

GitHub increases productivity and system visibility with Cloud Observability

Founded by Tom Preston-Werner, Chris Wanstrath, P.J. Hyett, and Scott Chacon, GitHub is a cloud-based platform and service for software developers that provides storage, distributed version control, bug tracking, access control, feature requests, continuous integration, task management, and wikis for projects.

Since it was launched in 2008, the site has amassed more than 100 million users and has become the de facto resource for open-source collaboration and community. In addition to a free, open-source tier, the company also offers paid plans with private repositories.

GitHub

Headquarters: San Francisco, CA

Industry Segment: Computer Software

Employees: 2,000+

Challenges

- Experiencing latency with unknown causes
- Incomplete system visibility
- Mixed naming conventions leading to team confusion

Business Results

- Reduced Mean Time To Resolution (MTTR)
- Complete visibility into the GitHub Load Balancer (GLB)
- Adopted OpenTelemetry and semantic conventions

The Challenge

The GitHub team had difficulty resolving an issue in their system: sporadic latency of 20–40 milliseconds with no visibility into where or why. Their engineers investigated it through their normal, manual methods, but they couldn't identify the problem.

Ariel Valentin, Staff Software Engineer, lamented, "Developers have limited visibility into what the cause of the latency stemmed from. We tried to connect things in a log stream from one system to the load balancer in another. None of those keys matched up because folks were using different logging formats. In all those cases, there was no normalization. It was really hard."

The Solution

Using Cloud Observability, their team found that the latency wasn't a part of their microservice but, rather, part of the GitHub Load Balancer (GLB). The GLB manages every interaction in the system and directs all traffic for both internal and external users, which meant this issue was directly affecting millions of developers. The team was also able to pinpoint the exact, end-to-end request that caused the problem (first leaving the monolith then trying to make a call to the auth system) and quantify the impact.

"I now get insights that were previously rooted in my limited understanding of a complex system. What I love about Cloud Observability is that it tells you what is actually happening in your system as opposed to what you think is happening. I cannot imagine doing my job without it. "

Ariel Valentin

Observability Engineer



Outcomes

In addition to improving mean time to resolution (MTTR) and visibility into the GitHub Load Balancer, the team also saw an increase in developer productivity from adopting OpenTelemetry semantic conventions. The SDK and best practices as a standard have saved countless hours of instrumentation when a back-end change is needed. "OpenTelemetry empowers us to build integrated and opinionated solutions for our engineers," said Wolfgang Hennerbichler, Senior Engineering Manager. "Designing for observability can be at the forefront of our application engineer's minds because we can make it so rewarding."