COMP90050 Advanced Database Systems: Tutorial
Winter term, 2023 (Week 1)

Exercises

Part 1: Walk through of the Group Project and presentation

Part 2:

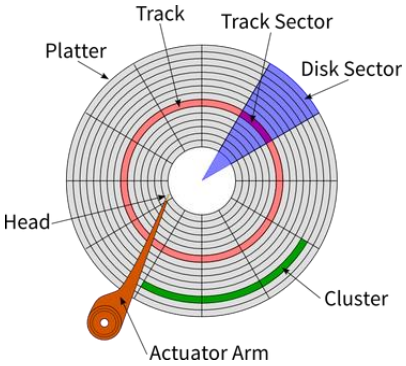
1. In a hard disk drive (HDD), the average seek time is 12 ms, rotation delay is 4 ms, and transfer rate is 4MB/sec. For simplicity, we assume in this question only 1MB equals 1000KB.

a) What is seek time delay?

b) What is rotation delay?

c) What will be the disk access time for a transfer size of 8MB? What will be the disk access time for a transfer size of 8KB?

d) In a solid state drive, what will be the disk access time for a transfer size of 8MB when transfer rate 4MB/sec? Is an SSD faster than an HDD for the same amount of data transfer? Why?



Solution:

1. (a) The seek time delay/seek latency is the period that the head of the actuator arm moves from a position to a required track.

(b) The rotation delay/rotation latency is the waiting period that the rotation of the disk brings the required sector of a track to head of the actuator arm.

(c) Disk access time for 8MB = seek time + rotation time + (transfer length/Bandwidth)= 12+4 + (8*1000/4) ms = 2016 ms

Disk access time for 8kB = seek time + rotation time + (transfer length/Bandwidth)= 12+4 + (8/4) ms = 18 ms

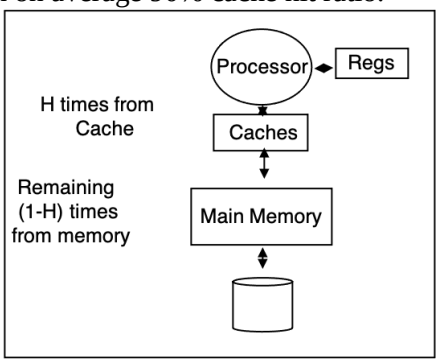
Comments: A comparison of the two cases highlights that sequentially reading large data pays off as seek time is buried under a lot of transfer time. For example, in the first case, seek time is only 0.6% of the total time while nearly all the time is spent on transferring data. In the second case, seek time is 66.7% of the total time while only a small fraction of the time is spent on data transfer.

(d) Unlike an HDD, an SSD do not have any rotating part. Hence there is no rotation delay or seek delay in and SSD. Therefore, for the same transfer rate and same amount of data transfer, an SSD is always faster than an HDD. Moreover, the data transfer rate of SSDs is usually higher than that of HDDs in general as well.

Disk access time of SSD = (transfer length/Bandwidth)= (8/4) sec = 2 sec

Unlike an HDD, an SSD does not have any moving parts. Hence there is no rotation delay or seek delay in an SSD. Therefore, the speed of SSDs is higher than that of HDDs given the assumptions in the question.

2. There are two different machines where machine A has a smaller cache with on average 50% cache hit ratio (H) and the other machine (machine B) has a much larger cache with on average 90% cache hit ratio. However, the memory access time of machine A is 100C and the memory access time of machine B is 400C (i.e., memory access in machine A is faster than memory access in machine B), where C is the cache access time. Which machine has overall faster effective memory access time?



Effective memory access time of A = 0.5*C + (1-0.5)*100C = 50.5C

Effective memory access time of B = 0.9*C +(1-0.9)*400C = 40.9C

Although memory access in machine A is faster than memory access in machine B, machine B has overall faster effective memory access time than machine A due to B's larger cache with higher cache hit ratio.

3. **More details on NoSQL databases - different types of NoSQL databases:**

a. Key-value storage: A key-value database stores data as a collection of key-value pairs where a key serves as a unique identifier. All accesses to the database are done via the keys. Both keys and values can be complex.

b. Document storage: Flexible for storing different kinds of documents, where they may not all have the same sections. XML, JSON, etc. are subclasses of document-oriented databases.

```
<contact>
  <firstname>Bob</firstname>
  <lastname>Smith</lastname>
  <phone type="Cell">(123) 555-0178</phone>
  <phone type="Work">(890) 555-0133</phone>
  <address>
    <type>Home</type>
    <street1>123 Back St.</street1>
    <city>Boys</city>
    <state>AR</state>
    <zip>32225</zip>
    <country>US</country>
  </address>
</contact>
```

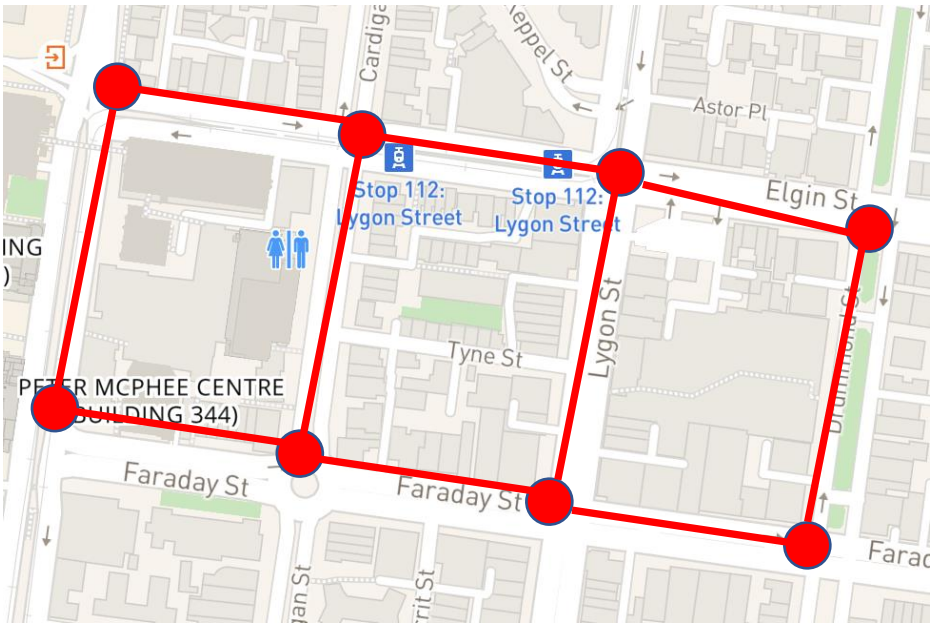
Image source: Wikipedia

c. Graph storage: Graphs capture connectivity between entities. Searching and traversing by relations are very fast in such structures.

The links can be material or immaterial:

- Links between two streets are junctions;
- Links between people as their Facebook connections (non-material links)

- a graph is a structure amounting to a set of objects (called vertices) where some pairs of the objects are connected/related in some sense. A connection is called an edge.



d. Discuss example applications of different type of NoSQL databases.

Solution:

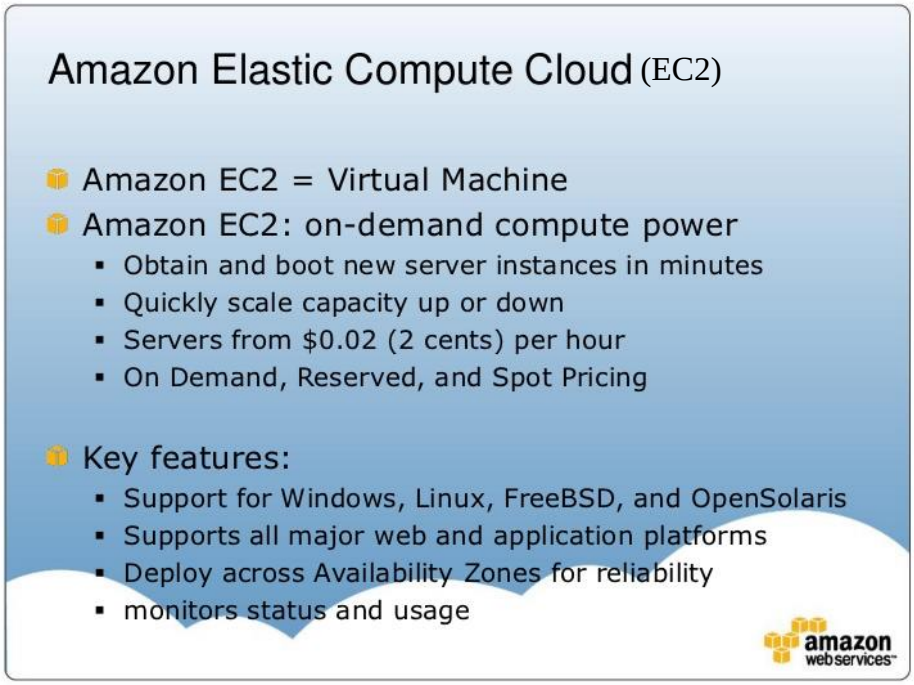
• Applications for key-value databases: Suitable if the dataset do not need complex relational table type of structure but can be expressed with simple key-value pairs. The simple structure allows faster insertion and search, and scales quickly. For example, shopping cart database in an e-commerce site.

• Applications for document storages: Well suited when different kinds of documents do not always have the same structure/sections. For example – a database of news articles.

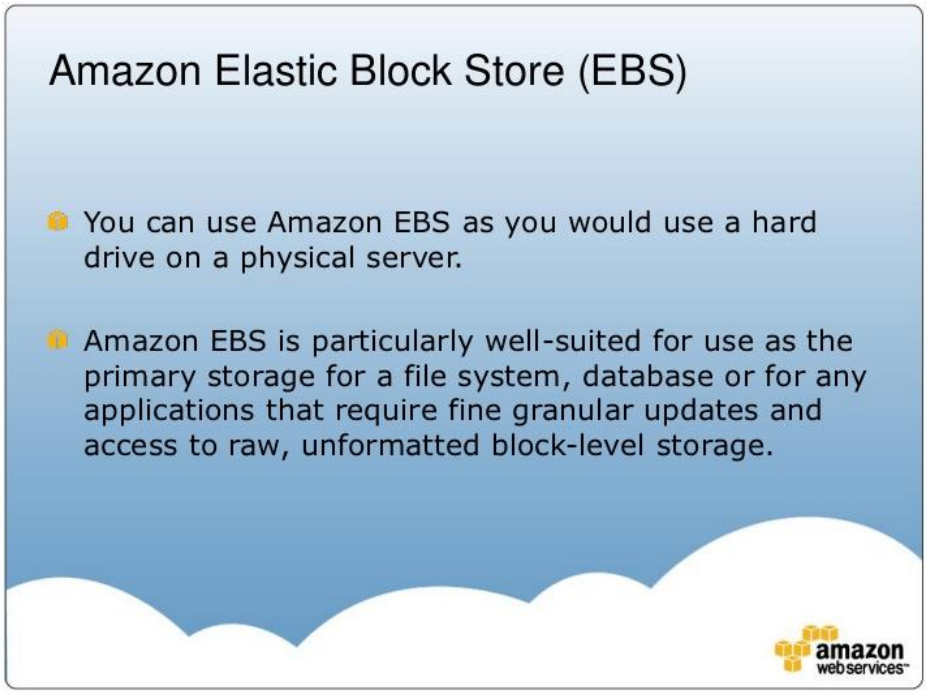
• Applications for graph databases: Well suited for storing connection data and querying such as social network connections (e.g., who are my friends of friends) and spatial data (e.g., route planning – which ways can I go now to reach destination).

Part 3

4. Review commercially available different cloud computing and database services from Amazon



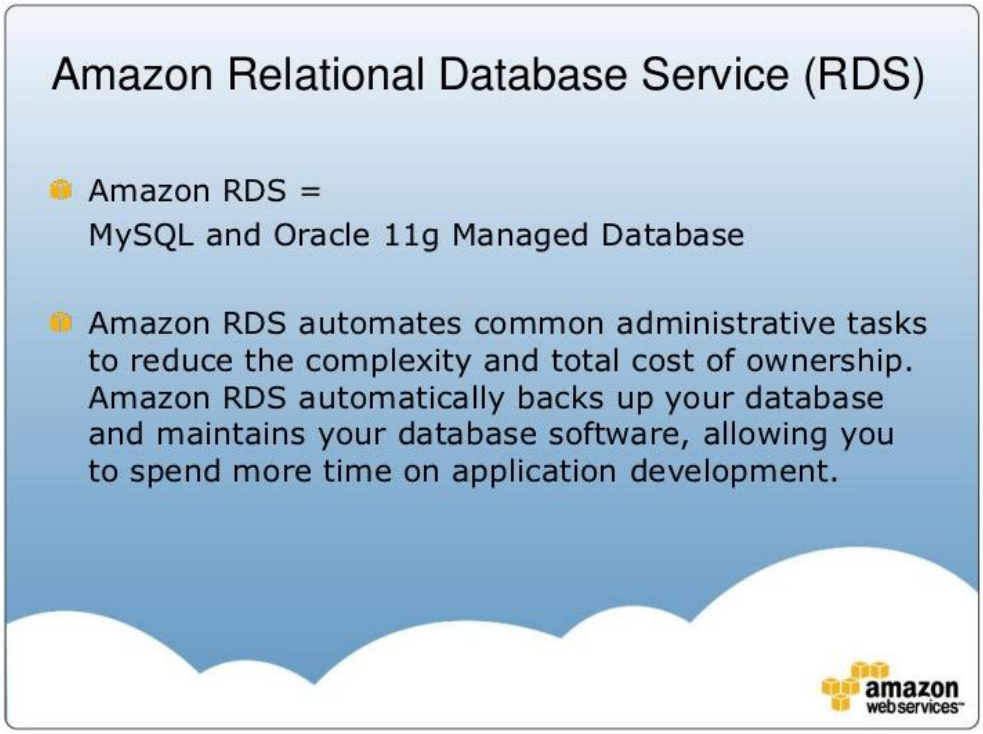
1



Amazon Simple Storage Service (S3)

- In traditional on-premise applications, this type of data would ordinarily be maintained on **SAN** or **NAS**. However, a cloud-based mechanism such as Amazon S3 is far more agile, flexible, and geo-redundant.
- Amazon S3 is a highly scalable, durable and available distributed object store designed for mission-critical and primary data storage with an easy to use web service interface.

amazon web services



5. Discuss the advantages and disadvantages of different database architectures for different application scenarios.

Solution:

(a) Centralised –

- Pros: Suitable for simple applications, easy to manage.
- Cons: May not scale well.

(b) Distributed -

- Pros: Scalable, suitable for large applications and applications that need data access from different physical locations.
- Cons: System administration and crash recovery are difficult. These systems usually have some data inconsistency issues.

(c) WWW -

- Pros: Very convenient to access and share data.
- Cons: Has security issues, no guarantee on availability or consistency. Extreme levels of administration issues.

(d) Grid -

- Pros: High processing capability as well as access at different locations.
- Cons: Similar issues to distributed databases. Less used nowadays, very similar to distributed systems with administration done locally by each owner.

(e) P2P -

- Pros: Suitable when the nodes of the network cannot be planned in advance, or some may leave and join frequently.
- Cons: Difficult to design transaction models. Applications are usually limited to simple file sharing.

(f) Cloud-based Databases -

- Pros: On-demand resources, cost-effective, maintenance done externally by the cloud provider.
- Cons: Has some privacy and confidentiality issues among others – but most trusted providers can address any issues emerging on this type relatively easily, e.g., Amazon etc.

6. Consider the different scenarios below and discuss which database architecture is the most suitable choice and why –

I. FriendBook is a new startup app that will launch its operation soon. They have only one office with not much budget right now, but they are expecting a high growth in the scale of millions of users across the globe in a couple of years. Which of the following database architecture is the most suitable choice for this scenario?

a. Cloud storage
b. World wide web
c. Distributed database
d. Centralised database

Solution:
Comments: Cloud storage allows for data to be stored across multiple servers in data centers, making it easier to scale horizontally as the number of users grows. This type of architecture also provides better reliability and fault tolerance.

II. FriendBook is a new social network site that will launch its operation soon. They have offices in many major cities of USA. They need a database that can handle millions of users across the globe. **For preserving privacy and security, they need their own data storage system, which is not shared or owned by any other company.** Which of the following database architecture is the most suitable choice for this scenario?

a. Cloud storage
b. World wide web
c. Distributed database
d. Centralised database

Solution:

Comments: Unlike the previous scenario, if data is transferred and stored in a 3rd party storage like cloud, the security is not in the hands of FriendBook (including encryption guarantee, data discloser agreement, etc.).

Hence, having the setup of their own distributed database (as they are located across many cities with many users across globe) is a more suitable solution.

Note: There is no universal truth or final answer on which architecture should be chosen for an application in some cases. The characteristics, advantages and disadvantages guide us on which one is more suitable over the others and even then some decisions are borderline if pros – cons is approaching the same level between two choices.