Blog-Service Technical Design Document

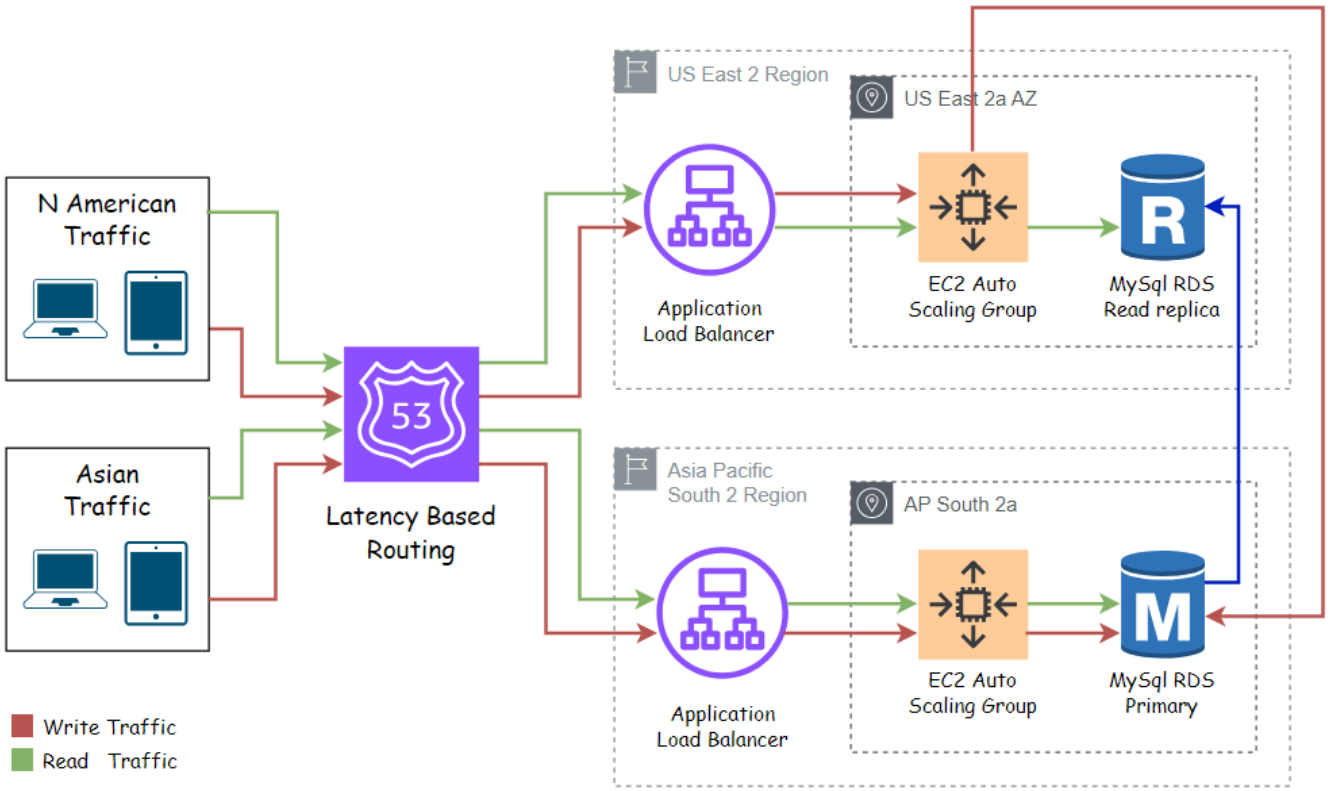
Introduction:

This document explains how one can architect a blogging service using Node.js on AWS. I am assuming a business requirement where the system should serve major traffic coming from Asia Pacific/India and some traffic coming from North America/US

System Requirements:

1. System should be Highly Scalable
2. System should provide low latency to both India and US traffic
3. System Should use AWS services
4. System should use MySQL Database

System Architecture:



AWS service Usage and Reasoning:

1. *Route 53*:

   I am using route 53 to achieve traffic routing based on the user location, if the user is in North America region, route 53 will route that request to Application load balancer that is deployed in US east 2 region.

2. *Application Load Balancer*:

   ALB is used to balance the load between EC2 instances, this will provide highly scalable architecture, ALB will regularly do health checks to make sure EC2 instances are healthy.

3. *EC2 Auto Scaling Group*:

   I am using auto scaling group to scale in/out the EC2 instances based on CPU, RAM, and latency metrics. This will achieve high scalability.

4. MySQL RDS:

   I am using AWS MySQL RDS instance to manage the application database, we will have one Primary that is deployed on Asia Pacific region that will server read traffic coming from Asia Pacific and write traffic coming from N America region.

   We will have one replica deployed on US East 2 region so that read traffic coming from N America can be routed to this read replica therefore we can achieve lower latency (I am assuming that this blog service will have heavy read and low write traffic).

   For the write traffic from N America, we will connect to the Primary instance and do the writes, write latency will increase because of cross region network connection but I think this will be ok considering that write traffic will be minimal.