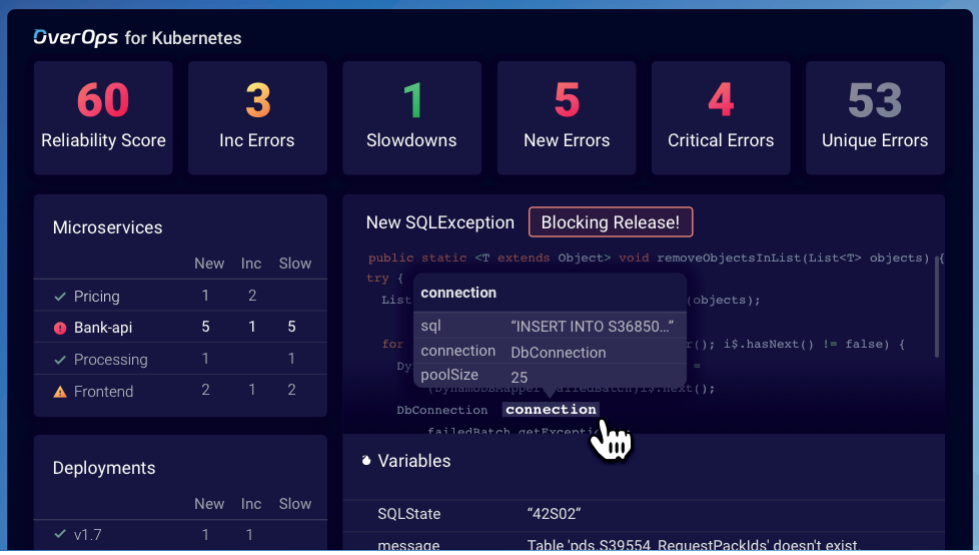
OverOps + kubernetes

# Don't Let Rapid Code Changes Break Your Kubernetes Apps

Identify, prevent and resolve critical issues in your Kubernetes-based apps with OverOps.




Container adoption is changing the way and pace in which we build, deliver and manage software. In this new era of software engineering, traditional tools like log analyzers and APMs fall short, requiring impossible foresight and lacking the depth of context needed to maintain and troubleshoot your Kubernetes-based apps.

Whether you're breaking down an old monolith full of legacy code, or building an entirely new containerized application, OverOps can help. The OverOps solution enables you to quickly identify, prevent and resolve critical issues in your K8s applications – before customer experience is impacted.

### Know **When** Code Breaks

Identify critical errors and slowdowns in real-time and certify the reliability of every release.

### Know **Where** Code Breaks

Identify when releases introduce critical issues. Prevent them from being deployed.

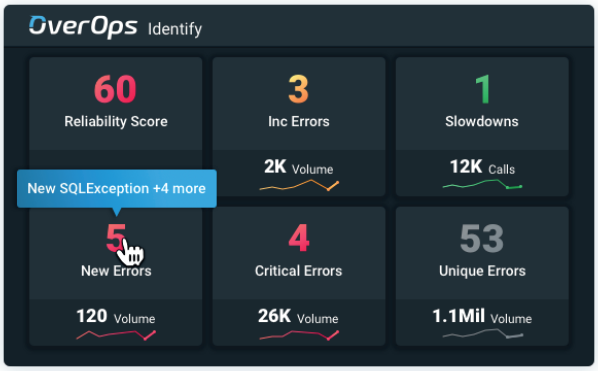
### Know **Why** Code Breaks

See the complete source code, variables, DEBUG logs and container state behind any error or slowdown.

Check out how OverOps helps maintain code quality in your containerized apps: **www.overops.com/product**

# Ensure Continuous Reliability in Your Container-Based Apps

OverOps enables you to understand what's happening inside your application and get a complete picture of the code, even when deployed to hundreds or thousands of individual microservices.

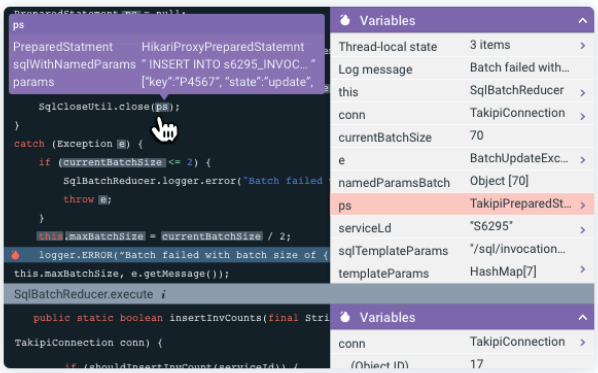



## Identify Critical Anomalies in Your Kubernetes Apps

- New errors introduced by each release
- Error spikes via built-in machine learning
- Slow methods compared to previous releases

## Prevent Issues Using Quality Gates

- Identify critical errors and slowdowns in every release
- Block unreliable releases from being deployed
- Report on prioritized issues and their cause


## Resolve Errors Using Complete Context

- Code variables across the entire call stack – 10 levels deep
- DEBUG-level logs directly from the JVM/CLR
- Complete environment state of the container

Learn more about OverOps for Kubernetes: **land.overops.com/overops-for-microservices/**