About Me?

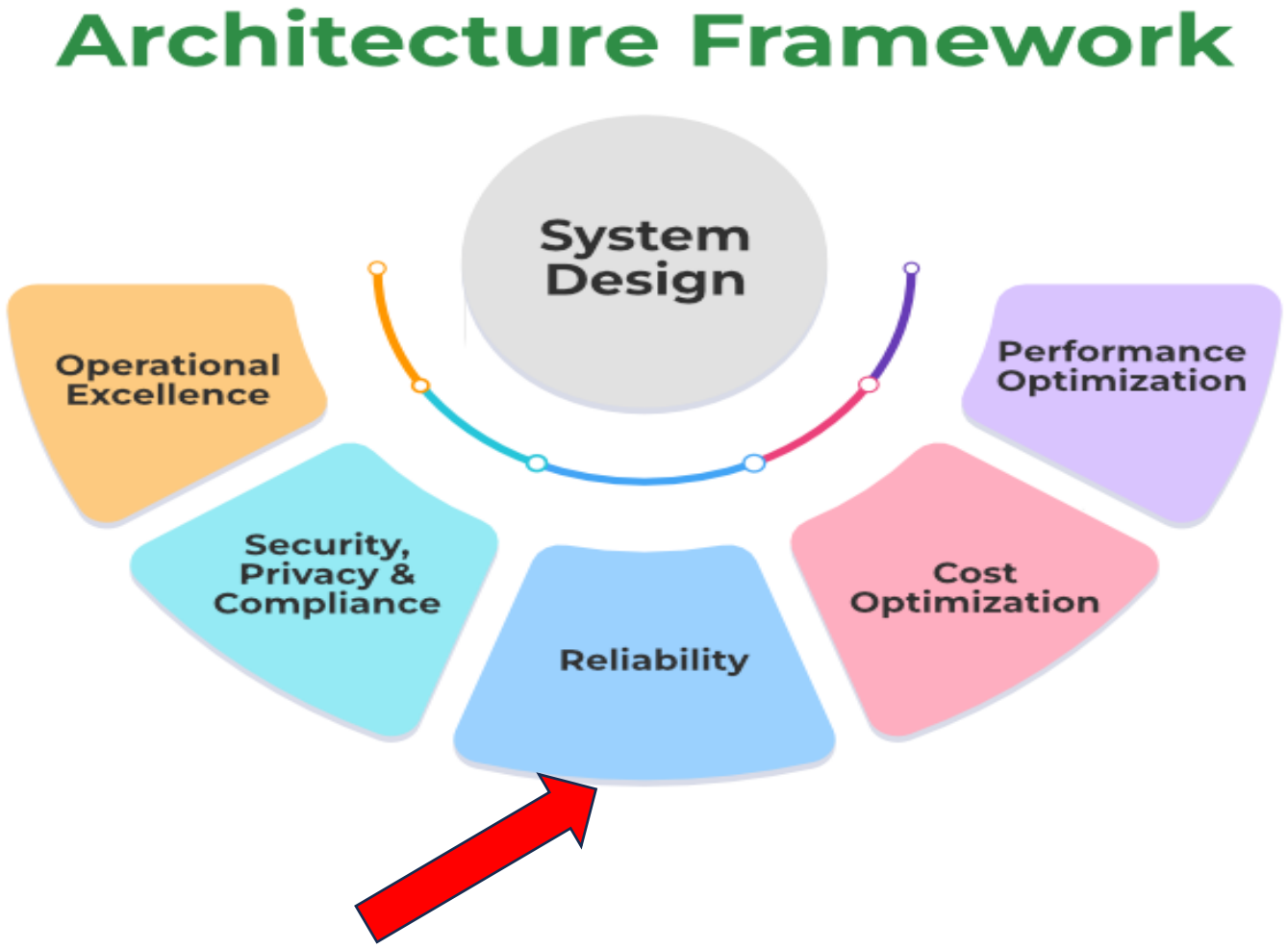
- ☆An Abide Learner
- ☆**14** years of experience.
- ☆Cloud Solution Architect
- ☆**8X** Multi-Cloud Certified.
- ☆AWS Community Builder **2023**

Sumit Kumar

Cloud Solution Architect

Building Reliable Infrastructure.

Cloud Architecture Framework



Reliability means **trust**. When you trust someone, you have confidence in their abilities, honesty, and dependability, which makes them reliable in your eyes

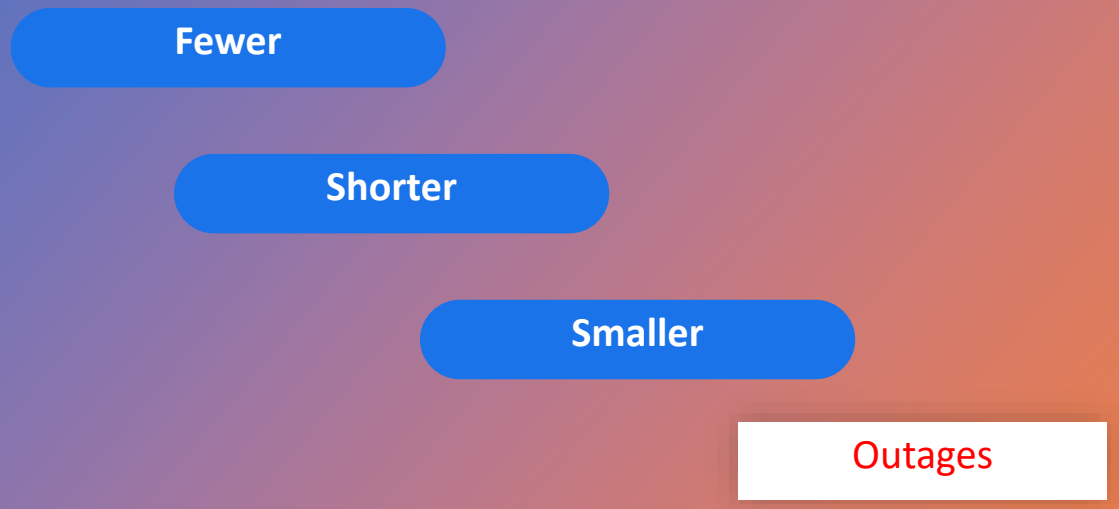
CLOUD NATIVE
COMPUTING FOUNDATION

In Context of System Design:

The extent to which customers can trust your system.

Reliable systems handle faults, failures, and errors.

Reliability means the system should be fault tolerant and working when faults/error happen.



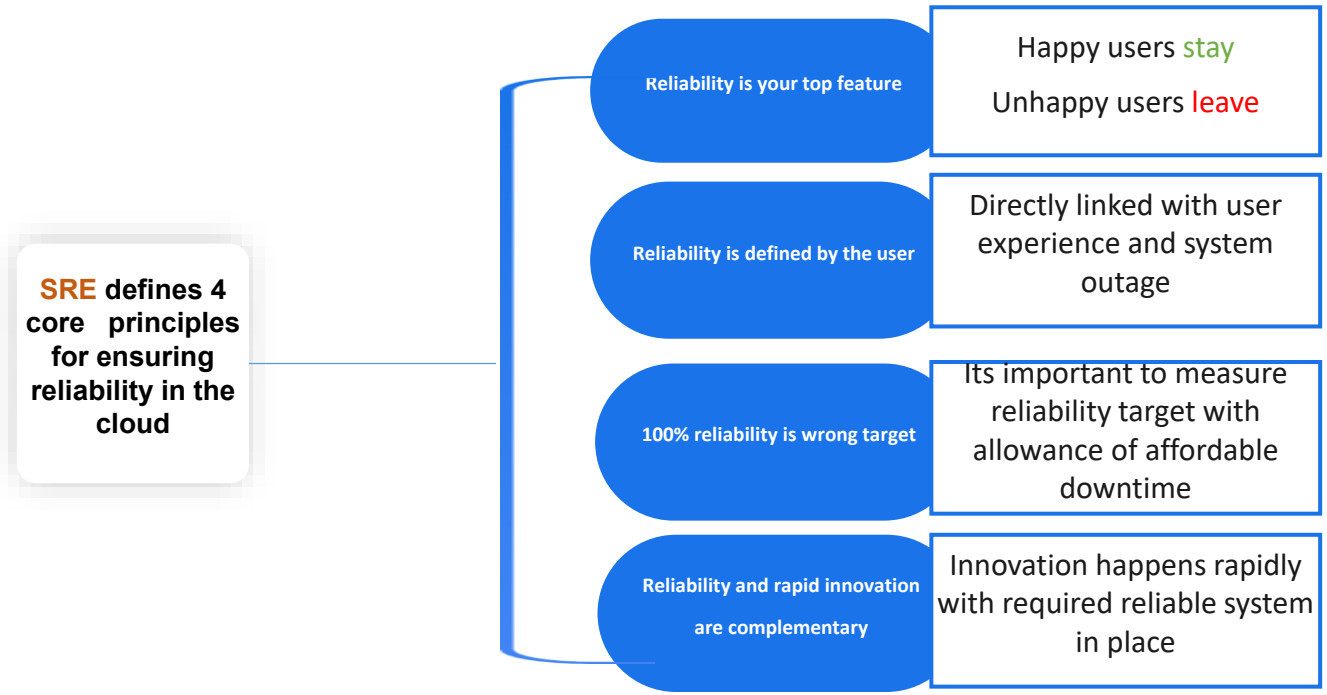
Reliability in Cloud

- Reliability is the probability that the system will meet certain performance standards and yield desired output for a specified period of time.

- A System becomes **more reliable** if it has….

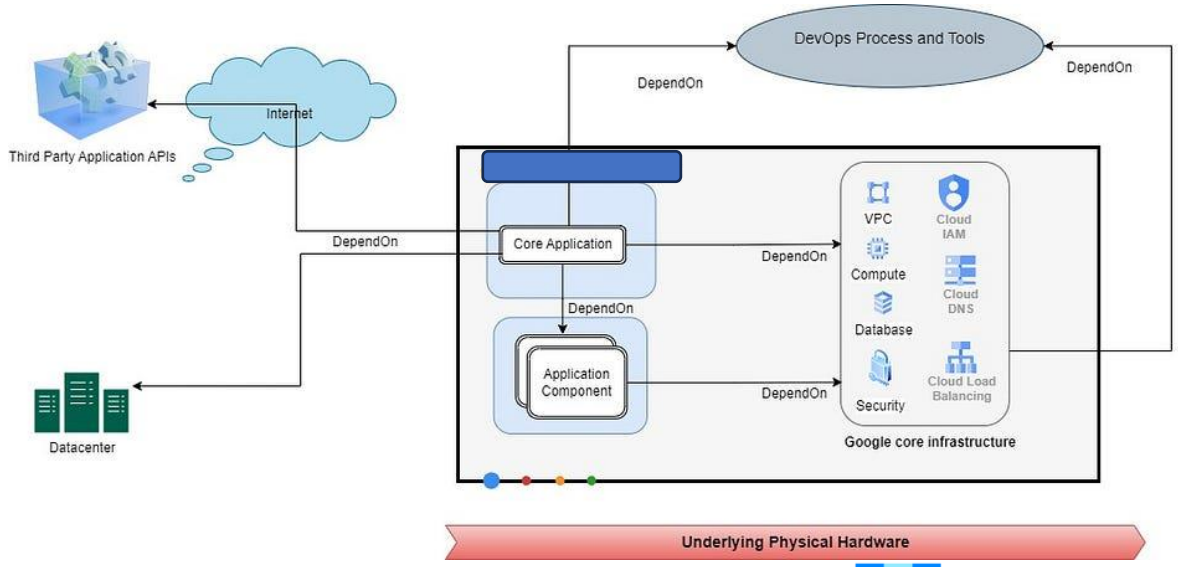
Core Principles of Reliability

Site Reliability Engineering (SRE) Defines 4 core principles for ensuring reliability in Cloud or On-prem infrastructure



Several important factors affect application reliability:

- The internal design of the application
- dependencies on secondary applications or components
- Cloud infrastructure resources
- Infrastructure capacity and quota
- DevOps processes and tools



Measuring Service Reliability

SLI – Service Level Indicator

SLI: a metric we track as a proxy for user happiness

SLI Example-

Request/Response	Availability Latency Quality
Data Processing	Coverage Correctness Throughput
Storage	Throughput Latency

SLO- Service Level Objectives

SLO: a target value for the SLI, below which the user is considered to be unhappy

- Agreed Upon by Internal Developers, Product Managers, DevOps/SREs
- Usually Defines in Percentage

Examples

Availability SLO
99.99% of requests return 200 status

Latency SLO
99.9% of requests get a response in >800ms

SLA – Service Level Agreement

SLA is a contractual agreement between the customer and the cloud provider for the level of service expected, failing to there are penalties.

Agreed upon by External Customer and Cloud provider.

.

Examples

Compute Engine SLA:
Instance in multiple AZ >=99.99%

A Single instance >=99.5%

Load Balancing >=99.99%

Measuring Service Reliability...

Error Budget

Error Budget: Allow for App downtime and slowness

Error Budget = 100% - SLO (Per minute or requests)

Example

If SLO = 99.9% of requests return successful stats than the Error Budget is 0.1% which means only 0.1% of requests can fail over the course of a month or quarter

Error Budget Policy

Action Taken by the organization to improve the stability when the error budget nearing exhaustion or exhausted.

Example

Reduce the Frequency of new releases and in extra cases freeze the release or product features for some time

Roll out changes in a smaller Gap

Critical user Journey

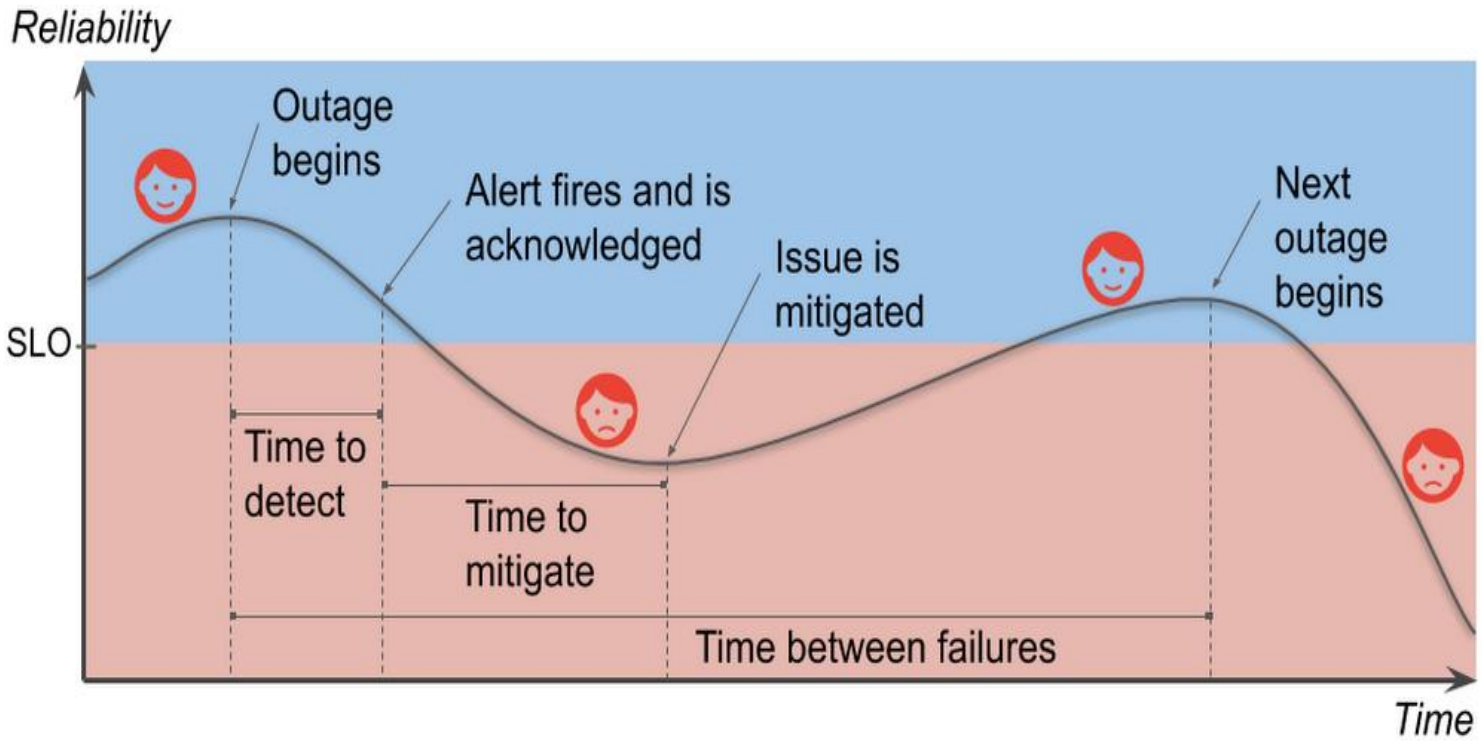
The most important service or operation in your application for the user. It varies by application and type of user

Example

1. Most apps are account creation and login.
2. Banking: the most important service or operation is view balance, transactions, etc
3. Online Shopping: Browse Store, Add to Cart and Checkout

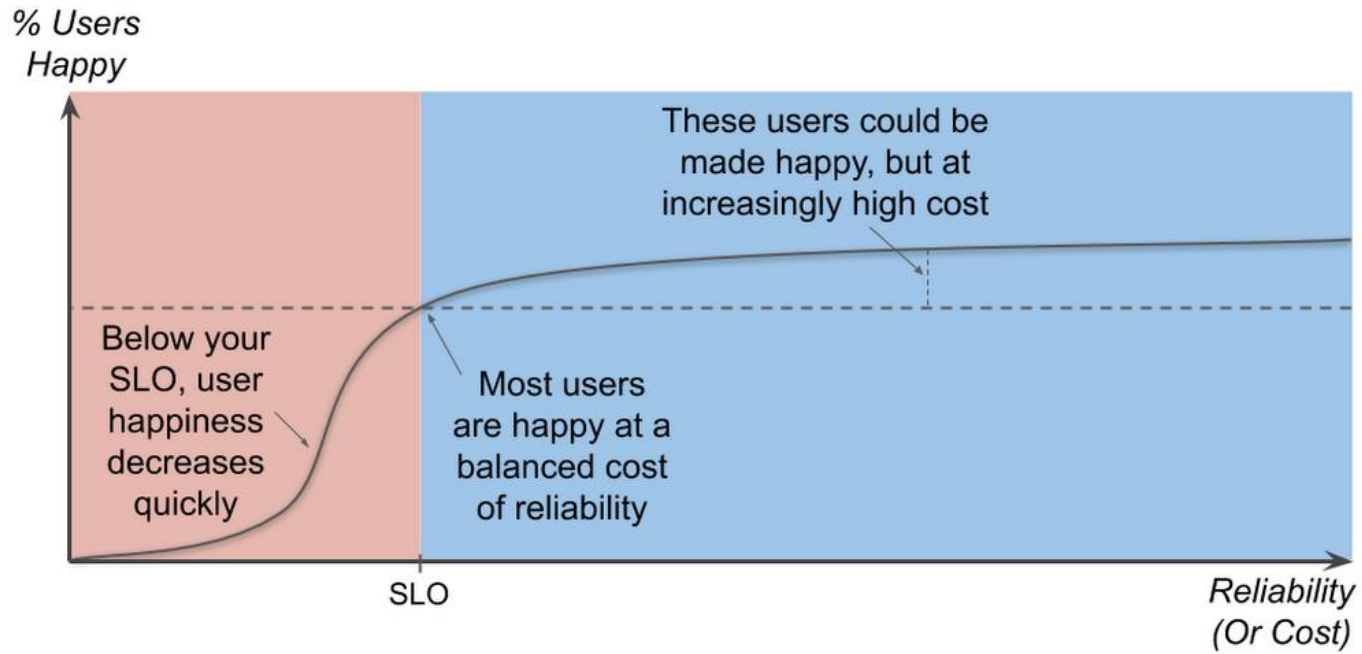
Minimize the Duration and Frequency of Outage

Understanding the production incident cycle



Graphing user happiness vs. reliability vs. cost

Understanding the production incident cycle

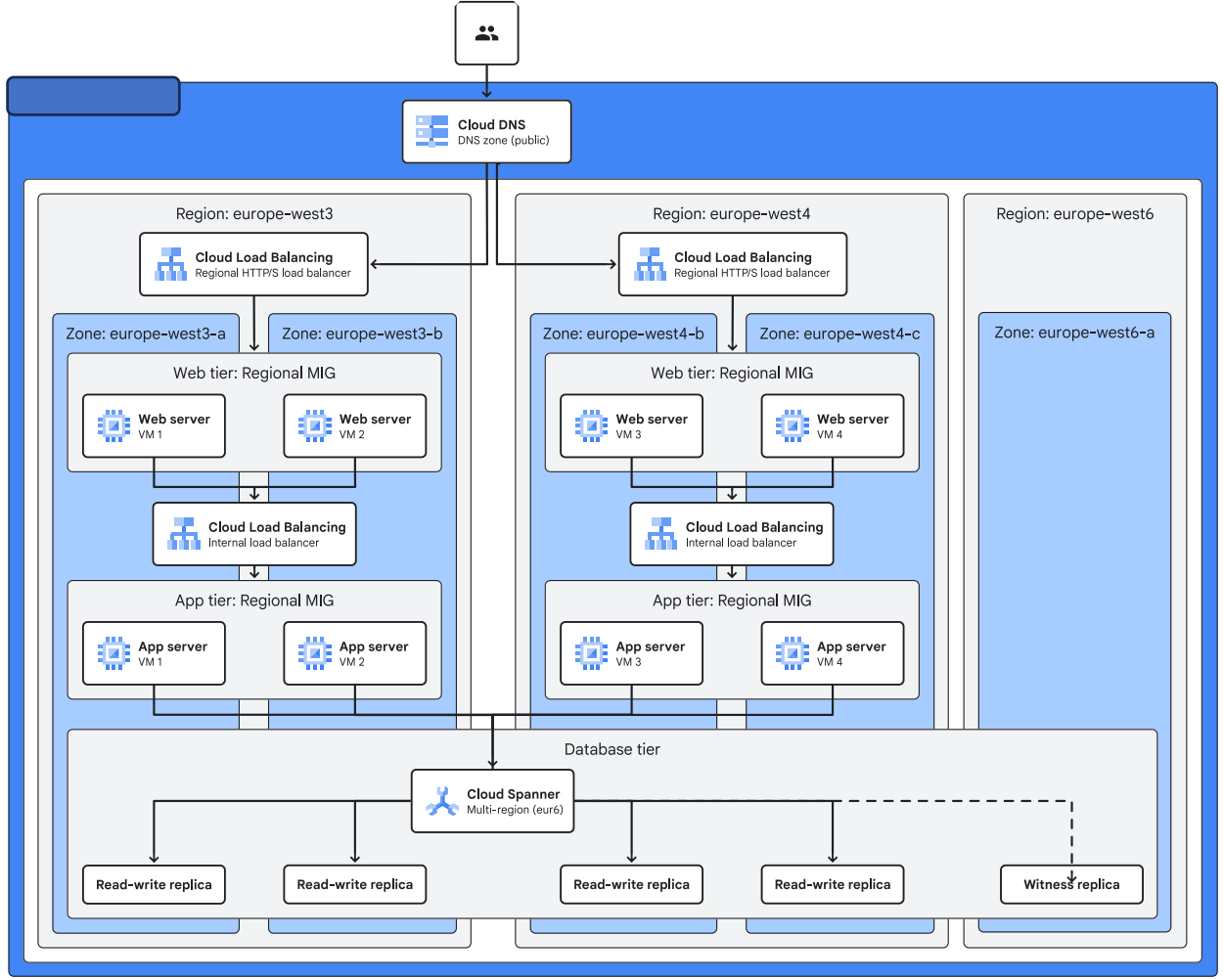


Case study

An Enterprise manufacturing organization is looking to modernize their On-Premises supply chain platform with Google Cloud

Key business priorities are as follows:

- No single point of failure
- Maximize business continuity
- Visibility into performance and reliability metrics
- Auto-healing environment
- No performance bottlenecks



No Single Point of Failure (SPoF) Multi-Zonal Architecture

Business Continuity (DB sync Replication for RPO/RTO)

Auto-Healing Environment (MIGs, Autoscaling, OS clustering)

No Performance Bottlenecks(CDN, Right Size instance, LB etc.)

Solving for Reliability

Thank you