

[Log In](#)

[Join](#)

[Back To Module Home](#)

Basic Building Blocks for Modern System Design

0% completed

Introduction to Building Blocks

Domain Name System

Load balancers

Cache


Databases

Key-value Store

Content Delivery Network (CDN)

Sequencer

Distributed Monitoring



- System Design: Distributed Monitoring
- Introduction to Distributed Monitoring
- Prerequisites of a Monitoring System

Distributed Cache

Distributed Messaging Queue

Pub-sub

Rate Limiter

Blob Store

Distributed Search

Distributed Task Scheduler

Sharded Counters

Conclusion

Mark Module as Completed

System Design: Distributed Monitoring

Learn about the importance of monitoring in a distributed system, and explore our high-level plan for designing it.

We'll cover the following

- Monitoring
- How will we design a distributed monitoring system?

Monitoring#

The modern economy depends on the continual operation of IT infrastructure. Such infrastructure contains hardware, distributed services, and network resources. These components are interlinked in such infrastructure, making it challenging to keep everything functioning smoothly and without application downtime.

Figuring out the issue

It's challenging to know what's happening at the hardware or application level when our infrastructure is distributed across multiple locations and includes many servers. Components can run into failures, response latency overshoot, overloaded or unreachable hardware, and containers running out of resources, among others. Multiple services are running in such an infrastructure, and anything can go awry.

When one of the services goes down, it can be the reason for other services to crash, and as a result, the application is unavailable to users. If we don't know what went wrong early, it could take us a lot of time and effort to debug the system manually. Moreover, for larger services, we need to ensure that our services are working within our agreed service-level agreements. We need to catch important trends and signals of impending failures as early warnings so that any concerns or issues can be addressed.

Monitoring helps in analyzing such complex infrastructure where something is constantly failing. Monitoring distributed systems entails gathering, interpreting, and displaying data about the interactions between processes that are running at the same time. It assists in debugging, testing, performance evaluation, and having a bird's-eye view over multiple services.

How will we design a distributed monitoring

system?#

We've divided the distributed monitoring system design into the following chapters and lessons:

1. Distributed Monitoring

- I. **Introduction to Distributed Monitoring:** Learn why monitoring in a distributed system is crucial, how costly downtime is, and the types of monitoring.
- II. **Prerequisites for a Monitoring System:** Explore a few essential concepts about metrics and alerting in a monitoring system.

2. Monitoring Server-side Errors

- I. **Designing a Monitoring System:** Define the requirements and high-level design of the monitoring system.
- II. **A Detailed Design of the Monitoring System:** Go into the details of designing a monitoring system, and explore the components involved.
- III. **Visualize Data in a Monitoring System:** Learn a unique way to visualize an enormous amount of monitoring data.

3. Monitor Client-side Errors

- I. **Focus on Client-side Errors:** Get introduced to client-side errors and why it's important to monitor them.
- II. **Design a Client-side Monitoring System:** Learn to design a system that monitors the client-side errors.

In the next lesson, we'll look at why monitoring is essential in a distributed system through an example. We'll also look at the downtime cost of failures and monitoring types.

[Back](#)

Unique IDs with Causality

Next

Introduction to Distributed Monitoring

Mark as Completed

Report an Issue