

OBSERVABILITY · OPERATIONALIZED

Your systems are talking. How are you mapping them?

Bridging the gap between system signals
and business context.

Most enterprises have monitoring. Very few have **business context**. When a transaction fails, a journey slows, a spike hits, the real question isn't what broke.

"What is this costing us right now?"

\$5K – \$1 M/hr

Gartner: Estimated downtime cost range for large enterprises **

**Source: Gartner (cited in 2024–2025 industry analyses)

It takes more than monitoring tools. Every signal must be interpreted **based on the use case**. That's the gap QKTech and Datadog are purpose-built to close.

WHY QKTECH + DATADOG

QKTECH BRINGS

Business Journey Intelligence

Deep expertise mapping IT signals to **business transactions**: onboarding, payments, claims, across complex enterprise stacks.

DATADOG BRINGS

Unified Observability Platform

APM, logs, infrastructure, and RUM connected in one platform. 700+ integrations, enterprise-grade at scale.

— Signal noise becomes business clarity.

INSTEAD OF CHASING ALERTS, YOUR TEAMS WORK WITH

1

Contextualized Incidents

What, where, and why.
One view. No more tool-hopping.

2

Cross-Layer Correlation

Application → Infrastructure → User impact. Connected end to end.

3

Pre-Defined Business Views

Journeys like **onboarding, payments, claims**. Always visible, always prioritized.

HOW SIGNALS FLOW ACROSS THE STACK

 **Application** →  **Infrastructure** →  **User Impact** →  **Business Context**

PROVEN IN COMPLEX ENTERPRISE ENVIRONMENTS

80%

Faster Issue Identification

Before it becomes a business incident

50%

Faster Resolution

Less diagnosing.
More resolving.



Reduced Customer Impact

Fewer transaction failures.
Fewer escalations.

FOR LEADERSHIP, THIS TRANSLATES TO

- Issues surfaced **before they reach the customer**
- Business and IT with a **shared understanding of what matters**
- Leadership with **control and speed**. Not just dashboards.

**The teams winning on reliability
didn't get more alerts.
They got context.**

Let's show you what that looks like in your
environment.