CUSTOMER STORY

Dgraph: Resolving anti-patterns in Go continuously with DeepSource

Dgraph

> "We are using DeepSource for maintaining the quality of Dgraph, our distributed graph database. They have helped decrease the amount of technical debt since it analyzes every GitHub PR."
>
> — Martin Martinez, Distributed Systems Engineer, Dgraph

540
anti-patterns resolved

759
total issues resolved

1.9M+
lines of code analyzed

About Dgraph

Dgraph is an open source, low-latency, high throughput, and distributed graph database. Designed to easily scale to meet the needs of small startups as well as large companies with massive amounts of data, Dgraph can handle terabytes of structured data running on commodity hardware with low latency for real time user queries. It's available under the Apache 2.0 license for easy adoption.

Challenge

Written entirely in Go, Dgraph is built to embrace simplicity and robustness. To keep these intact, maintaining code quality is one of the essential measures. The team had been using a code quality tool to keep everyone aligned. But due to limitations like lack of workflow integration, limited category of issues reported and a few others, it wasn't serving the purpose.

Solution

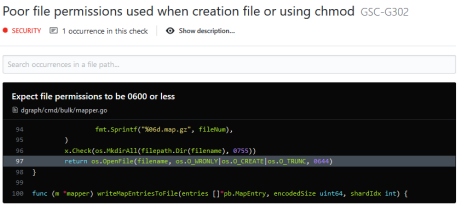
When Dgraph's founder saw the issues DeepSource's Go analyzer was able to detect, he soon onboarded his engineering team to evaluate the tool. DeepSource's native integration with GitHub and the way quality checks fit in the code review workflow, led to quick adoption across.

Within a few days, Martin Martinez, Distributed Systems Engineer at Dgraph and one the team members, noticed the value DeepSource was bringing and further added it to their related Go projects — Badger (key-value store) and Ristretto (highly performant cache).

Identifying and resolving bugs risks, anti-patterns

Today, Dgraph uses DeepSource effectively to detect critical issues across categories like bug risks, anti-patterns, performance issues, security flaws and not just style and formatting warnings. Additionally, since each issue detected by DeepSource is tagged with the category it belongs to, the developers quickly prioritize which one to address first.

Few instances of the issues detected:



Pre-merge reviews reducing technical debt

Earlier, the code quality tool Dgraph was using needed manual effort to run analysis. This becomes a tedious task when you are creating close to 30 pull requests per developer every week.

With DeepSource, Dgraph's code review is now streamlined. DeepSource checks fit right into the code review workflow. The team sees the status of their pull request directly as a GitHub check within a few seconds of creating them. This way every PR is automatically analyzed before the code is merged. DeepSource's pre-merge checks act as a gatekeeper preventing issues from entering the codebase. ***"DeepSource has helped decrease the amount of technical debt since it analyzes every GitHub PR", says Martin,*** after seeing developers regularly resolve issues flagged and the positive impact of this on code quality.

Extensive issue coverage to minimize risks

Dgraph's engineering team is continuously rolling out fixes, features, new versions. In such a fast-paced cycle, maintaining consistency and best practices across the codebase gets difficult. Moreover, they can neither afford the time-intensive code review loops nor the risk of missing issues.

That's where DeepSource adds value. Martin says,

> "DeepSource has larger number of checks than the other tools we tried."

Our dedicated analyzer team adds new issues regularly and keeps an eye on false positive reports as well. This has helped Dgraph team in two ways:

1. Spotting most of the issues in one review itself, resolving to ensure these don't bypass the checks and allowing developers to get back to focusing on building software faster
2. Accuracy of issues encouraging developers to take the reports seriously. The low frequency of false positives has made DeepSource a reliable check

Result

Dgraph has been using DeepSource since October 2019. The team has grown to adopt DeepSource as their go-to code quality tool. Dgraph uses DeepSource to identify issues in their pull request, merge cleaner code every time, and reduce the risk of code breaking in production. Till date, DeepSource has helped

- Resolve 750 issues
- Resolve 540 anti-patterns
- Analyze 1.9M+ lines of code