

DI/FCT/NOVA
Mestrado Integrado em Engenharia Informática

Cloud Computing Systems
1st Semester, 2020/2021

Midterm Test (4/November/2020)

- 1) The first cloud service provided by Amazon, the first cloud provider, was the Elastic Compute Cloud, a service that allowed clients to rent virtual machines to run their programs (often referred as IaaS). Discuss possible reasons for this fact. [In your reply consider the alternative types of cloud services that could have been offered, such as PaaS and SaaS]

Amazon's business load peaks at different times of the year so if they want to handle peak loads that means they will have idle servers duration times of the year when load is lower. To make use of this idle servers they decided to rent their servers for others to use. It is possible that they were already using virtual machines for their own services so it might not have been too difficult to adapt that system for external users. Since this was the first cloud service being provided things like PaaS and SaaS might not have been well known at the time and the software required to manage all this kinds of services probably didn't exist yet or was harder to develop than the software required for renting virtual machines.

- 2) Consider you want to develop a reservation service similar to the one being developed in this course's project. **NOTE: the following questions must be answered considering this specific service – avoid generic comments. Specify your assumptions regarding the service, when relevant.**

- a) Would it be simpler to develop your solution based on a PaaS platform or on a IaaS platform? Justify.

PaaS simpler than IaaS, because.... / IaaS simpler than PaaS, because....

PaaS is simpler because we don't need to manage and deploy the services across VM's or deal with individual storage disks. With PaaS we just need to develop our application, request the required services (app service, database, cache, ...) and the cloud provider takes care of managing the hardware and service failures.

b) Are there benefits of using a CDN in the context of this specific service? Justify.

Yes, because... / No, because...

Yes, because we can use CDN to serve static content and that way the user only needs to contact the datacenter to make requests to the backend.

c) Azure functions allow to define functions activated through HTTP-triggers. Discuss the conditions under which it would be preferable to implement your service using this functionality (when compared with the more traditional solutions of using Azure App Service).

If our service has small inconsistent load then using azure functions with HTTP-triggers might be a better option since we don't need to pay for idle servers and the provider automatically provisions a server for our function to run on demand. If we have higher sustained loads it might be cheaper to pay for Azure App Service.

3) Consider the pricing of Azure Blob Storage service, as presented next (separately for data storage and for operations).

Data storage prices pay-as-you-go

All prices are per GB per month.

	PREMIUM	HOT	COOL	ARCHIVE
First 50 terabyte (TB) / month	€0.16445 per GB	€0.0166 per GB	€0.00844 per GB	€0.00152 per GB

Operations and data transfer prices

	PREMIUM	HOT	COOL	ARCHIVE
Write operations (per 10,000) ¹	€0.0193	€0.0456	€0.0844	€0.1012

Do you think the relative price difference between hot, cool and archive tiers for data storage and operations can be justified based on the resources used? Justify (given the techniques that can be used to provide reliable storage).

Yes / No, because....

Yes. From premium to archive the storage price increases. This can be justified by the type of storage device used, premium might use faster but more expensive flash storage while the others might just use slower but higher capacity hard drives / tape. For transfer operations the pricing increases the other way around. This could be because the premium/hot machines were designed for read/write operations and are always kept online at all times while cool and archive tiers might use slower machines that are only kept online for some time to save money.

4) The first generation of geo-replicated cloud databases adopted weak consistency models (in contrast with traditional SQL databases). Present reasons for such option.

Having a geo-replicated database with a stronger consistency will increase the latency of operations. Since not all applications require the strong consistencies offered by the traditional databases then trading consistency for lower latency may have been a good option.

5) Azure Cosmos DB supports different consistency models: strong consistency, bounded staleness, session, consistent prefix and eventual. It has additional support for transactions and conditional updates.

a) Session consistency level provide (among other) the following guarantee:
Read your writes: a read always reflects the writes executed in the session.

Although this seems the minimum an application would like to have, some consistency levels do not support it. Which consistency levels do not support this guarantee? Explain the reason why these consistency levels might fail to provide such guarantee.

Consistent prefix and Eventual consistency don't provide this guarantee.

Consistent prefix could be ordering all writes by sending them to a master replica and while the write is not ordered then the user will not be able to read is write until the replica he is reading from received the ordered write operation.

Eventual consistency provides no ordering guarantees for reads so just like in consistent prefix, a user might write and then read but he wont see his write.

- b) Cosmos BD defines the following service level agreements, regarding the Recovery Point Objective and the Recovery Time Objective.

Note: **Recovery time objective (RTO)** is the time until the system recovers from a disruptive event.

Recovery point objective (RPO) is the period of time for which updates might get lost in a failure.

Region(s)	Replication mode	Consistency level	RPO	RTO
1	Single or Multi-Master	Any Consistency Level	< 240 Minutes	<1 Week
>1	Single Master	Session, Consistent Prefix, Eventual	< 15 minutes	< 15 minutes

Consider the values presented in the second line. Given these values, what is possible to conclude regarding the replication process for those consistency levels? Justify. NOTE: specify whether replication is synchronous or asynchronous and what are the implications of the values 15 minutes for RPO and RTO.

The replication is asynchronous. If we perform a write and there is still a chance of dataloss in the next 15m that means our data is not yet replicated to other regions and that replication is being made asynchronously. The 15 minutes RPO means that after our write is confirmed than we could still lose our data if there was a failure in that region in the next 15m. The 15 minutes RTO means that we might no be able to contact or perform operations on the database for 15 minutes after a failure in a region.

- 6) Redis supports several data types, including strings, lists and sets. Consider that in the project you are developing you intend to store in the Redis cache the reservation an entity has for a given day. Explain how you could Redis for such goal. In your reply, specify the data type you would use and explain the reason, the keys used to store information and how this information is accessed/updated when clients issue operations.

Strings / Lists / Sets

String

Key:

entitiyid:timeslot

How used:

To check for the existence of a reservation at a given time just create the appropriate key and check if it exists in cache. To know what reservation exists at a given time just create the key and get the reservation id from cache. To update a reservation just use the key and rewrite the reservation id associated with that timeslot.

- 7) Lambda@Edge is a Amazon service that allows to run Lambda functions in points of presence located very close to clients (e.g. in ISPs). What kind of service can better use this service – stateless services, stateful services, both. Justify.

Stateless services, because... / Stateful services, because... / Both, because...

Stateless services, because POP's typically don't provide storage so using state means contacting a datacenter which would lose some of the benefits of running Lambda functions in POP's.