

## Plaid tackles CI/CD issues 20x faster with Cloud Observability

Plaid is a data network that powers the fintech tools that millions of people rely on to improve their financial lives. Since 2012, Plaid has been focused on democratizing financial services through technology. They build beautiful consumer experiences, developer-friendly infrastructure, and intelligent tools that give everyone the ability to create amazing products that solve big problems.

As an engineering organization building a polyglot architecture (primarily Go, Node.js, and Python), Plaid was looking for a distributed tracing solution with high-quality client libraries that could drive their observability, monitoring, and alerting initiatives.



Headquarters: San Francisco, CA  
Industry Segment: Financial Technology  
Employees: 500

---

### Challenges

- Incomplete visibility into microservices architecture
- Manual process for finding and resolving errors
- High storage and labor costs to build an in-house solution

### Business Results

- Reduced the total time to identify the root cause of integration test failures by 20x
- Removed the need to hire one or more full-time engineers to manage observability efforts
- Enabled engineers to spend more time building features and less time fixing bugs

## The Challenge

Plaid had rolled out a homegrown observability prototype backed by their existing logging stack: Elasticsearch, Kibana, and Logstash. While the stack worked well for logging, it became clear that tracing required a different solution. “We realized that we ingest a huge volume of data and to deploy Jaeger or Zipkin with their own separate data store – it would require at least one full-time engineer for maintenance. That’s when we started looking at hosted solutions,” said Omar Mezenner, software engineer at Plaid.

Besides expanding what types of telemetry they could ingest and observe, cross-service visibility was also important to Plaid. Even more, they needed the ability to dig deep into their microservices architecture and surface exactly what causes an issue. “Imagine you have large backend integration tests touching 20-plus services. Someone has a pull request, but it breaks an integration test. How do they debug that?” asked Mezenner.

---

## The Solution

Using Cloud Observability, the Plaid team can view and interact with their entire system architecture in real time – as it operates, with no ingestion lag. Engineers can see how services relate to one other, reveal dependencies, and dynamically highlight services and components that contribute to the latency or are experiencing errors. Rather than having to comb through all the dependencies, Cloud Observability shows Plaid exactly where the issues are—even down to a specific MySQL query. It fundamentally improves and expedites investigations.

**"It takes us three minutes to know which service is affected, where before required grepping for an hour. This is a 20x reduction in identifying the root cause of an integration test failure. The time savings is huge. "**

**Omar Mezenner**

Software Engineer



## Outcomes

In addition to reducing the amount of time it takes to complete root cause analysis by 20 times, Plaid was also able to remove the need to hire one or more full-time engineers to manage their observability program. And since they're reducing the time it takes to figure out why and where issues are happening in their estate, and even less time fixing bugs, their engineers are able to spend more time making their platform even better.