Higher I/O Performance

Higher application performance through scaling of read-only database instances

Background

With growing business needs, maintaining high performance of your application is crucial. Even the smallest of failures can lead to greater response time and hamper user experience. The traditional single instance database design may not be sufficient enough to meet a large volume of I/O requests.

Highlights

- Faster response time
- High I/O performance
- No single-point-of- failure (SPOF)

Benefits

- ✓ Reduced burden on single database by efficiently handling more read request
- ✓ Cache layer saves the efforts of writing complex SQL queries
- ✓ Improved response time
- ✓ Disaster recovery capabilities by maintaining read replicas of databases

Recommended Solution Architecture
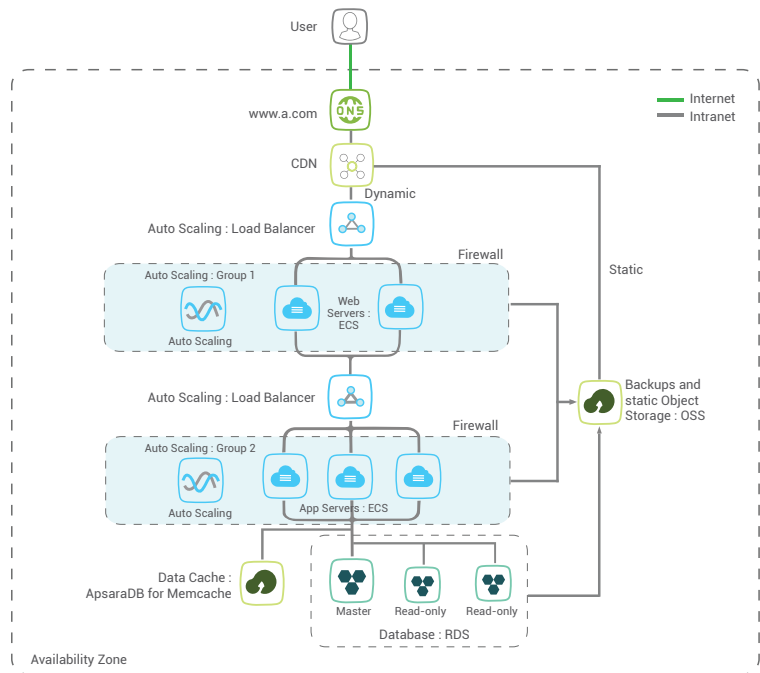
This architecture diagram illustrates a typical web application hosting architecture having higher I/O performance:

1. User request is received and served by the nearest DNS server, and automatically routed to the CDN for accelerated content delivery.

2. The request is then sent to the mapped Server Load Balancer, which distributes incoming application traffic among multiple ECS instances in a round robin manner.

3. To scale servers based on real-time traffic demands, auto scaling service is configured on web servers and application servers. This service ensures that servers are automatically added or removed from SLB and RDS whitelists.

4. By adding a cache layer and read-only database instances, the most queried data will be retained in the cache and read requests will be evenly distributed across scalable read-only database instances.

5. To store and manage relational data, application servers are connected to ApsaraDB for RDS databases. RDS is provisioned in master-slave manner to provide high I/O performance with no single-point-of-failure. A Memcache layer is also introduced in the architecture to cache result of the database query, which further enhances the overall I/O performance.

6. All database backup archive files, root location backup and log files of the web servers are stored in scalable OSS, which scales up or down automatically ensuring no disruption of services.