chronosphere

What are the 3 phases of observability?

Overview

Before DevOps, not many engineers needed to think about operating the systems that they built. Now that engineers both build and operate, it's critical to think about building systems that are easier to observe. It is much easier to maintain and manage an observable system than a non-observable one. But this can be easier said than done. Looking at the outcomes you are trying to achieve – ensuring performance of your cloud native apps – there are three observability questions you must answer to ensure success:
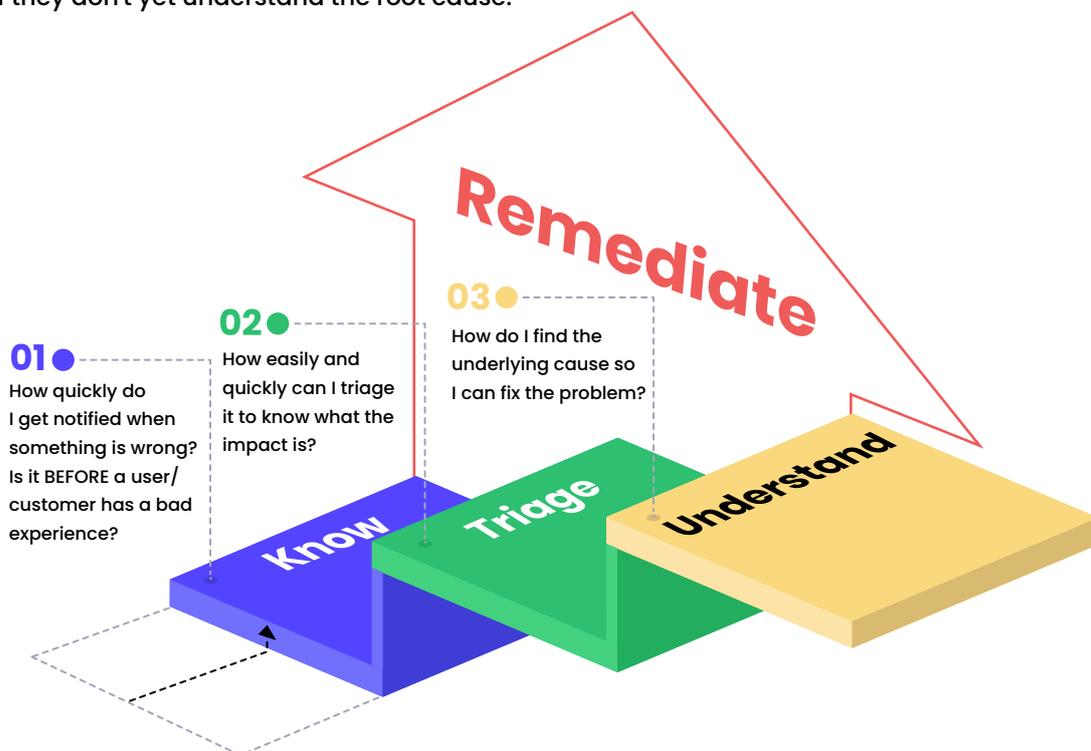
✓ How quickly are you getting notified when something is wrong?

✓ Can you rapidly triage the problem and understand its impact?

✓ How do you find the underlying cause so you can fix the problem?

Why you need observability now

The need to introspect and understand systems and services is not new — many of the basic goals of observability have been in practice for decades. What's changed is the nature of the applications and infrastructure that teams are operating. Moreover, cloud adoption is now surging in response to the pandemic. Instead of letting change negatively impact your outcome, implement the phases that lead to great observability: know, triage, understand.

The goal: Rapid **remediation**

Understanding and embracing the three phases of observability is the best way to respond to these questions. During each phase, the focus is on alleviating the customer impact — or remediating the problem — as fast as possible. Remediation is the act of alleviating the customer pain and restoring the service to acceptable levels of availability and performance. At each phase, the engineer is looking for enough information to remediate the issue, even if they don't yet understand the root cause.

01 How quickly do I get notified when something is wrong? Is it BEFORE a user/customer has a bad experience?

02 How easily and quickly can I triage it to know what the impact is?

03 How do I find the underlying cause so I can fix the problem?

Remediate

Know   Triage   Understand

Phase 1: **Know** about the problem

The first step to resolving an issue is knowing the issue exists — ideally before your customer does. Sometimes, just knowing an issue is occurring is enough to trigger a remediation. For example, if you deploy a new version of a service and an alert triggers for that service, rolling back the deployment is the quickest path to remediating the issue without needing to understand the full impact or diagnose the root cause during the incident. Those can be examined after the issue is remediated, when there isn't active customer impact. Introducing changes to a system is the largest source of production issues, so knowing about problems as these changes are introduced is key.

Phase 2: **Triage** the problem

The goal of this phase is to quickly understand the context and impact of an issue. Once an alert goes off, if it is not obvious that a recent change to the system needs to be rolled back, the next step is to understand the business impact and the severity. Often, understanding the scope of the issue can lead to remediation. To help triage issues, you need to be able to quickly put an alert into context of understanding how many customers or systems are impacted, and to what degree. Great observability allows you to pivot the data and shine a spotlight on the contextualized data to diagnose issues.

Phase 3: **Understand** the problem

This phase occurs ideally after remediation, when you can take the time to locate and understand underlying issues without the pressure of a ticking clock of customer expectations. With an ever increasing volume of microservices, doing a post mortem on an incident is often an exercise in navigating a twisted web of dependencies and trying to determine which service owner you need to work with. Great observability gives direct line of sight linking your metrics and alerts to the potential culprits. Additionally, it provides insights that can help fix underlying problems to prevent recurrence of incidents.

Bracing for a cloud and cloud native spike

Cloud native architectures are experiencing rapid growth

Public cloud infrastructure market will grow 35% to $120 billion in 2021

60% of companies will leverage containers on public cloud platforms

30% of developers will use containers regularly by year end

> "No one knows how far we'll continue to work from home, shop primarily online, or avoid air travel — but it's clear that every enterprise must become more agile, responsive, and adaptive than ever before."

Dave Bartoletti

Vice President, Principal Analyst, Forrester

Predictions 2021: Cloud Computing Powers Pandemic Recovery