From: https://a16z.com/2022/02/09/investing-in-asserts-ai/

Investing in Asserts.ai

by Peter Levine

With the rise of the cloud, we are witnessing a major transformation in how software is developed and maintained. As a developer, microservice architecture is quite appealing. It enables us to decouple singular monolithic software into a smaller set of services, making it possible for small teams of developers to rapidly iterate and release new features faster. However, from a systems perspective, the tradeoff for velocity has been operational complexity. We have shifted from local complexity to distributed system level complexity. We have added various abstractions like infra-as-code, kubernetes, serverless, cloud, third party frameworks, and SaaS services to deploy and run these microservices. Imagine the permutations of failure modes across these abstractions and hundreds of micro-services. It dramatically increased the operational overhead causing a significant drag on developer productivity.

As part of our developer productivity thesis at a16z, we believe there needs to be a new paradigm around tools and processes that eases this operational burden on the production side. The software industry has made excellent progress on instrumentation, collection of metrics, traces, logs, etc. Open source projects, like Prometheus, and standards, like Open Telemetry, have democratized the collection and storage layers of the observability stack. However, even with these improvements in instrumentation, developers are still inundated by an almost incomprehensible barrage of metrics. From those dreaded pages in the middle of the night, to logs, traces and dashboards that are created by other developers, the entire troubleshooting process is still very much static and archaic. We have done a good job of answering the “What is broken” question, but not the “Why is it broken” question.

We have been looking for innovations in the “Why” layer, which we call the contextual intelligence layer, that leverages the underlying metrics or logs. Several attempts have been made in the past around anomaly detection and ML, which have failed because high rates of false positives resulted in developer alert fatigue. More than predictive analytics, a developer needs a contextual intelligence that aids them through the aggregated signals in a workbench format when they deal with an alert.

We were thrilled when we met Manoj, the founder of Asserts.ai, who had a similar thesis around contextual intelligence to drive observability forward.

He is the ex-VP of engineering at AppDynamics where he built the core APM product line from scratch, and is deeply aware of the troubleshooting nuances. He and his team tried various root cause analysis (RCA) approaches and eventually came to the conclusion that we needed an assertions-based approach. Manoj and his team are on a mission to achieve true developer productivity on the production side by reducing the time from an alert to an RCA. Similar to the pre-production side, where developers write tests to assert their business logic, Asserts is bringing an assertions approach to the production side that is contextually shown to a developer as part of their regular troubleshooting process.

We are happy to back Manoj and his team as they launch their next generation contextual intelligence product, Asserts.ai, to make developers more productive. If you are an engineering leader who cares about developer productivity and reliable systems, check them out.