OverOps

Where does OverOps fit in your monitoring ecosystem?

Most organizations use a suite of tools to understand how their applications are behaving, yet still lack needed visibility. Here's how OverOps fits in and fills the gaps in your SDLC.

OverOps picks up where log analyzers and APM solutions leave off, going wider and deeper to provide extensive contextual data about the root cause of errors and the overall quality of your applications.

APM tools show you the when and the where.

OverOps shows you the why and the how.

Traditional Application Performance Monitoring tools help you identify when and where your application is experiencing performance or availability issues, but typically don't provide the complete picture of an error's root cause. OverOps supplements APM data by injecting links directly in your APM instance that provide the complete source code, objects, variables and values that caused an error or exception to be thrown.

APPDYNAMICS dynatrace New Relic.

Log analyzers help find the errors you expect.

OverOps finds errors you never saw coming.

Log analyzers store application logs to make them easily searchable, especially when troubleshooting. Logs fail to capture every error and often contain too many entries, requiring you to manually find the needle in the haystack. OverOps captures 100% of errors – even those uncaught – and inserts log links with the stack, source and state behind each event so you can easily reproduce them. We also de-duplicate the logs' contents to reduce operational noise.

splunk> elastic sumo logic

Identify errors not found in logs & APM solutions.

OverOps adds True Root Cause across the entire SDLC.

Developers

Fix code faster and overcome roadblocks in existing data

QA

Systematically prevent bad releases from being promoted

DevOps/SRE

Analyze trends and prioritize anomalies in existing & new data

	Log Analyzers	APM	OverOps
Anomalies			
Logged errors & warnings	✓	✓	✓
Unknown errors (uncaught/swallowed)	✗	✗	✓
Increasing errors	✗	✗	✓
Slowdowns	✗	✓	✓
True Root Cause			
Call stack and exception message	✓	✓	✓
Source code	✗	✗	✓
100% variable coverage	✗	✗	✓
Complete log and JVM state	✗	✗	✓

> “We could think of at least a 100 different reasons for [an exception] to be thrown, and investigating the error with logs and APM didn’t make sense. OverOps immediately identified the exception’s cause, when APM tools and log files were no help.”
>
> Sumit Nagal | Principal Engineer in Quality at Intuit

See how OverOps maximizes your existing investments: www.overops.com/integrations