



Engineering Manager Case Study

MOVING FASTER AND SAFER

MEET GEOFF RAYBACK, SR. SOFTWARE DEVELOPMENT MANAGER

Geoff Rayback leads a team of developers that work together on the front end of their website's main application page. Most frequently, this is what users see when visiting the website, which means it receives a high volume of traffic. It's important that the application page, despite the amount of traffic they are handling, is running properly at all times. The team frequently experiments with this experience, and there is a lot of data stored on it. In order to guarantee a positive experience for users, Rayback's team doesn't have room to make mistakes when making changes and rolling out new features. He turned to Split to break the tradeoff between speed and safety.



BEFORE SPLIT, SPEED & SAFETY BUTTED HEADS

With managing the high traffic as the team's main priority, they needed to be strategic with their everyday deployments. In order to not cause issues for users, the team would test every feature before rolling it out, which meant some features sat in the testing phase for days before they were ready to be released.

Although the team was being cautious, this slow deployment process was causing them to be more risk averse and backing up the release of features. The team would release larger features, only a couple times a week. They did not have the flexibility to change or experiment with any of their features before they were rolled out to their users.

"They were talking about it, you've got to pick between resiliency and speed, pick one of those things. And I think that's a totally false statement. You don't have to pick one of those, in fact, if you do it right, speed can create resilience."

- GEOFF RAYBACK

Internally, there was a solution the company had for this issue. The team had a feature flag type tool that allowed them to put these features behind a switch to turn features on or off. However, this tool wasn't made for what the front-end team needed it to do; it was made for storing URLs and took over 10 minutes to update. Within this tool, many features were connected to one another, and changing one part of the code could complicate the entire chain. The team relied heavily on their teammates before they released any new features into production. If there was a feature that didn't pass a test, they would have to roll it back, fix the issue, and re-release the feature. The team was stuck in a cycle: slow testing process, large deployments, and little to no experimentation availability. All of these things caused them to be very risk averse, they needed to reverse the cycle.

AFTER SPLIT, SPEED & SAFETY WORKED TOGETHER

Rayback's company was introduced to Split in 2017, initially to utilize the A/B testing feature, when his team learned they had access to Split's excellent feature flagging capabilities, they ran with it. This allowed the team to test features on a small percentage of users, before officially rolling out the feature for all users. When the team learned that Split would allow them to do more than an on/off switch in testing, everything changed for the team. After applying Split to their practices, Rayback's team had the ability to roll out features by percentage, which was a huge improvement for his team.

In the past, before rolling out a new feature, the front-end team would warn the back-end team to prepare for a large influx of traffic on their service. The back-end team would say they were prepared for traffic, however, they were continuously knocked over because the traffic was more than they anticipated. After getting "knocked over" by the web traffic, the front-end team would have to turn off, or roll back, the feature in order to prevent their site from breaking on the user end.

Split, however, gave the front-end team power to control the roll out of these features with slow release by percentage. Rather than just turning the feature 100% on or 100% off, Rayback's team

"The entire event of us knocking [the back-end team] over and them absolutely collapsing lasted about 60 to 120 seconds. As opposed to 15 minutes of panic and a whole part of the site being down to customers."

- GEOFF RAYBACK

would start off by rolling out to 5% of users and see how the backend team would react. If they were able to handle the small percentage of traffic, the front-end team would increase the percentage. If the team was "knocked over", they

would roll the feature back down to zero and allow the back-end team to recover and prepare for an increase in traffic. This would happen in real time, meaning little to none of their users would experience any issues on their end.

Rayback's front-end team began protecting every new deployment behind a feature flag. By protecting their features, the team had the flexibility to deploy as frequently as they needed and were no longer dependent on their teammates. With each new feature protected by a Split, the team could roll out features at any time of the day. This continuous roll out allowed Rayback's team the flexibility to make small changes that wouldn't affect other teammates. Smaller and more frequent deployments gave the team space to make mistakes without jeopardizing their users' experience.

Additionally, Split updated much faster than the in-house tool the team was previously using. This meant customers saw changes immediately and the team no longer had to wait 10-15 minutes before seeing if their releases were working. Ultimately, the team did not have to panic if a feature wasn't working. Within seconds, they had the ability to kill a feature so that little to none of their users experienced any outage when navigating the webpage. Knowing whether or not a live feature was working for users right away relieved a lot of stress on the front-end team.

"The team does not make [fewer] mistakes. We did not become more disciplined, we did not improve our code rigor, we did not improve our tests, we are not making less mistakes. In fact, I think we're probably making more mistakes than we did before. It's just that the mistakes don't affect [our users]."

- GEOFF RAYBACK

FROM 2 TO 64 DEPLOYMENTS A DAY

Rayback's team went from deploying two or three times a day to deploying a record of 64 times in one day. Without Split, they would have not otherwise been able to do this many deployments. Rayback's team of seven developers have not had a single customer facing mistake in over 4 years. Split has given the team peace of mind, knowing their features, tested or not, are protected and can be rolled back with the flick of a switch. Rayback's team is able to rest easy knowing that their webpage will not crash in the middle of the night if they make a mistake. Now that they have the tools to do their jobs efficiently, Geoff's team has become faster and more disciplined.

BREAK THINGS RESPONSIBLY

Geoff's team went from 2 deployments a day to 64 with Split. Learn how you can increase your release cadence and eliminate risk. Sign up for a [demo](#) of the Split Feature Data Platform™ or switch on a [free account](#) today.



Switch It On at split.io