

Log In

Join

Back To Course Home

Grokking Modern System Design Interview for Engineers & Managers

0% completed

System Design Interviews

Introduction

Abstractions

Non-functional System Characteristics

Back-of-the-envelope Calculations

Building Blocks

Domain Name System

Load Balancers

Databases

Key-value Store

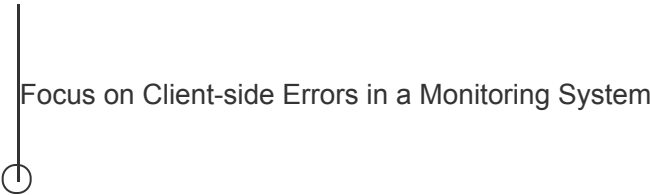
Content Delivery Network (CDN)

Sequencer

Distributed Monitoring

Monitor Server-side Errors

Monitor Client-side Errors



Design of a Client-side Monitoring System

Distributed Cache

Distributed Messaging Queue

Pub-sub

Rate Limiter

Blob Store

Distributed Search

Distributed Logging

Distributed Task Scheduler

Sharded Counters

Concluding the Building Blocks Discussion

Design YouTube

Design Quora

Design Google Maps

Design a Proximity Service / Yelp

Design Uber

Design Twitter

Design Newsfeed System

Design Instagram

Design a URL Shortening Service / TinyURL

Design a Web Crawler

Design WhatsApp

Design Typeahead Suggestion

Design a Collaborative Document Editing Service / Google Docs

Spectacular Failures

Concluding Remarks

Course Certificate

Mark Course as Completed

Focus on Client-side Errors in a Monitoring System

Learn what client-side errors are and their impact on the service.

We'll cover the following

- Client-side errors
- Failures due to a routing bug

Client-side errors#

In a distributed system, clients often access the service via an HTTP request. We can monitor our web and application servers' logs if a request fails to process. If multiple requests fail, we can observe a spike in internal errors (error 500).

Those errors whose root cause is on the client side are hard to respond to because the service has little to no insight into the client's system. We might try to look for a dip in the load compared to averages, but such a graph is usually hard. It can have false positives and false negatives due to factors such as unexpectedly variable load or if a small portion of the client population is affected.

There are many factors that can cause failures that can result in clients being unable to reach the server. These include the following:

- Failure in DNS name resolution.
- Any failure in routing along the path from the client to the service provider.
- Any failures with third-party infrastructure, such as middleboxes and content delivery networks (CDNs).

Server-side errors versus client-side errors

Failures due to a routing bug#

Let's look at a real-world example of an error that impacted a large number of service customers, but the service wasn't readily aware of it.

One of Google's peer ISPs accidentally announced Internet routes that it wasn't supposed to. As a result, the traffic of many of Google's customers started routing through unintended ISPs and wasn't reaching Google. Clients were frustrated because they weren't able to reach Google, while Google might have been unaware of such problems right away because these issues didn't happen on its infrastructure.

We can learn more about this event by clicking **here**.

BGP leak

The above leak isn't unique. Similar issues keep arising. Another such leakage happened on April 16, 2021, when an AS mistakenly announced over 30,000 BGP prefixes. This resulted in a 13 times spike in the inbound traffic to their network. However, an increase in influx was observed, and the problem was solved.

The impacted services' monitoring systems might not catch the above events readily. Monitoring such situations is crucial so that the application remains available for all of its customers. Therefore, in the next lessons, we'll go through methods that help us to monitor the situations mentioned above.

Back

Visualize Data in a Monitoring System

Next

Design of a Client-side Monitoring Sy...

Mark as Completed

Report an Issue