#ASLI ENGINEERING

What to do after the outage is mitigated

SWIPE

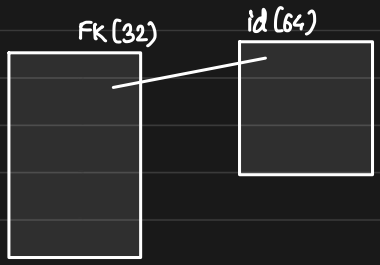
BY

ARPIT BHAYANI

Dissecting GitHub Outage

After the outage is mitigated

What happened?



Foreign key crossed INT32 limit

This led to outage in Actions & Pages (High throughput services)

and led to failures to some "push" and "pull"

Mitigation: Schema migration and altered the FK from 32 bit int to 64 bit — Very time consuming

* Please refer to other Outage Dissection videos to understand how such situations are actually mitigated and why it takes so much time

When do we say that issue is mitigated?

We say the issue is mitigated as soon as the usual operations for end user resumes

What after mitigation ?

1. Check for data inconsistencies

During outage work fails, and if we have not ensured atomicity our DB might be in an inconsistent state

eg: update a = a + 100
CRASH
~~update b = b - 100~~

Identifying such inconsistencies and resolving them

For this specific outage, there was nothing from GitHub

2. Cache invalidation

Some of the data in the cache might need to be deleted because the partial entries are not fruitful

For this outage, GitHub deleted token records from cache that became invalid → expired
→ stray entries

Also check dependent services for inconsistencies
[notify the team at least]

3. Setup alerting

Audit the alerting config to ensure right set of alerts are configured and we get notified about similar outage sooner

For this issue, GitHub team audited the alerts to ensure the outage does not happen for the same reason again

Alert on integer column reaching INTMAX limit

4. Preventive measures

Take preventive measures to ensure another outage does not happen for the same reason.

* Actions depend on the usecase

For this outage GitHub team configured linters to ensure no one accidentally cause the same outage again.

Audited all INT32 columns for their usecase and risk of reaching INTMAX