

Can't you just "game" the metrics?

Tags: [Flow](https://help.pluralsight.com/help/search?phrase=:Flow) (<https://help.pluralsight.com/help/search?phrase=:Flow>)

This question comes up quite a bit, and the answer is complicated.

First, what do you mean by gaming metrics? If you're asking about whether you should change all of your coding habits just to get "good" metrics without considering the quality of the work you are producing, the answer is of course not. We don't want you to game metrics like this. Employing bad coding practices to artificially inflate numbers isn't what Flow is about.

If you're asking about whether you should look at Flow metrics and find productive and quality ways to make them go in the right direction for your organization, the answer is yes. Our customers and their teams have actually had a lot of fun by making a "game" out of Flow and seeing how they can improve their numbers. Please game the metrics in this way.

Phrased another way: Flow tracks both outputs and outcomes. Take commits for example. Your coding days is your output, but your efficiency is your outcome. If you focus too much on outcomes without considering your outputs, you risk taking shortcuts to success instead of building sustainable solutions. Or you risk not being successful in the first place.

But if you focus only on outputs, you risk gaming the metrics in a negative way. You get so focused on improving the numbers to the exclusion of all else, that you don't write good code or create positive outcomes for your organization. Placing outputs ahead of outcomes jeopardizes your ability to deliver value to your customers.

Flow helps you build good habits and improve processes to help you optimize for your best end result, combining both outputs and outcomes.

It can be a tough balance to strike, but Flow helps by generating insights within your outputs to better enable you to have data-driven conversations about what is leading you to your outcomes.

Let's take a look at what focusing on a few key metrics might look like

Unreviewed PRs: Unreviewed PRs is the percentage of merged pull requests that did not receive comment from the team. We want to establish a healthy review cadence to ensure that unnecessary risk is not introduced into the code base. A focus on this metric also encourages knowledge sharing. Engineering organizations have world class minds who can aid in growth and career development. Focusing on this metric ensures you are catching bugs ahead of production, while making the team stronger along the way.

Don't start rubber-stamping PRs to increase this metric. That's a risk to your organization. To help strike a good balance, Flow shows both [Unreviewed PRs](https://help.pluralsight.com/help/unreviewed-prs) (<https://help.pluralsight.com/help/unreviewed-prs>) and [Thoroughly reviewed PRs](https://help.pluralsight.com/help/thoroughly-reviewed-prs) (<https://help.pluralsight.com/help/thoroughly-reviewed-prs>). Good managers will look at both to gain a fuller understanding of the picture.

New Work percentage: If you're a senior engineer, chances are you have higher than normal Legacy Refactor in your statistics. This is awesome, but it will naturally cause lower New Work ratios. The best thing to do here is pick your favorite junior engineer and help them get into the maintenance business with you: stop taking on all the risky work and spread the load. By doing this, you'll be able to take on more New Work, which will re-balance your work trend numbers and ensure that new features contain healthy patterns for others to follow. This reduces the amount of "other" work for the whole team, not just you.

Coding Days: This one is easy—just push your work every day. No need to merge into a particular branch, since Flow will see anything you add to the repo. This is a good practice, since it increases visibility and makes sure your work is saved somewhere other than your local machine in the event of a hard drive crash.

Time to first comment: This metric encourages you to grab PRs in a timely manner. Eliminating costly wait states is primarily a courtesy to your peers. Delayed PR review either creates a blocker for individuals on the team, or results in task switching which is arduous and inherently error prone.

Rework: The most reliable way to improve the Rework metric is to be methodical about incorporating any recurring feedback from your code reviews. Before checking in a commit, give your code a once-over and make sure you've handled anything that commonly comes up in your pull requests. Focusing on clean PRs is a great way to reduce Rework: pre-PR edits will nearly always show up as Rework in the reports.

Time to merge: Time, in hours, it takes a pull request to be merged. Unresolved PRs are costly to the business, and long running back-and-forth discourse has a tendency to become unhealthy. Establishing resolution targets helps to verify our work is not done in vain. Attention placed on Time to merge results in a decrease to delayed releases.

What does "good" look like in Flow?

In short—it depends.

There isn't a single answer for what your metrics should look like. Every organization is different, and context often changes. The point isn't to give you a single answer for what your organization should do. It's about giving you enough information to have the conversations with your team, your department, and your organization about what success looks like for you.

Flow can tell you where you're at now, how that's changed over time, and what good looks like for your team. Flow can't tell you what targets to set, or why those targets make sense for you. But we will show you the targets you've set yourself.

Look at the metrics, ask questions, and try something out.

So, can you game the system?

Yes and no. The point is to game the system in a way that drives better outputs and outcomes for your organization. The point is not to inflate or deflate your numbers based on what you think looks good. What looks best is a good product; Flow just helps you get there.

Flow offers you a path to reducing the miscommunication and political overhead that comes from working on things that are invisible to the rest of the business.

Consistently talk about what good looks like, what problems you're trying to solve, and what opportunities you're looking to invest in. Use Flow to show, not tell, that your team is focused in the right areas. Flow gives you the power to share that you're delivering meaningful solutions efficiently.

[back to top](#)

If you need help, please email [Support \(opens email form\) \(\)](#) for 24/7 assistance.