit to make the commit history easier to understand. Since a lot of development happens incrementally, not all of the commits in your history will be valuable to have in the final code.

Note: While these examples are specific to squashing, Flow only tracks code in the increments it’s been committed in, regardless of the specific practice. If code is committed in a “lump sum” rather than incrementally, Flow only tracks the single change.
Actual changes in stats will vary a bit depending on what's being squashed.

Here's a great math analogy to illustrate how squashing will end up affecting the rolled-up statistics:

Suppose you have this math expression:

\[ 2 + 3 - 4 + 6 - 8 + 3 = 2 \]

Squashing this thing would be like placing parentheses somewhere in the middle:

\[ 2 + 3 - (4 + 6 - 8) + 3 = 6 \]
\[ 2 + (3 - 4 + 6 - 8) + 3 = 2 \]
\[ 2 + 3 - (4 + 6) - 8 + 3 = -10 \]

What's curious is that as long as you're only squashing additive parts, it's still possible to sometimes get the same results as before squashing:

\[ (2 + 3) - 4 + 6 - 8 + 3 = 2 \]

So the effect on stats will depend a bit on the exact changes you're squashing.
Consider the following scenario where **commits A, B, C, D, and E** are made on a feature branch early in the week, and then Squashed in **Commit F** on Friday:

![Diagram of pre-merge and post-merge branches](image)

What you’ll see in Flow is just commit F and not the others.

Flow visualizations use Git as the record of authority and display the revised history post-Squash. In the Work log report, we filter by unique SHA. Even though commit F is on the feature and master branch, the commit only appears once in this report alongside the gray merge commit.

**How we got here**

We've arrived at this solution so far because of concerns that inconsistency might lead to misunderstandings.

The challenge is primarily on the data side. Because nobody really stores this information over the long term, the commits post-Squash get trashed eventually.

If we store it ourselves, this leads to some issues with how we might present it:

1. Historical data would always be incomplete—and inconsistent with contemporary data.
2. There can be a near-infinite amount of "true" histories, depending on how branching, squashing, and rebasing happens. It's not clear which one we should present by default.

If you need help, please email Pluralsight Support (opens email form) for 24/7 assistance.