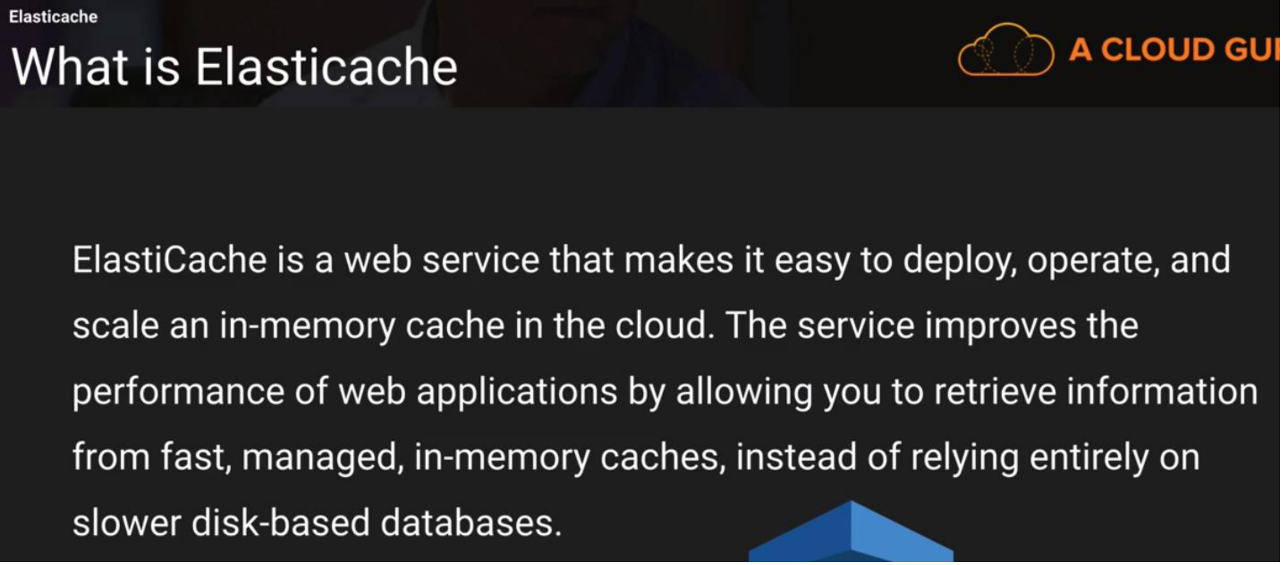
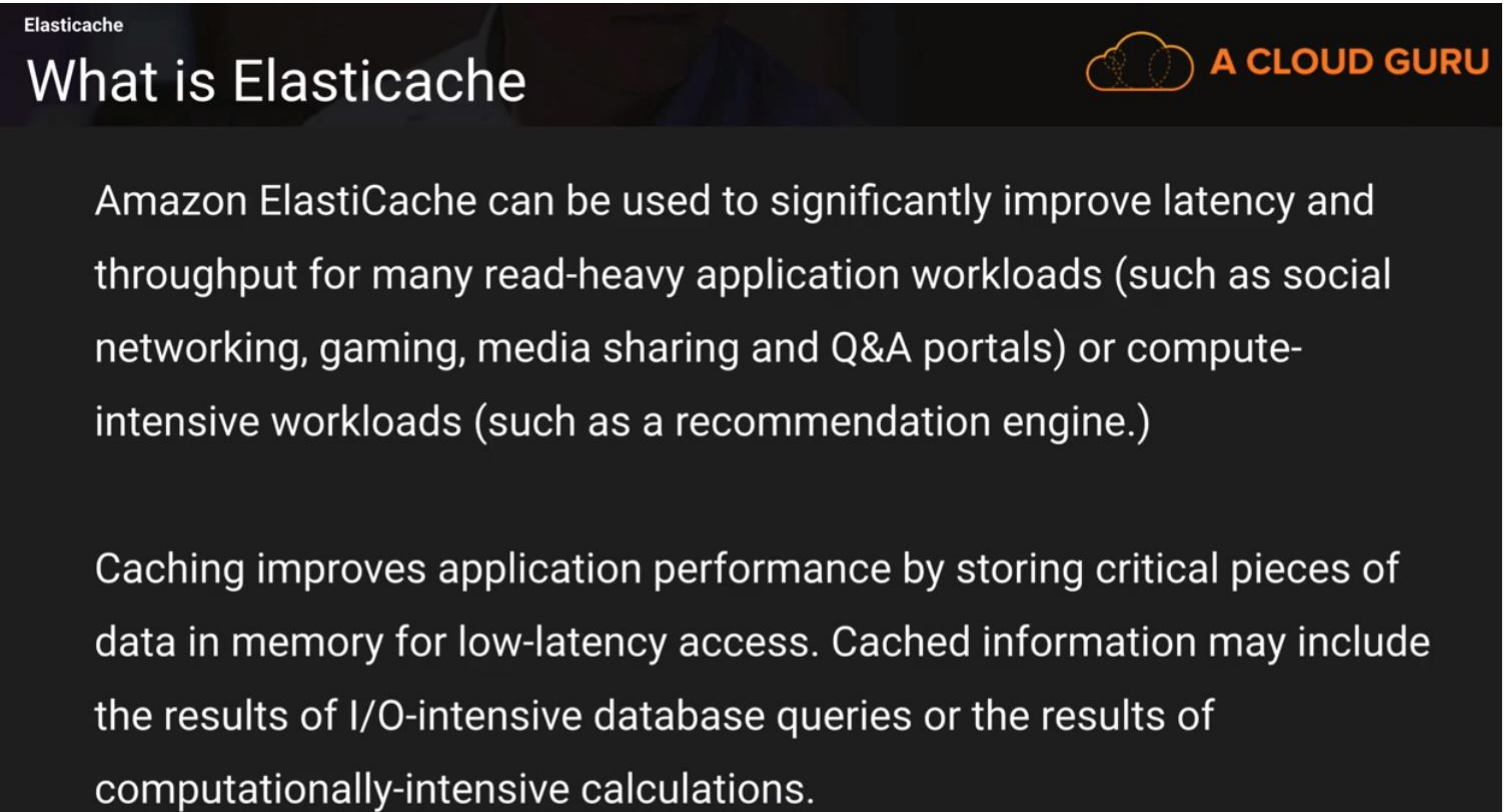


What is Elasticache

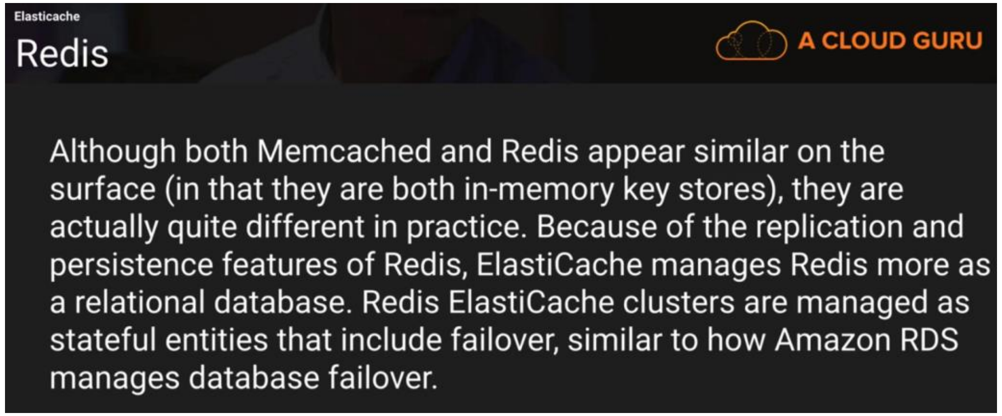
ElastiCache is a web service that makes it easy to deploy, operate, and scale an in-memory cache in the cloud. The service improves the performance of web applications by allowing you to retrieve information from fast, managed, in-memory caches, instead of relying entirely on slower disk-based databases.



What is Elasticache

Amazon ElastiCache can be used to significantly improve latency and throughput for many read-heavy application workloads (such as social networking, gaming, media sharing and Q&A portals) or compute-intensive workloads (such as a recommendation engine.)

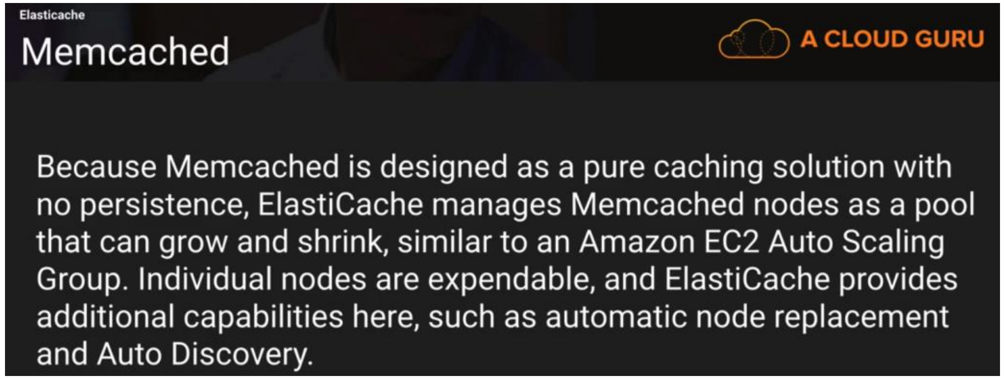
Caching improves application performance by storing critical pieces of data in memory for low-latency access. Cached information may include the results of I/O-intensive database queries or the results of computationally-intensive calculations.



Elasticache

Redis

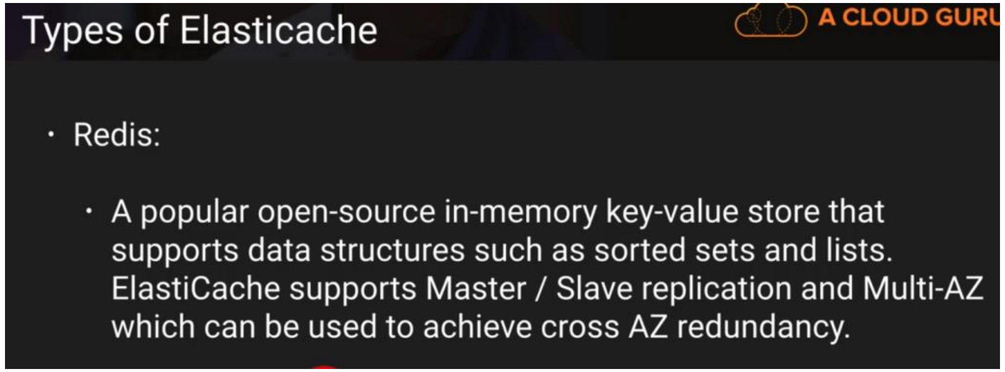
Although both Memcached and Redis appear similar on the surface (in that they are both in-memory key stores), they are actually quite different in practice. Because of the replication and persistence features of Redis, ElastiCache manages Redis more as a relational database. Redis ElastiCache clusters are managed as stateful entities that include failover, similar to how Amazon RDS manages database failover.



Elasticache

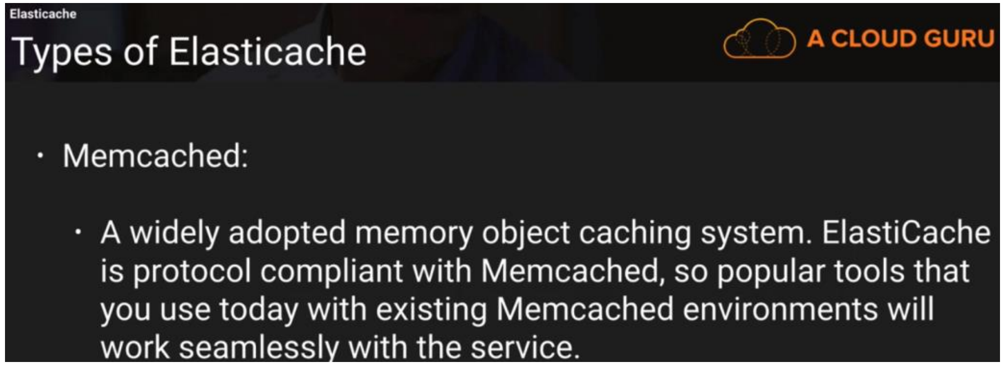
Memcached

Because Memcached is designed as a pure caching solution with no persistence, ElastiCache manages Memcached nodes as a pool that can grow and shrink, similar to an Amazon EC2 Auto Scaling Group. Individual nodes are expendable, and ElastiCache provides additional capabilities here, such as automatic node replacement and Auto Discovery.



Types of Elasticache

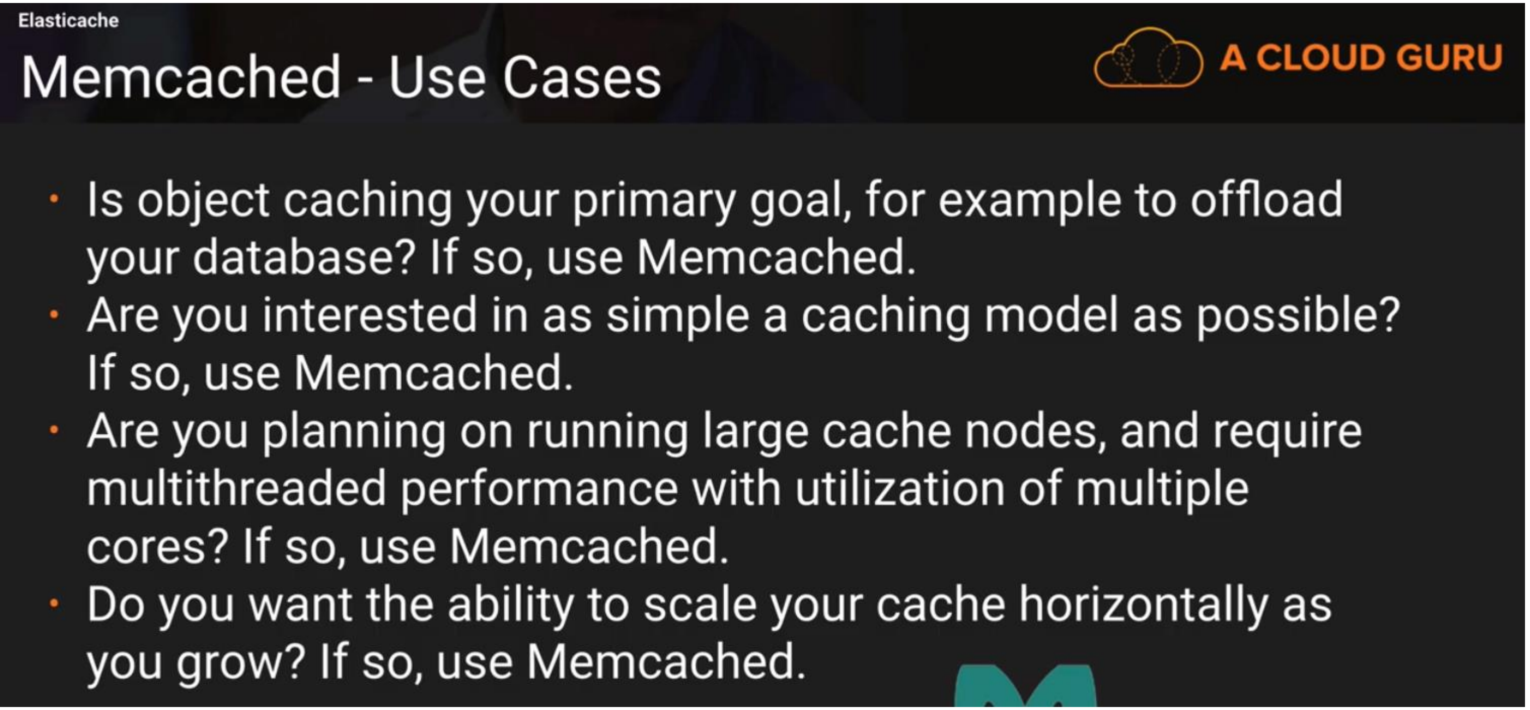
- Redis:
  - A popular open-source in-memory key-value store that supports data structures such as sorted sets and lists. ElastiCache supports Master / Slave replication and Multi-AZ which can be used to achieve cross AZ redundancy.



Elasticache

Types of Elasticache

- Memcached:
  - A widely adopted memory object caching system. ElastiCache is protocol compliant with Memcached, so popular tools that you use today with existing Memcached environments will work seamlessly with the service.

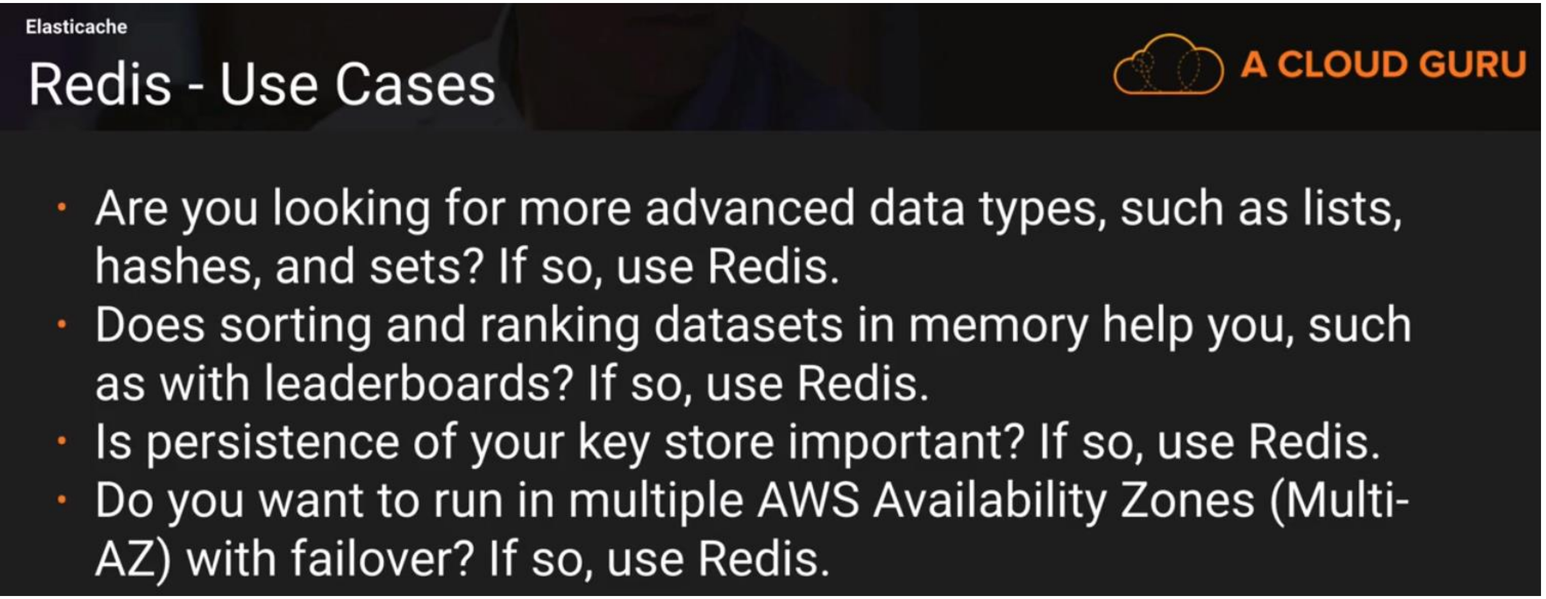


Elasticache

Memcached - Use Cases

A CLOUD GURU

- Is object caching your primary goal, for example to offload your database? If so, use Memcached.
- Are you interested in as simple a caching model as possible? If so, use Memcached.
- Are you planning on running large cache nodes, and require multithreaded performance with utilization of multiple cores? If so, use Memcached.
- Do you want the ability to scale your cache horizontally as you grow? If so, use Memcached.

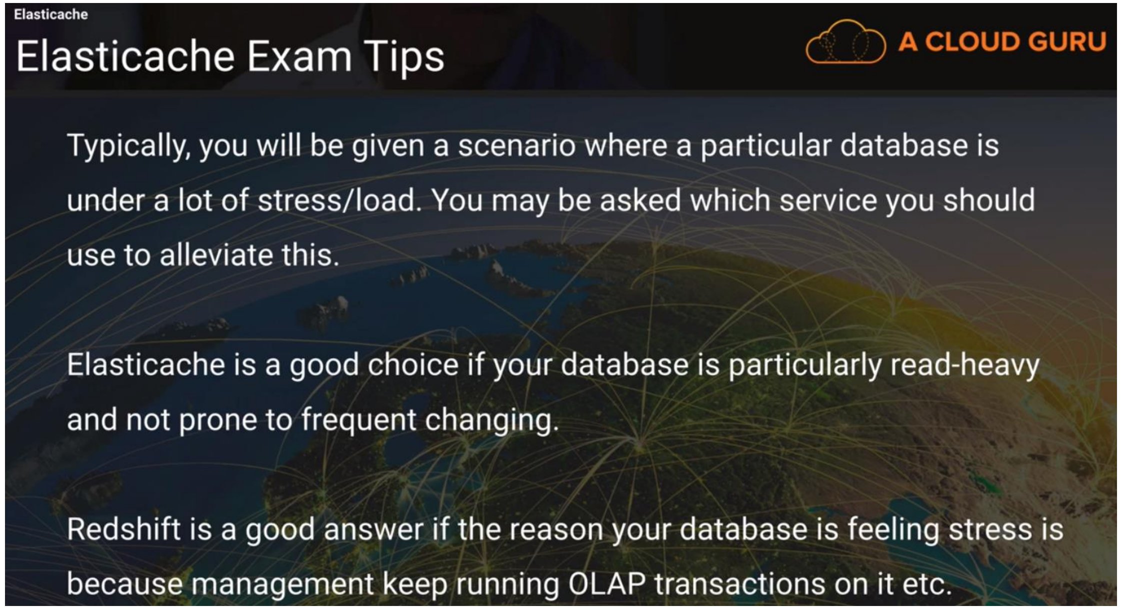


Elasticache

Redis - Use Cases

A CLOUD GURU

- Are you looking for more advanced data types, such as lists, hashes, and sets? If so, use Redis.
- Does sorting and ranking datasets in memory help you, such as with leaderboards? If so, use Redis.
- Is persistence of your key store important? If so, use Redis.
- Do you want to run in multiple AWS Availability Zones (Multi-AZ) with failover? If so, use Redis.



Elasticache

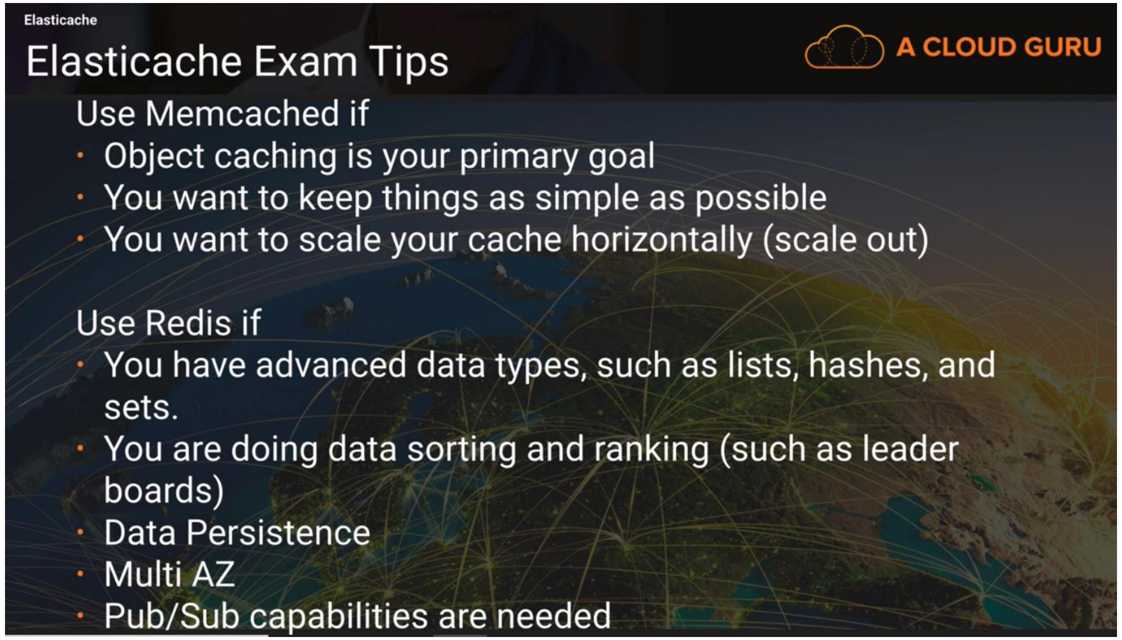
Elasticache Exam Tips

A CLOUD GURU

Typically, you will be given a scenario where a particular database is under a lot of stress/load. You may be asked which service you should use to alleviate this.

Elasticache is a good choice if your database is particularly read-heavy and not prone to frequent changing.

Redshift is a good answer if the reason your database is feeling stress is because management keep running OLAP transactions on it etc.



Elasticache

Elasticache Exam Tips

A CLOUD GURU

Use Memcached if

- Object caching is your primary goal
- You want to keep things as simple as possible
- You want to scale your cache horizontally (scale out)

Use Redis if

- You have advanced data types, such as lists, hashes, and sets.
- You are doing data sorting and ranking (such as leader boards)
- Data Persistence
- Multi AZ
- Pub/Sub capabilities are needed