



E-BOOK

The Essential Guide to Observability

Use data from your systems, services, and
apps to drive your business

Table of contents

Ever heard of digital transformation?	3
What modern applications need—observability	4
The three pillars of observability	6
Building a better observer experience	7
Artificial intelligence and machine learning in observability	9
OpenTelemetry	13
Key metrics	14
Observability in practice	19
Why Sumo Logic?	20

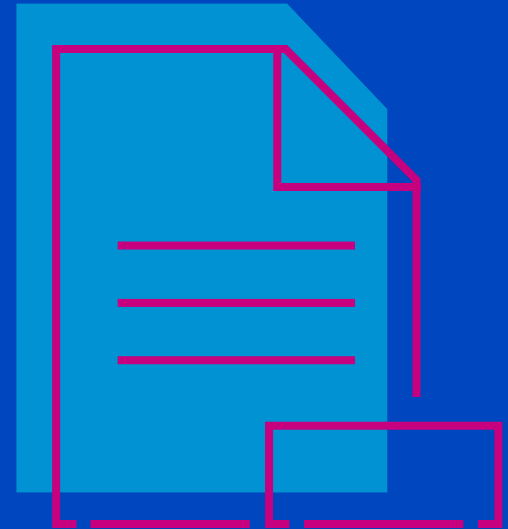
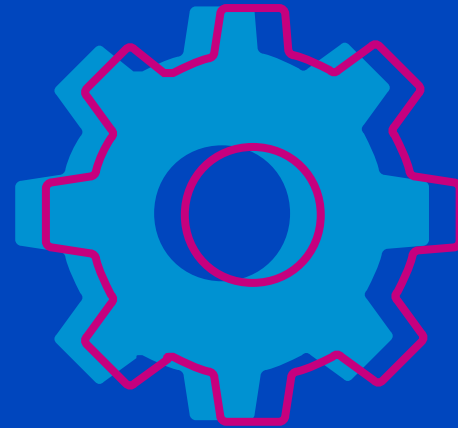
Ever heard of digital transformation?

Digital transformation shifts analog business models to digital formats and is one of the most predominant forces in business today. Think of the rise of Amazon, Netflix, and Airbnb and the highly engaging, revenue-generating services these platforms deliver compared to their predecessors.

But this significant digital opportunity brings risk, too.

A poor customer experience in one physical store is far less likely to derail your business than your digital shopping cart feature failing during peak online shopping. Even short-lasting issues can have a significant effect because they can impact thousands of customers at once.

In the world of digital services, customers are won and lost in seconds.

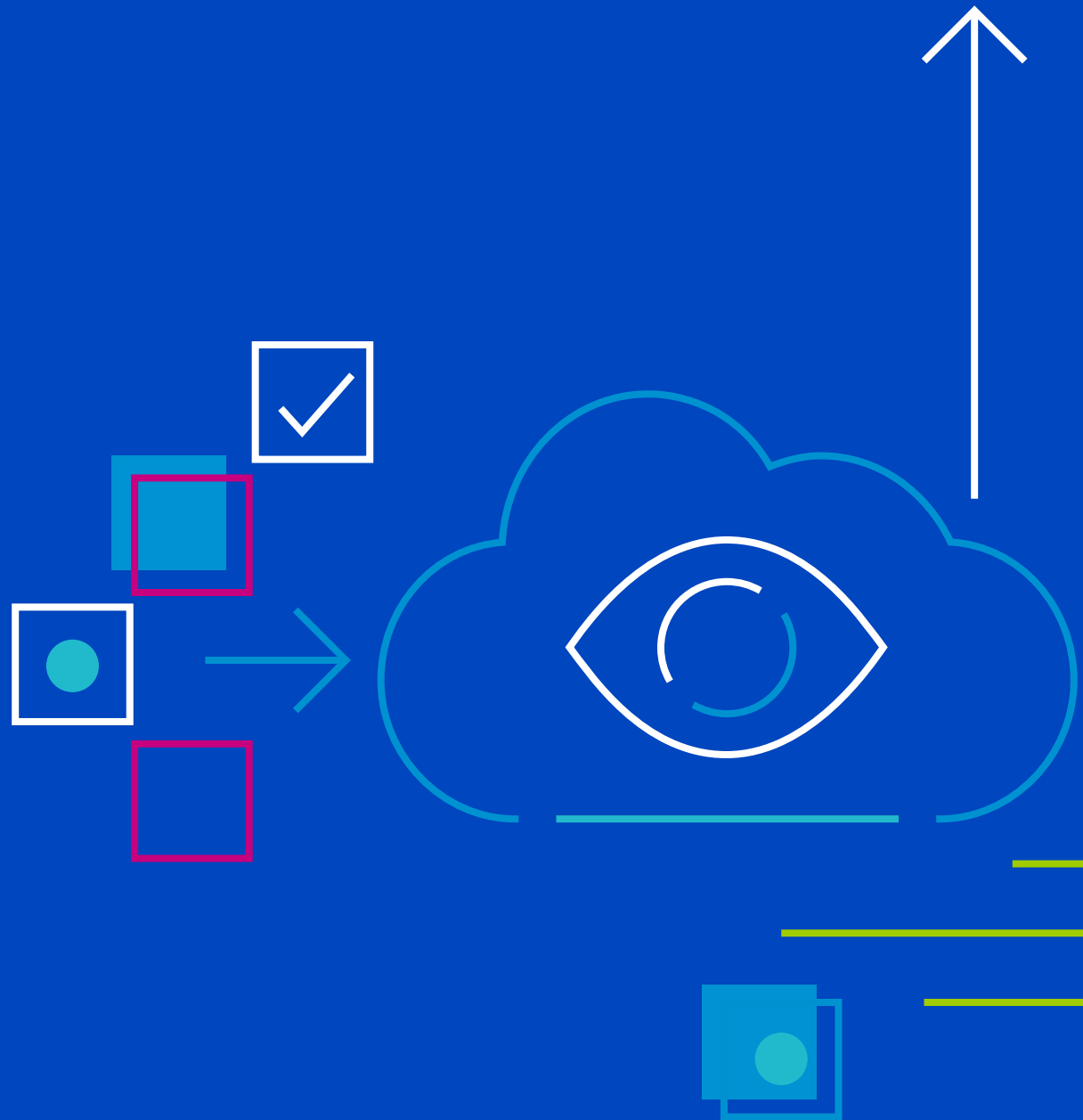


What modern applications need—observability

Digital transformation is driving a change to the way applications are architected and delivered. Many businesses are finding that the continuous delivery of modern applications, composed of an array of distributed services, microservices and containers causes applications to fail in new and unpredictable ways.

As businesses become more reliant on digital innovation to drive revenue, the need to manage these failures has given rise to a new requirement for modern applications: **observability**.

Creating an **observable system** helps ensure you have the data needed to identify the cause of unanticipated failure modes.





What the heck is observability, anyway?

“Observability is a measure of how well internal states of a system can be inferred by knowledge of its external outputs.”

According to Wikipedia

Monitoring

Can tell you if something is wrong

Observability

Can tell you what is wrong

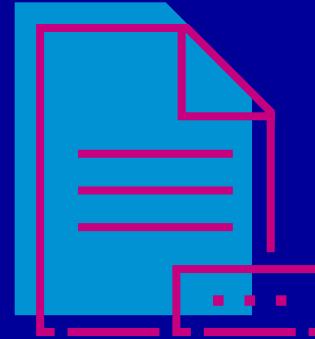
The three pillars of observability

According to author Cindy Sridharan in O'Reilly there are three pillars of observability: **logs**, **metrics**, and **traces**:

- **Logs** are an immutable, timestamped record of discrete events that happened over time.
- **Metrics** are a numeric representation of data measured over intervals of time.
- **Traces** represents a series events that describe the end-to-end request flow through a distributed system.

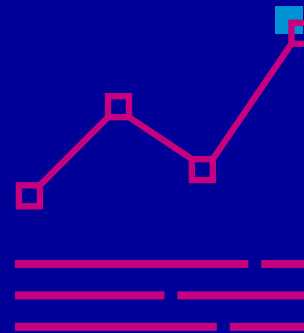
Individually, any of these external outputs provides a useful way to monitor a system. When brought together, analyzed and interrogated in real-time, these outputs make systems observable.

1



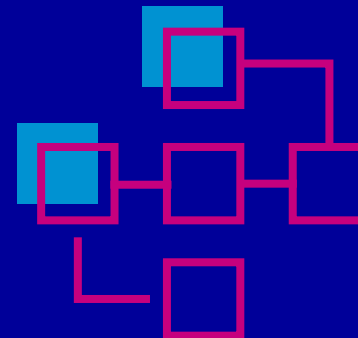
..... Logs

2



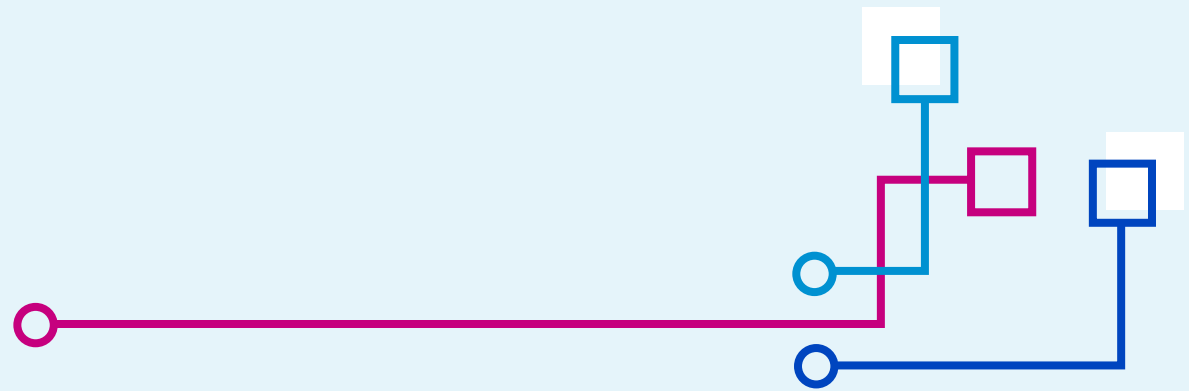
..... Metrics

3



..... Traces

Building a better observer experience



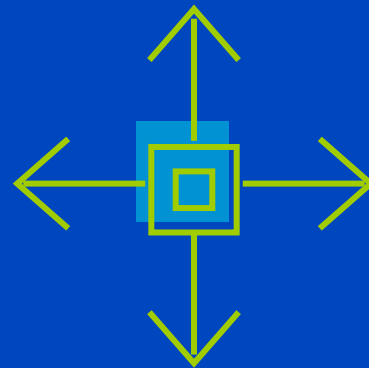
	Legacy monitoring	Modern observability
Technology and data model	closed and proprietary	open and extensible
Complex troubleshooting steps	humans handle	AI/ML handles
Management overhead	high	minimal
Data and analytics	per domain	across domains
Data is	dashboard centric	data relationship centric
Data ingestion	is constrained	is elastic
Data triaging	all data treated equally	important data handled with priority
Implementation and configuration	is tedious	is easy
Data security	customer's responsibility	built-in
Automation	reactive and difficult	proactive and prescriptive
Data	multiple, disjointed query languages	single, powerful flexible language



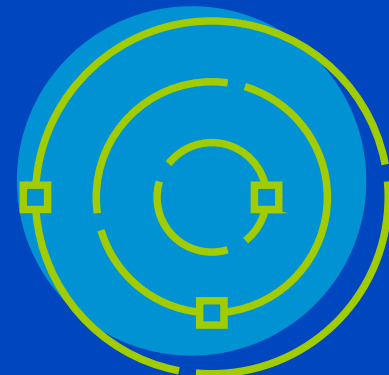
Why use observability?



Faster innovation



Improved reliability



Greater efficiency



Because of the volume, velocity and variety of data, modern observability solutions use AI/ML to sift through complex datasets to identify root-cause problems more efficiently, make recommendations and automate analyses

Artificial intelligence and machine learning in observability

Data deluge

Digital services emit huge amounts of data from every interaction with the customer. This data, coming in the forms of logs, metrics, and traces, is generated by infrastructure (servers, virtual machines, containers, cloud, etc.), applications (your code, databases, web servers, etc.), clients (web, mobile, etc.), and security technologies (endpoint, SSO, etc.).

To make sense of this data, artificial intelligence (AI) and machine learning (ML) are critical components of observability.

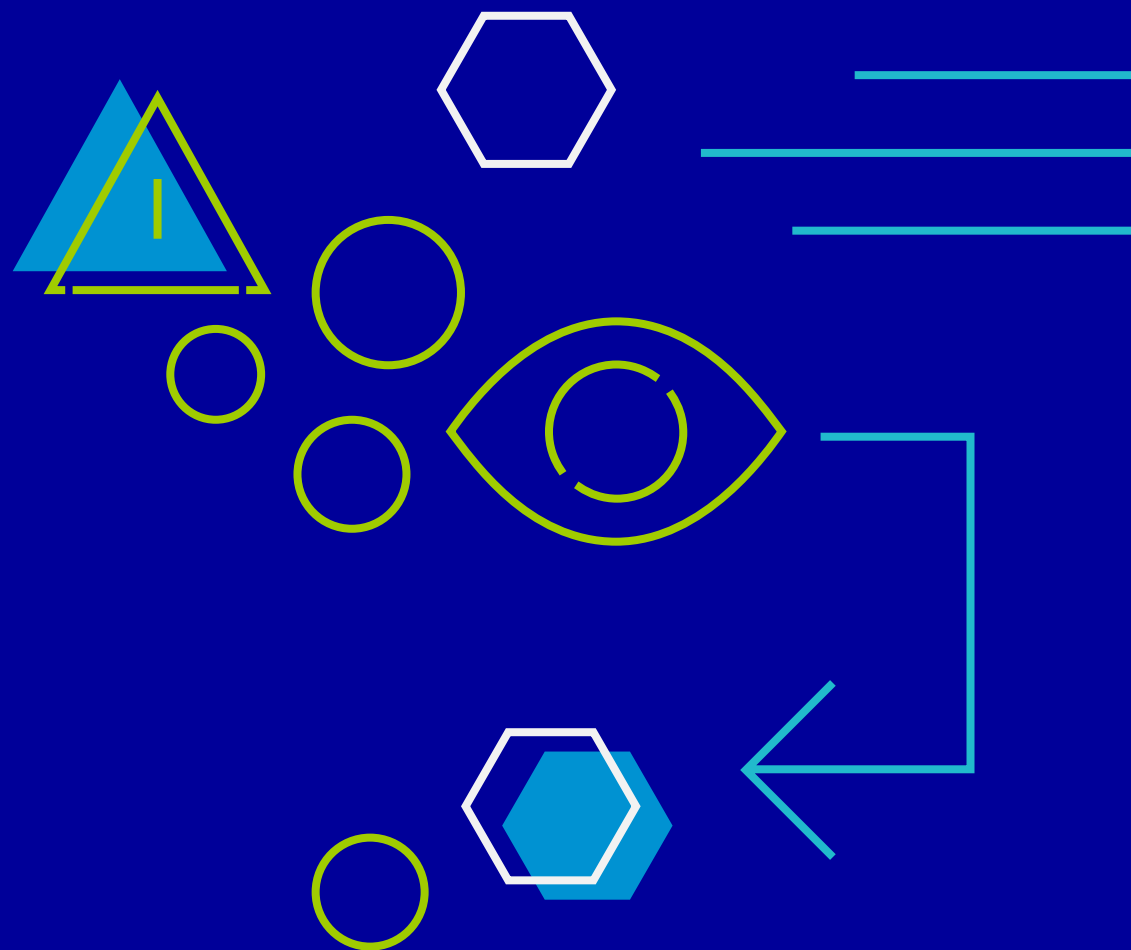
Collected and analyzed effectively, this data can dramatically improve your team's ability to support a high-quality customer experience.

Most organizations have already started their journey to observability by cobbling together a collection of open source and proprietary “point products.”

The more modern of these “point products” will have an ML/AI attached to it that is tuned to solve a specific use case within a specific domain, such as infrastructure monitoring, or application performance monitoring (APM).

This tuning means that it’s great at understanding one type of metric, but not necessarily good at understanding how those metrics might relate to other types of telemetry.

End-to-end visibility cannot be realized in silos. Not only is it impractical, but it is inefficient to get answers from dispersed data. Insights from telemetry must draw from cross-platform AI/ML algorithms that detect anomalies and connect dots, processing data in volumes far beyond the human brain’s capacity. For algorithms to perform to their potential, all data should be centralized.



“AIOps platform offerings fall into two categories: domain-agnostic and domain-centric solutions. However, requirements for increased flexibility for processing highly diverse datasets are having a significant impact on the market and shifting AIOps platforms toward domain-agnostic functionality.”

Gartner

Market Guide for AIOps Platforms

By Analysts Pankaj Prasad, Padraig Byrne, Josh Chessman

Published 6 April 2021



In his groundbreaking book, “The Myth of the Superhuman AI,” Kevin Kelly talks about the fact that intelligence is a “combinatorial continuum.”

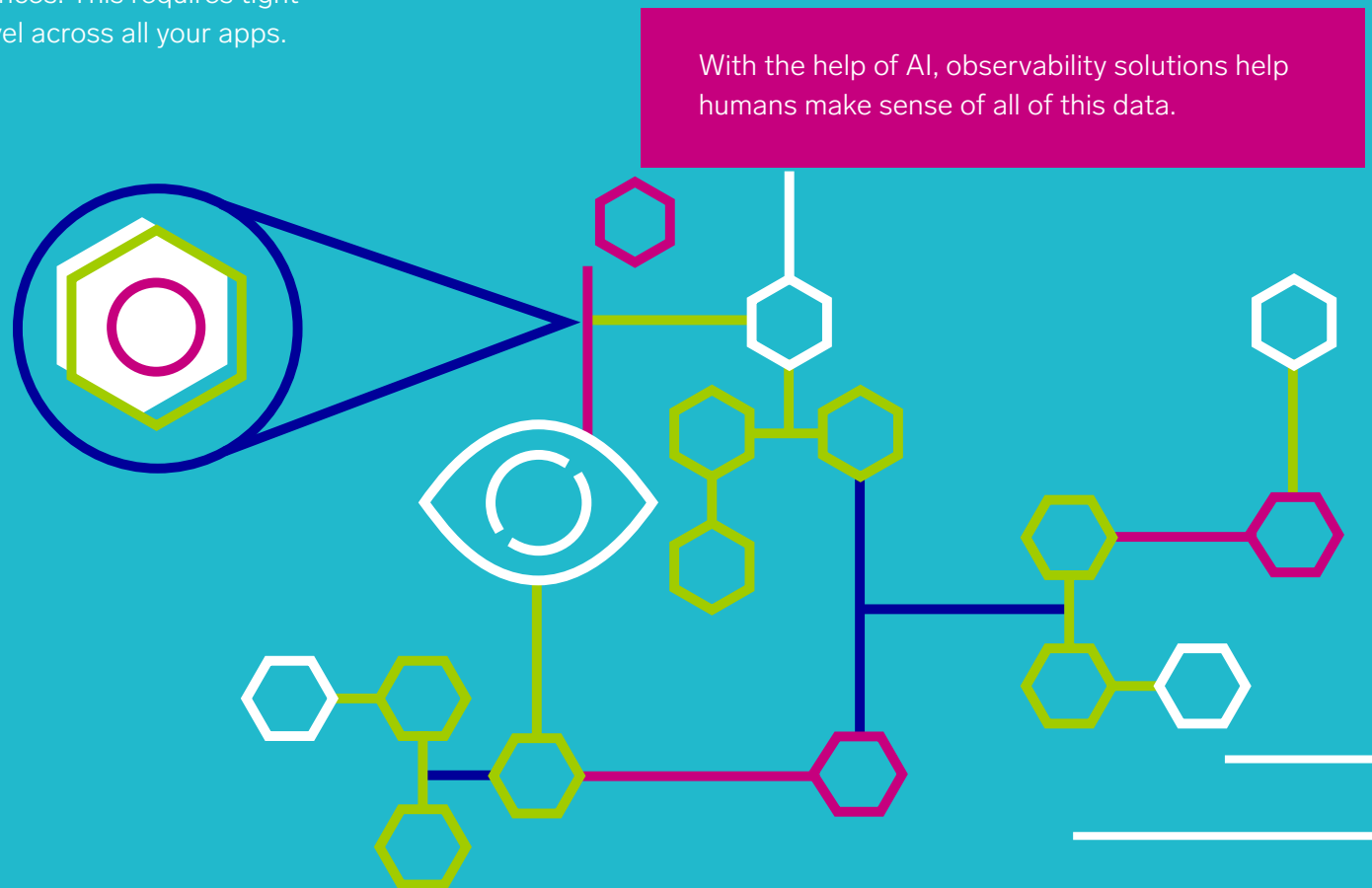
Put more simply, the more information you feed into an AI, the more relationships it will be able to find and the more intelligent it will become.

Organizations looking to maximize their observability efforts will be well-served with a single unified repository of telemetry data, accompanied by a domain-agnostic AI that can look across all the data with full fidelity to detect and identify patterns and relationships.

How do you make your applications observable?

To make your apps observable, you need to design them so they emit enough useful signals to describe their state. That needs to happen across your delivery chain, so the app is equally observable on a dev's laptop, in pre-prod and in production instances. This requires tight integration with something on the code level across all your apps.

Enter OpenTelemetry.



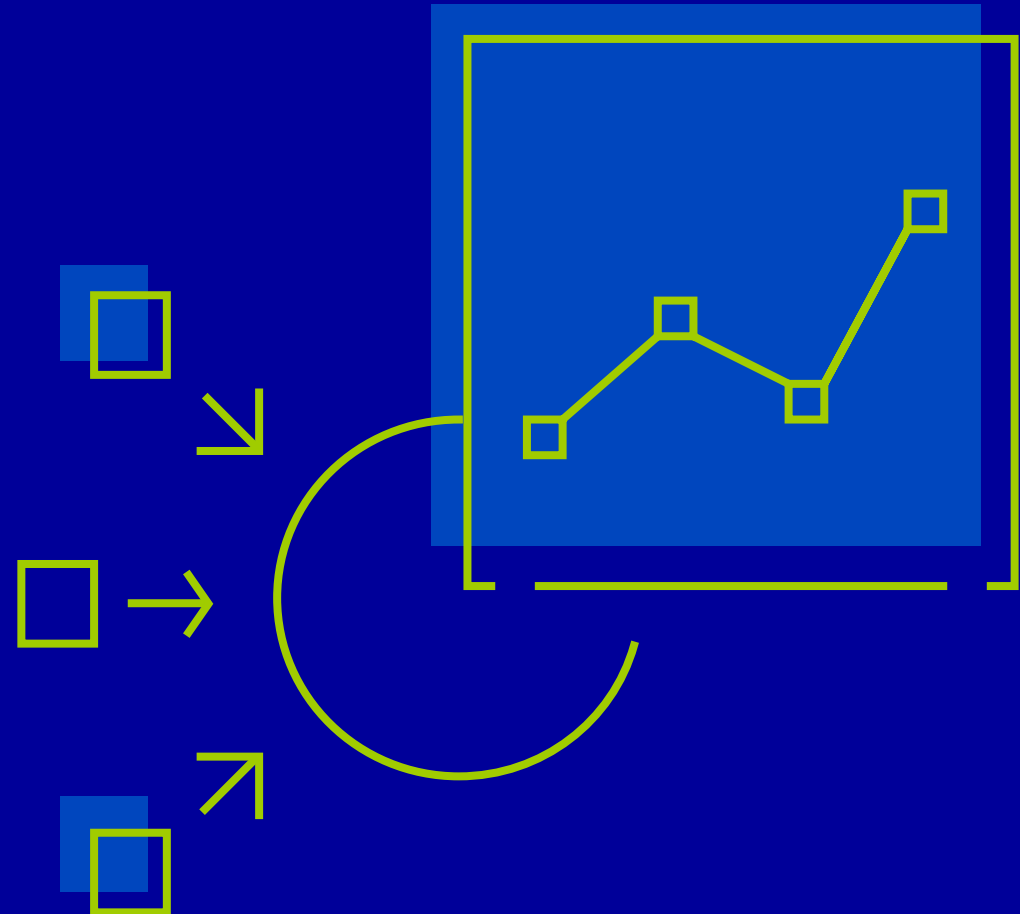
OpenTelemetry

Domain-agnostic analytics and a unified telemetry datastore don't amount to anything if there is no data being ingested.

OpenTelemetry is a collection of tools, APIs, and SDKs that organizations use to instrument, generate, collect, and export telemetry data (metrics, logs, and traces).

Openness drives innovation and choice, making it a good strategy for your observability journey.

OpenTelemetry is a CNCF incubating project with quite a history. It was born out of two other projects, OpenTracing and OpenCensus, both of which were working towards the common goal of developing an industry wide standard for software instrumentation. Both projects were solving the same problems and this led users to have to choose which to standardize on. To reconcile this conflict, in 2019 both projects merged to become OpenTelemetry.



Whither,
observability?

“You can’t transform something you don’t understand.

If you don’t know and understand what the current state of the customer experience is, how can you possibly design the desired future state?”

Annette Franz
Founder and CEO of CX Journey



Key metrics

While there is no single set of metrics that every business needs to drive observability operations, every organization should track four basic categories of metrics to achieve observability.

- Application metrics
- Infrastructure metrics
- Software delivery metrics
- Business metrics

Some, like application and infrastructure metrics, are obvious to most teams.

Application metrics

The most basic types of observability metrics are those associated with approaches like the RED Method or Google’s Four Golden Signals. They boil down to tracking performance

at the application level by monitoring:

- Request rates, which track how many requests the application receives in a given period.
- Latency, which measures how quickly the application responds to requests.
- Errors, which record how often a request results in an error.

When used for monitoring, these metrics help identify issues like an application that has become slow to respond or that is generating a higher-than-usual volume of errors.

In the context of observability, however, these metrics can be taken further to provide deeper insight into performance issues. For instance, if you can correlate

a spike in errors with an increase in request rates, it’s more likely than not that the application is generating errors because it is receiving more requests than it can handle. The solution in that case would probably be to spin up more instances of the application.

Real User Monitoring (RUM)

is a capability that helps observers understand how real users interact and experience a digital interface (web application, mobile application) and whether or not their experience is satisfactory. It is often used as a starting point for problem detection and diagnosis.

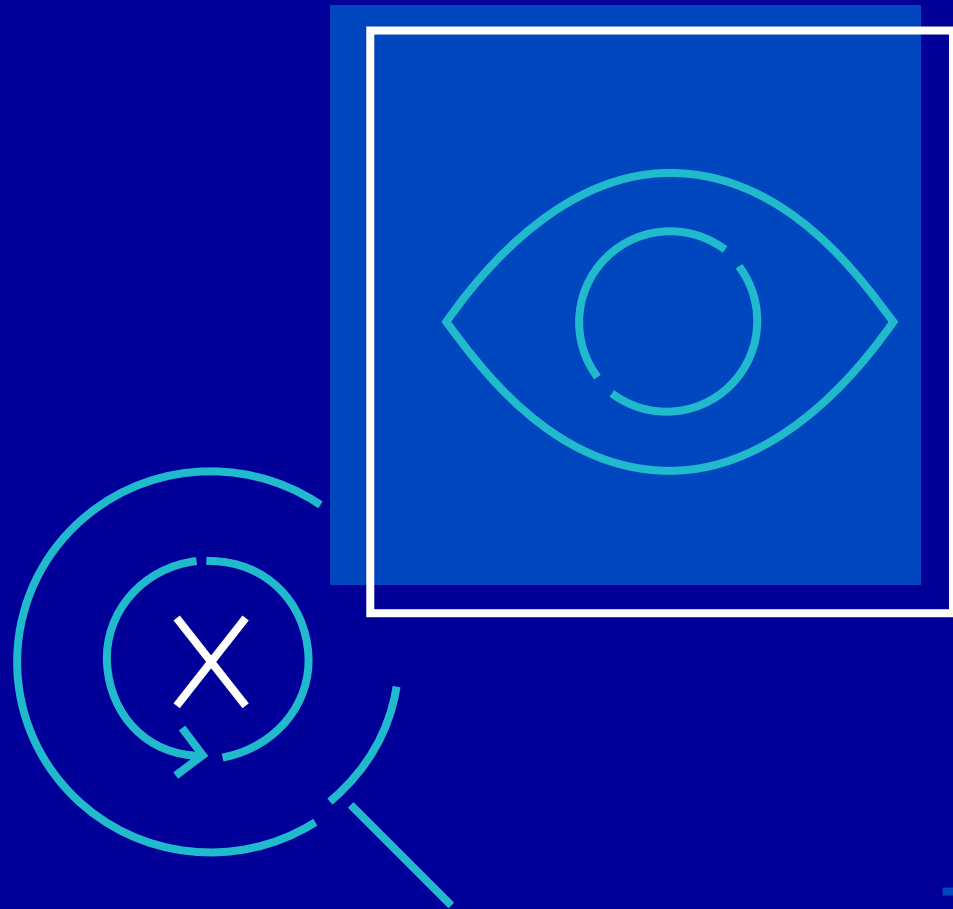
Infrastructure metrics

You can gain further observability insights by tracking infrastructure metrics. The exact metrics to work with here depend on how your application is hosted – whether it runs in the cloud or on-premises, for example, and whether it's orchestrated via Kubernetes. But in general, you'll want to track:

- CPU usage, based on total CPU availability.
- Memory usage, based on total memory availability.

In a distributed infrastructure like a Kubernetes cluster, you should also track the total number of nodes and changes in node state in order to ensure that you get ahead of issues such as a lack of available nodes.

To apply these metrics to observability, you should correlate them with other data points. An exhaustion of CPU and memory resources that occurs at the same time as an increase in application error rates may mean that the application is dropping requests due to lack of infrastructure resources, for example.

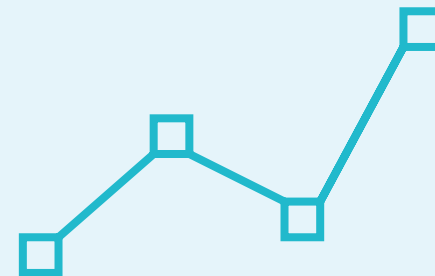


Software delivery metrics

Delivering quality software quickly is a key predictor of an organization's productivity, profitability, and customer satisfaction. However, siloed tool data blurs the view of how your organization is operating from a software delivery and DevOps perspective.

Observability is the backbone of your continuous integration (CI) pipelines to ensure the health, performance and reliability of the applications at each phase of the software delivery process. The [2021 DORA State of DevOps Report](#) highlights five metrics that are key indicators of organizational software delivery and operational performance:

Deployment frequency	How often does your organization deploy code to production?
Lead time for changes	The time to go from code committed to code running in production.
Time to restore service	The time to restore service when an incident or defect occurs.
Change failure rate	A percentage of changes to production resulting in degraded service.
Reliability	Degree to which the software operates reliably—measured by SLIs, SLOs, and error budgets.



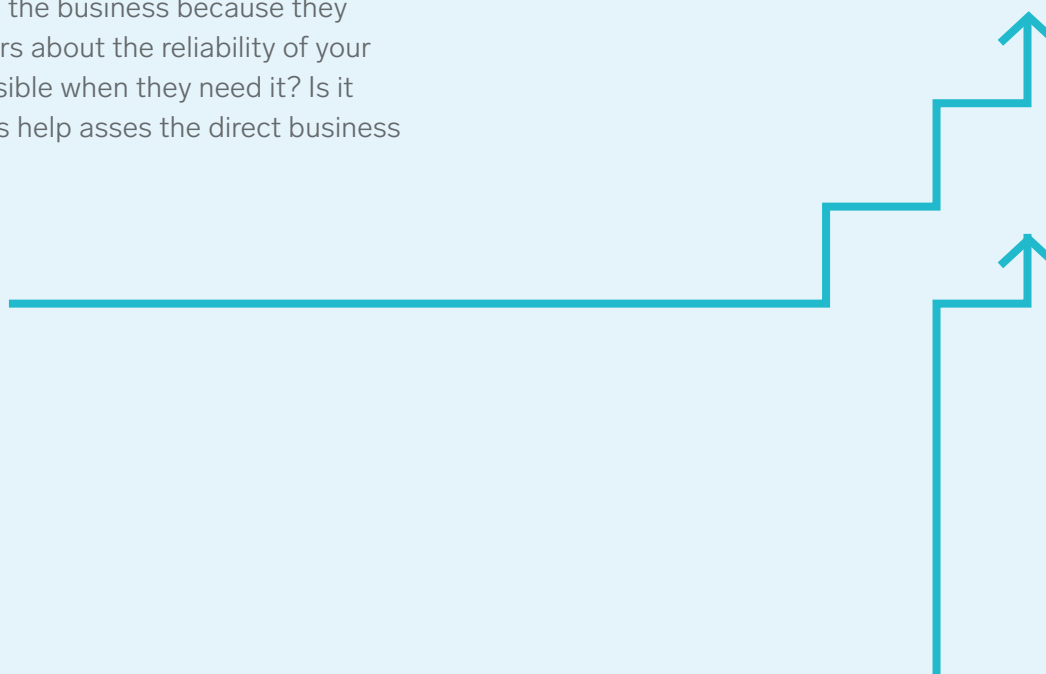
Finally, consider tracking business metrics, meaning metrics that align technical goals with business goals. Examples include:

- Additionally, metrics such as SLAs, SLIs, and SLOs, while critical in software delivery, are also important to the business because they represent the promise you make to users about the reliability of your service. Will your application be accessible when they need it? Is it performing as expected? These metrics help assess the direct business impact of observability efforts.

The formal agreement between you and your customers about the performance of your service.

The measurements that define your service's performance in support of the SLA, i.e., availability or meantime to respond.

How your service is really performing, i.e. are you keeping your promise to your customers, outlined in your SLAs?



Your people stack is as critical as your tech stack

Data silos and hoarding of information have no place in organizations leaning into observability.

Make sure all stakeholders become “observers,” who can access software that helps them track observability metrics. Avoid allowing any one team to “own” a particular tool or the data it generates. When observability and the insights it delivers are shared across the team, it becomes much easier to establish a feeling of shared responsibility—people are collectively invested in understanding the whys of something not working, than simply that it is not working. As this mentality pushes further left, teams begin to design, build, release—and resolve—collectively, spurring greater efficiency, reliability, and even innovation.

Observability begs a specific type of culture

Teams can identify helpful practices that foster information flow and trust by examining the six aspects of [Westrum's model of organizational culture](#), focusing on those behaviors seen in the generative culture:

- High cooperation
- Messengers are trained
- Risks are shared
- Bridging is encouraged
- Failure leads to inquiry
- Novelty is implemented

Observability in practice

Here is how teams are using observability to make meaningful differences in their business. See more examples of companies using observability to drive innovation and digital service reliability [here](#).



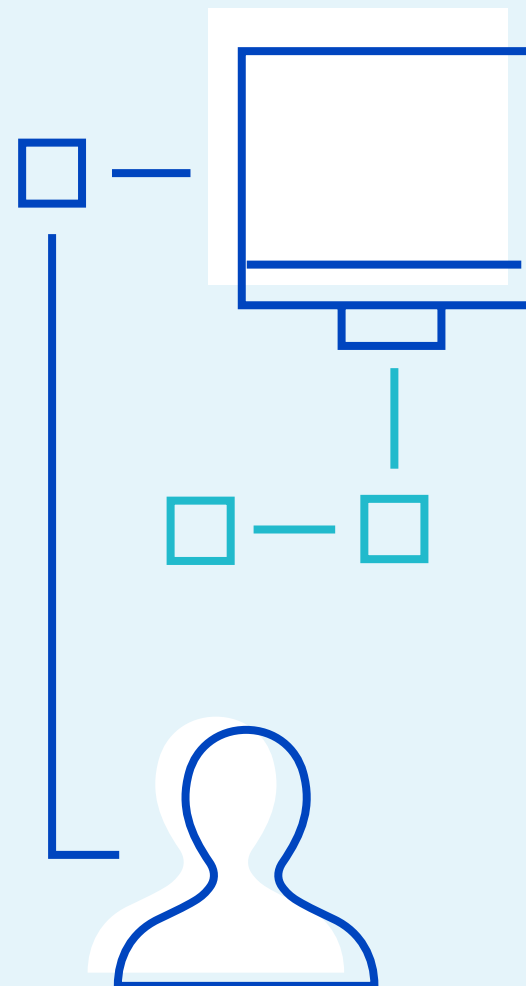
To keep its more than eight million customers happy, LendingTree needed a unified approach to understanding the state of its distributed environment. The team needed to surface issues, regardless of their origin, to resolve them more quickly.

LendingTree saved millions of dollars each year by deploying Sumo Logic to gather and analyze machine data to reduce the number and severity of service disruptions.



At any given moment, Medidata's customers are managing more than 5,000 active clinical trials on its platform. To increase the quality and availability of its digital offerings, management expanded access to its machine data.

Medidata leverages Sumo Logic's API to uphold SLA commitments by creating a resource that crawls through accumulated machine data and inspects it for load, latency, and throughput. It then notifies relevant users should it detect deviation from established SLA commitments, to make sure it upholds its promises to customers.



Why Sumo Logic?

Sumo Logic's full-stack observability provides:



Unified application observability

Adaptive analytics

Analytics on all structured and unstructured application data including transaction traces, logs and metrics.

Powerful query language and user-friendly UX

Explore all data and find the unknown unknowns using pattern detection with AI/ML.

Open Source Collection

Works with OpenTelemetry, Telegraf, and other Open Source tools.



Simple and transparent licensing

Price transparency

Eliminates license waste and on-demand overage bills.

Simplicity

Single licensing model—no need to license every product or functionality.

Flexible data tiering

Cost-effectively index all your data without sacrificing performance on our secure and compliant platform.



Most comprehensive security portfolio

Cloud Security Monitoring & Analytics

Out-of-the-box apps and integrations for security and compliance content, for both cloud and on-premise data sources.

Cloud SIEM

Automatically triage alerts and correlate threats across your entire environment.

Cloud SOAR

Take your security to the next level with security orchestration, automation and response.

About Sumo Logic

Sumo Logic Inc., (NASDAQ: SUMO) is the pioneer in continuous intelligence, a new category of software, which enables organizations of all sizes to address the data challenges and opportunities presented by digital transformation, modern applications, and cloud computing. The Sumo Logic Continuous Intelligence Platform™ automates the collection, ingestion, and analysis of application, infrastructure, security, and IoT data to derive actionable insights within seconds. More than 2,100 customers around the world rely on Sumo Logic to build, run, and secure their modern applications and cloud infrastructures. Only Sumo Logic delivers its platform as a true, multi-tenant SaaS architecture, across multiple use-cases, enabling businesses to thrive in the Intelligence Economy.



To learn how Sumo Logic Application Observability solution
can accelerate your DevOps processes, visit:

<https://www.sumologic.com/solutions/application-monitoring/>