

# Bottleneck of Bottleneck of a Complex System

---

## What's the most fragile part of a system?

If a system is broken into stateful and stateless. Stateful is anything you store for any amount of time.

Stateful components can have a bad state.

## Which part of a system is most stateful?

Database.

Database framework is detail. SQL vs NOSQL vs GraphQL or which vendor.

Encapsulate global data.

Do not be too flexible when creating schema.

It will create a nightmare to application code and performance issue.

Good schema design should shoot for human readability.

Bad programmers worry about the code. Good Programmers worry about data structures and their relationships.

Show me your flowchart and cancel tables and I shall continue to be mystified. Show me your tables and I won't need your flowchart.

## What can be wrong about database?

Write and transactions creates overload and death spiral makes it more overloaded.

How to avoid that?

### Leverage database

- Use joins and do not do things in memory

### Write-Read Replica

Use read replica.

### Guard in the first line

- Throttling
- Rate limiting
- Circuit break

## What can slow you down?

Separate the system into fast and dumb and slow and smart.

For slow one, run as a queue of background jobs.

## How to fast things up?

Cache. But DO NOT cache everything. Try indexing or a more efficient query first.

Cache can be a bad state.

## How to design in your head and what you should focus on?

Focus on hot path or trunks end-to-end through every layer will the simplest hello world example you can imagine.

Help you stay see main points instead of losing in details too soon.

trader-authenticate-risk management check-wallet check-order match-market data check-order status.

## How to log and care about metrics?

Living in a non-linear world, p95 and p99 more important than average.

Log more detailed about unhappy path.