A Beginner's Guide to **Observability**

Taking the deep dive into what your systems, services and apps are really doing

splunk>

turn data into doing™

Observability has been called everything from a trendy tech buzzword to a "monitoring-on-steroids" must-have. The truth is more involved — especially given the increased complexity of the modern infrastructure and the undisputed need for better monitoring higher in the stack, and deeper in the system.

Teams requiring operational visibility have expanded beyond sysadmins and IT Ops analysts — even developers are taking greater ownership of knowing what's going on for a better customer experience. To effectively do this, all roles need visibility inside their entire architecture — from third-party apps and services to their own — to fix and eventually prevent problems. When that capability is built-in — the premise of observability — it not only makes visibility easier, enables greater insight and leaves more time for more strategic initiatives, but it's also critical to the overall success of Site Reliability Engineering (SRE). This provides a bridge between developers releasing code and operators maintaining infrastructure impacted by code. On top of this, it shifts some of the monitoring workload onto development.

In this guide, we'll define what observability is and what it takes to achieve it. We'll also give some examples of observability in action and guidance for what to look for in a solution to help your organization achieve observability.

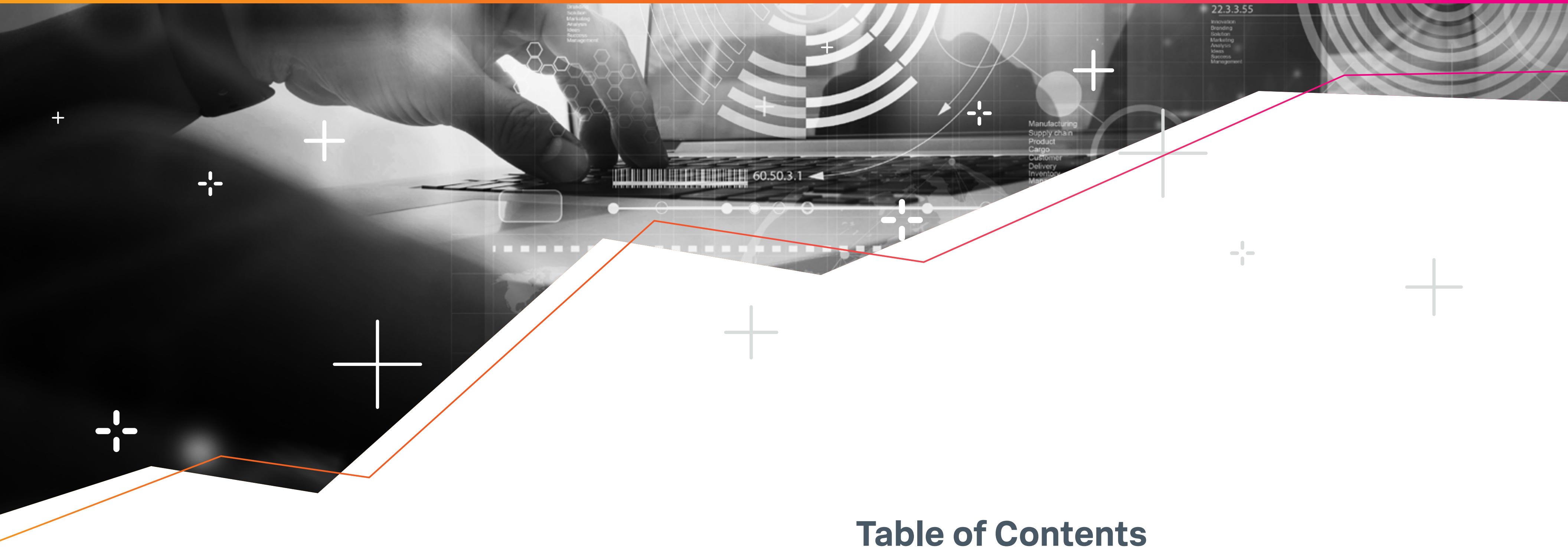


Table of Contents

Visibility feedback METRICS INSTRUMENTATION

exception tracking CONTROLLABILITY EVENTS

operational intelligence monitoring logs

externalize its state

AIDevOps TRACING DIGITAL EXHAUST availability

OBSERVABILITY

CHAPTER 1

Observability — What It Is and Isn’t

INTRODUCTION

Building in Feedback

Simply defined, observability is instrumenting systems and applications to collect metrics and logs. It’s building apps with the idea that someone is going to watch them. It comes from system control theory, the foundation of feedback systems, where observability is a measure of how well the internal states of a system can be inferred from knowledge of its external outputs1—a kind of digital exhaust. Think of it as a property of a system—another attribute, like functionality, performance or testability.

> “You can monitor a system using various instrumentation. But if the system doesn’t externalize its state well enough that you can figure out what’s actually going on in there, then you’re stuck.”[2]

[1] “Observability,” Wikipedia, 2018.

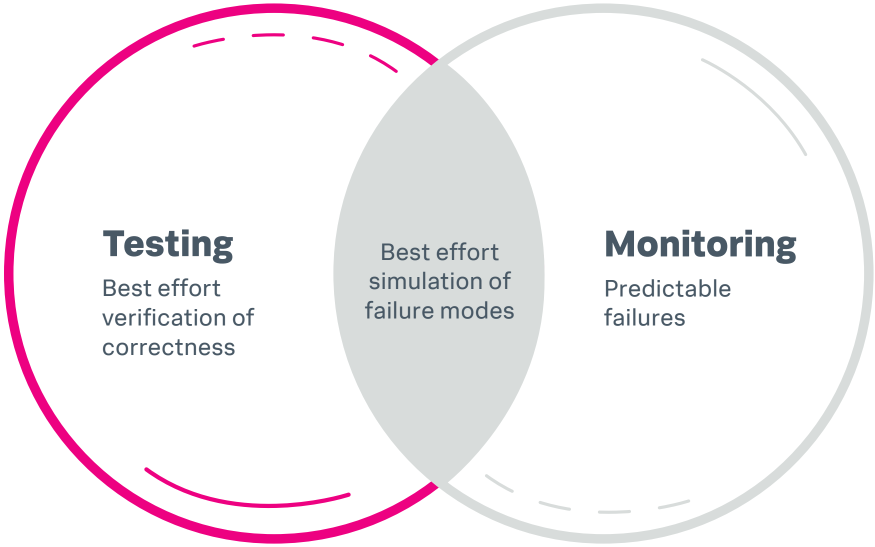
[2] Ernest Mueller, “Monitoring and Observability,” agileadmin.com, February 2018.

Observability vs. Monitoring

Monitoring	Observability
Tells you whether the system works	Lets you ask why it's not working
A collection of metrics and logs about a system	The dissemination of information from that system
Failure-centric	Understands system behavior regardless of an outage
Is "**the how**" / Something you do	Is "**the goal**" / Something you have
I ***monitor*** you	You ***make yourself*** observable

Observability

All possible permutations of full and partial failure


Testing
Best effort verification of correctness
Best effort simulation of failure modes
Monitoring
Predictable failures

Observability as a Culture

Observability isn't a substitute for monitoring; they're complementary. But it's nearly impossible to have effective monitoring without a culture of observability. Tools are not enough, and none are going to magically "give you" observability.

The Value of Observability

- Planning and development
- Problem-solving
- Powering more useful incident reviews
- Improving uptime and performance

Observability as a culture is the degree to which a team or company values the ability to inspect and understand systems, their workload and their behavior. Companies that have a strong observability culture often have observability teams, although they may not be named as such.

> "If observability is a cultural value, the results will come."
>
> —**Andi Mann,** Splunk Chief Technology Advocate

CHAPTER 2

Roadmap to Observability

Pillars of Observability

With a clearer handle on what observability is, the next step is achieving it. The following three pillars are critical to achieving observability:

Events

Immutable record of discrete events that happen over time

Metrics

Numbers describing a particular process or activity measured over intervals of time

Tracing

Data that shows which line of code is failing to gain better visibility at the individual user level for events that have occured

Modern Event-Handling Techniques

There are three techniques used in handling events with the end goal of shared insights, a collaborative response, data-enabled IT and intelligent operations.

Collect all relevant data

This allows complete visibility across stacks, technologies and all environments:

- Cloud-native
- Traditional, on-premises, monolithic, etc.
- Hybrid environments

De-spam

Separate valuable signals from the noise

Add Context

Prioritize resolution to ensure service availability, to provide business detail and to enhance ITSI service insights

Role of AI and ML

The volume, velocity and variety of the data that is being collected is fundamentally unmanageable by humans. Observability allows the questions to be asked and the systems to manage themselves, using artificial intelligence (AI) and machine learning (ML) for the sophisticated analytics.

Learning algorithms can understand the past health of your services and applications to predict what's going to happen in the future. Data will enable you to apply ML, a subset of AI, to the historical and real-time data you've collected and use it to help predict high-likelihood, potential future events and truly harness the power of AI to achieve prediction.

As AI becomes more enmeshed with DevOps tools and systems, doing this type of analysis will become easier over time, providing continuous and deeper insights and achieving a more agile and productive state in IT.

Sophisticated Event Analytics

Advances in event analytics enable the following:

- Reduce event clutter and false positives with multivariate anomaly detection
- Automatically conceal duplicate events to focus on relevant ones
- Easily sift through vast amounts of events by filtering, tagging and sorting
- Enrich and add context to events to make them informative and actionable

> “Manage the incident, not the event.”

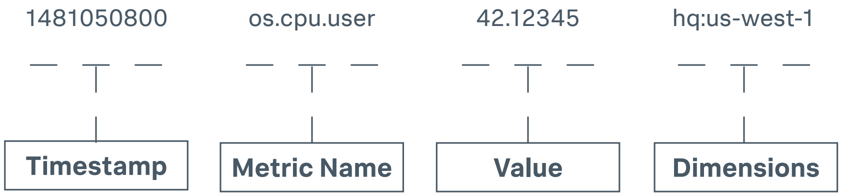
The Metrics That Matter

Analysts including Gartner, Forrester, IDC and Computing UK have all developed their own set of “Metrics that Matter.” The following is a list of observable metrics and events that we’ve found to be critical for achieving full observability.

Metrics

Common metrics sources include:

- System metrics (CPU, memory, disk)
- Infrastructure metrics (AWS CloudWatch)
- Web tracking scripts (Google Analytics)
- Application agents (APM, error tracking)
- Business metrics (revenue, customer signups, bounce rate, cart abandonment)



1481050800 os.cpu.user 42.12345 hq:us-west-1

Timestamp Metric Name Value Dimensions

Events

Events come in three forms — plain text, structured and binary. Common event sources include:

- System and server logs (syslog, journald)
- Firewall and intrusion detection system logs
- Social media feeds (Twitter, etc.)
- Application, platform and server logs (log4j, log4net, Apache, MySQL, AWS)

Collecting Observability and Monitoring Data

The good news is that so much data exists; the challenge is aggregating all of it. The following lists the type of data sources that have evolved over the years — all important in achieving observability.

Existing Sources

- Network flow data
- Virtual servers – VC Logs, ESXi Logs, etc.
- Cloud services – AWS data sources such as EC2, EMR, S3, etc.
- Docker – logging driver, syslog, apps logs, etc.
- Containers & MSAs – container and microservices logs, container metrics and events, etc.
- Third-party services – SaaS, FaaS, Serverless, etc.
- Control systems – vCenter, Swarm, Kubennetes, etc.
- Dev automation – Jenkins, Sonarcube
- Infra orchestration – Chef, Puppet, Ansible
- Signals for security analytics – DLP, device telemetry, metadata
- Signals from mobile devices – product adoption, users and clients, feature adoption, etc.
- Metrics for business analytics – app data, HTTP events, SFA/CRM
- Signals from social sentiment analytics – analyzing tweets over time
- Customer experience analytics – app logs, business process logs, call detail records, etc.
- Analytics for service intelligence – ER visits, treatment wait time, RXs, etc.
- Message buses and middleware

Newer Sources

- collectd – a daemon that collects metrics
- statsd – a daemon that listens for statistics
- fluentd – a daemon that unifies log data collection
- Signals from modern services, e.g., Splunk
- Zipkin, Jaeger – a back-end distributed tracing system of choice
- Semantic logging – creating new signals

These daemons send metrics to a defined location vs. observability, which creates and defines the metrics that matter and will drive action when those metrics are out of limits.

Your code isn't "done" until you've built analytics to support it.

Without building in this kind of visibility, you're unable to determine why a system has failed, which slows response and resolution time to business-critical issues.

CHAPTER 3

Observability in Action

INTRODUCTION

Now that we've talked about what observability is, why it's important, and the metrics and events it's based on, let's see observability in action and the IT and business benefits it can help achieve.

The following case studies show actual customer results using Splunk software to visualize and correlate events and metrics, as well as troubleshoot and remediate issues that arose within the data sources. Modern IT organizations experience tremendous benefits in both key areas of Operations and Dev Logging.

Operations

Cox Automotive – To achieve better operational visibility into its on-premises and online auction platform so it could pinpoint, troubleshoot and prevent issues in real time, Cox Automotive deployed Splunk IT Service Intelligence (ITSI), Splunk Enterprise and Splunk Cloud. The company has seen the following benefits:

- Increased simulcast reliability boosts bottom line
- Significantly improved mean-time-to-identify (MTTI)
- Reduced number of auction incidents by 90%
- Enabled more proactive equipment replacement: KPIs predict outages and provide real-time degradation monitoring

Operations

ENGIE – This energy trading platform needed a centralized, real-time view of the health of its critical trading application to speed troubleshooting and ensure performance. Since deploying Splunk Enterprise and Splunk IT Service Intelligence (ITSI), the company has seen the following benefits:

- Holistic view of health of key business services
- Faster resolution of business-impacting issues
- Improved collaboration between development and infrastructure teams during incident triage

Operations

TransUnion – This financial services company needed to establish the baseline for external customer traffic and customer volume transactions. Since adopting Splunk Enterprise, Splunk IT Service Intelligence (ITSI) and the Splunk Machine Learning Toolkit for enterprise IT monitoring, the company has gained the ability to:

- Provide reliable transactions and meet customer SLAs
- Monitor, forecast and maintain transactions in real time based on machine data
- Discover incident root causes in minutes instead of hours
- Reduce the number of false alerts
- Increase revenue by improving transaction processing

Dev Logging

Hyatt – This global hospitality company needed a centralized solution to monitor and troubleshoot server issues and improve application delivery. Since implementing Splunk Enterprise and the Machine Learning Toolkit, the Hyatt developer team has experienced benefits including:

- Faster mean-time-to-resolution (MTTR) from hours or minutes to real time with instantaneous visibility across the entire business
- Increased developer productivity
- Improved customer experience from proactive monitoring

Dev Logging

Yelp – To ensure a great customer experience for the millions of people it connects with local businesses through its website and mobile app, Yelp standardized on Splunk data analytics platform, enabling hundreds of technical and non-technical users—from site reliability engineers to product managers — to gain actionable business insights. Since deploying Splunk, Yelp has seen benefits including:

- Improving website uptime with real-time notifications
- Quickly and reliably delivering application features to users
- Uncovering business insights and improving the customer experience
- Saving engineering hours by liberating the data for all users

Dev Logging

FamilySearch – This genealogy organization needed a way to move to a continuous delivery model, manage its all-in migration to Amazon Web Services (AWS), and immediately troubleshoot website errors. Since deploying Splunk, the organization has seen benefits including:

- Successful migration from monthly releases to over 900 deploys per day
- Ability to re-allocate 12 developers to more value-added tasks
- Visibility into the AWS environment to support AWS migration strategy

CHAPTER 4

Options in Observability

INTRODUCTION

As the need and demand for observability grows, some monitoring tool vendors are jumping on the bandwagon — about as fast as they did with DevOps a few years ago. No tool is going to “give you” observability. That’s a silver-bullet fallacy that should be a red flag as you research your options.

Many vendors claim to have full observability capabilities, but upon closer inspection, they only offer a portion of the observability picture. Providing partial views is only a component of observability, which, by definition, fails at achieving it.

Observability Built for New IT

As organizations start to implement observability, there's an increasing focus on new IT — one with a DevOps (i.e. CI/CD) operating model. An ideal observability solution has attributes that support the following three pillars, which define the new direction IT is heading:

Monitor

to get centralized visibility into all relevant infrastructure and applications. Detect root cause analysis with context, creating a culture of observability to prevent — and eventually predict — issues from recurring.

Collaborate

across organization silos and share the data that matters with every stakeholder to manage incident resolution and build repeatable action plans.

Automate

the mundane, orchestrate the complex. Automate processes to save time to focus on what really matters — like strategic business initiatives — rather than just monitoring and troubleshooting. Eventually, orchestrate multiple processes to optimize human intervention in monitoring, root cause analysis and remediation.

Splunk for NEW IT

Splunk for IT supports teams as they evolve from reactive to proactive to predictive IT, providing solutions throughout the cycle of a problem to help organizations create a culture of observability and make data-driven decisions. Splunk for IT provides a closed-loop approach to help IT teams centralize all of their most actionable data, focus on the insights that matter most, and then continuously improve their monitoring, incident response and incident resolution strategies.

By applying the lessons learned from root cause analysis, IT teams can refine their systems to avoid repeating the same issues over and over, and allow them to focus on innovating.

The majority of our customers start their journey with Splunk doing reactive troubleshooting, simply looking through metrics, logs and traces to find the root cause of a problem. This is the world of Splunk Enterprise, allowing you to index data from many sources and search through it. As customers move up in maturity, they want to monitor their data, organize this data into services, and eventually predict issues and prevent them from recurring. This is the world of Splunk IT Service Intelligence (ITSI) — Splunk's answer to monitoring services. Customers that want to take action on the issues identified throughout monitoring can use Splunk with VictorOps, which enables Splunk users to collaborate in a virtual setting to resolve problems as quickly as possible.

IT creating business value

With Splunk, organizations are able to:

- Discover and create actionable value from their digital exhaust
- Maximize the value of IT resources
- Shorten the time to decision so they can act faster and smarter when solving IT problems
- Take a new approach that allows IT to become value creators for the business by delivering timely and trusted data-driven decisions

Conclusion

The hype of observability is well earned. It allows Ops to take greater ownership of uptime and performance, and it requires an organizational culture of observability to succeed. Observability provides end-to-end visibility across your system so you have outcomes that are quantifiable. This allows IT to more quickly fix and eventually prevent problems, leaving more time for strategic initiatives. The best way to achieve observability is an approach that aligns with the new direction of IT, following the cycle of a problem through monitoring, collaborating and automating, aided by AI and ML. Customers who have achieved observability using Splunk have achieved a wide range of measurable business results and have allowed IT to become value creators for the business.

Splunk Insights for Infrastructure (SII) is the first step to achieving observability in your IT infrastructure.

Get Started for Free