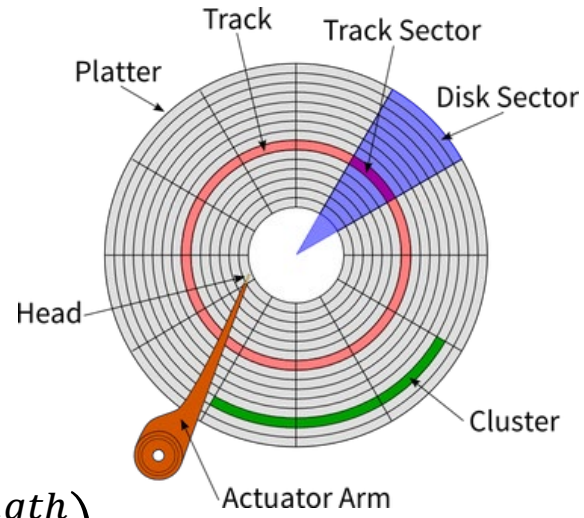
Advanced Database Systems Winter Semester

Week 1- Part Two (HDD/ SSD/ Memory Hierarchy)

Ahmad

Hard Disk Drives (HDD)



1. How does HDD work?
   - https://animagraffs.com/hard-disk-drive/
   - How can we formulate the read/write time?
     - $Disk\ access\ time \ = \ seek\ time \ + \ rotation\ time \ + \left(\frac{transfer\ length}{Bandwidth}\right)$.
   - [Think for now] If you were about to improve HDD, what changes would you made to the architecture?
     - Check the disk access time and respond.

Question 1 (Breakout rooms)

- 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 1MB equals to 1000KB.
  a) What physical property of an HDD causes the seek time delay?
  b) What physical property of an HDD causes the 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 (Assuming the base sequential data transfer rates are the same for the given two drives.)? Why?

Question 1- Solution (1)

a) What physical property of an HDD causes the seek time delay?
   - 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) What physical property of an HDD causes the rotation delay?
   - 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) 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?

Question 1- Solution (2)

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?

- c-1: Disk access time for 8MB =

$$seek\ time\ +\ rotation\ time\ +\ \left(\frac{transferlength}{Bandwidth}\right)$$

$$=\quad 12\quad +\quad 4\quad +\quad \left(\frac{8}{4}\right) * 1000\ ms\quad =\ 2016\ ms$$

- c-2: Disk access time for 8KB =

$$seek\ time\ +\ rotation\ time\ +\ \left(\frac{transferlength}{Bandwidth}\right)$$

$$=\quad 12\quad +\quad 4\quad +\quad \left(\frac{8}{4\ *1000}\right) * 1000\ ms\quad =\ 18\ ms$$

Question 1- Solution (2)

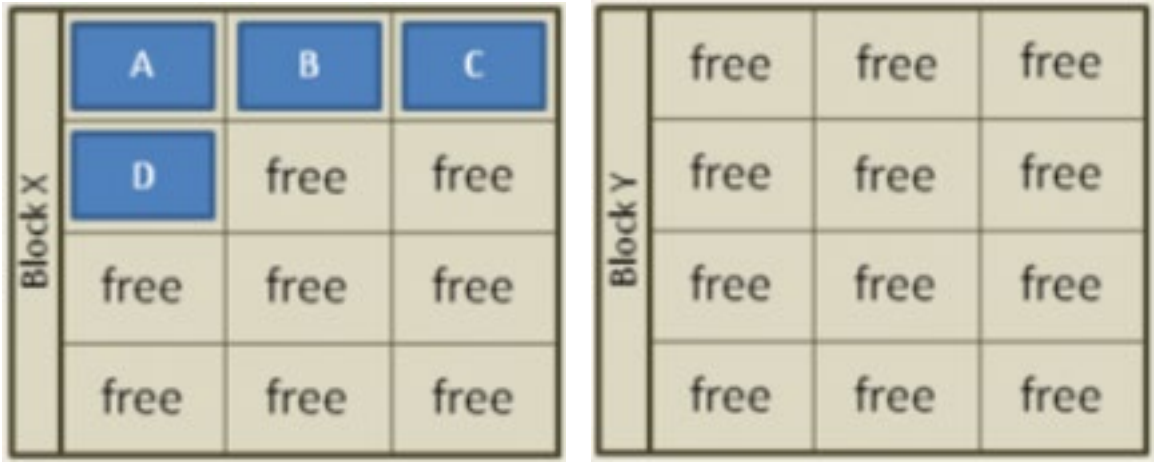
c) What will be the disk access time for a transfer size of 8MB? What will be the disk access time for a transfer size of 8KB?

- 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.

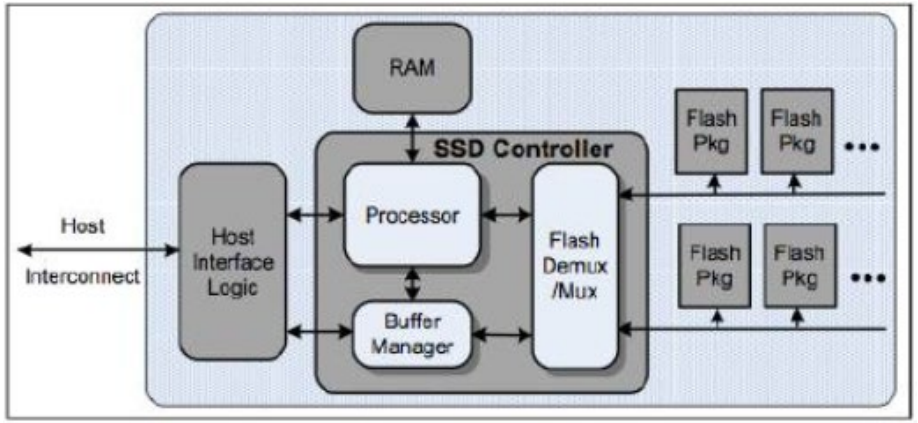
Improvement over HDD with SSD

$$Disk\ access\ time\ = seek\ time + rotation\ time + \left(\frac{transfer\ length}{Bandwidth}\right)$$

- Given the above disk access time, how can we improve HDD?
- Solid State Drives:
  - Called solid state, because they do not have a moving component.
  - Data is stored in grids of cells. Each grids is called a block and each row in a grid is called a page.



SSD (2)



- Comparison between SSD and HDD
  - https://www.youtube.com/watch?v=f_2Axf5XAlk
    - 00:23
- Disk access time in SSD:
  - $SSD\ access\ time \quad = \sout{seek\ time + rotation\ time +} \left(\frac{transfer\ length}{Bandwidth}\right)$
- Advantages of SSD:
  - Faster, Quiet, Not sensitive to movement or light hit.
- Disadvantage of SSD:
  - It is slow when overwriting data.
  - More expensive.

Back to Question 1 (Breakout rooms)

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?

Question 1- Solution (2)

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?

- 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 =

$$\left(\frac{transferlength}{Bandwidth}\right) = \left(\frac{8}{4}\right) = 2\ sec$$

Where does the Data Drives Fit in Computers?

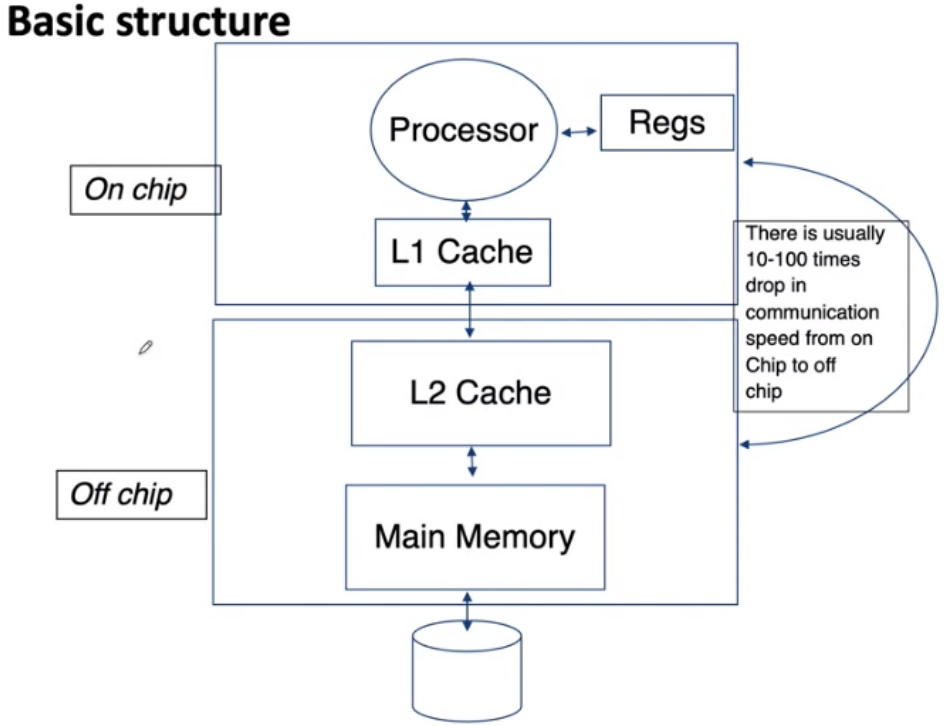
- Data after reading should be passed to the processors.
- The processors need to read from the storage in orders of Millions of bits.
- The access time for HDD ($\approx 7 \times 10^{-3}$) and SSD ($\approx 55 \times 10^{-6}$) are high.
- For a simple task with 1 Million bit reading from the memory, the processing time will be :
  - $\approx 7 \times 10^{-3} \times 10^{6} = 7000\ seconds\ \approx 117\ mintues$ for HDD
  - $\approx 55 \times 10^{-6} \times 10^{6} = 55\ seconds\ \approx 1\ mintue$ for SSD

How to solve the performance issue?

- Any ideas?

How to solve the performance issue?

- Any ideas?
  - Improve the access time for the memory.
    - Requires new technological design.
    - SSD is pretty much at the limits of a fast memory.
    - Fast memories will be very expensive.
  - Change the structure
    - Not all the data in memory is always required.
    - Design a new fast and small memory: Cache.
    - Read from Cache instead of main memory.
  - Access time for hierarchical structure with one cache.



$$access\ time = access\ cache * hit\ ratio + access\ mem\ * (1 - hit\ ratio)$$

Question 2 (Breakout Rooms)

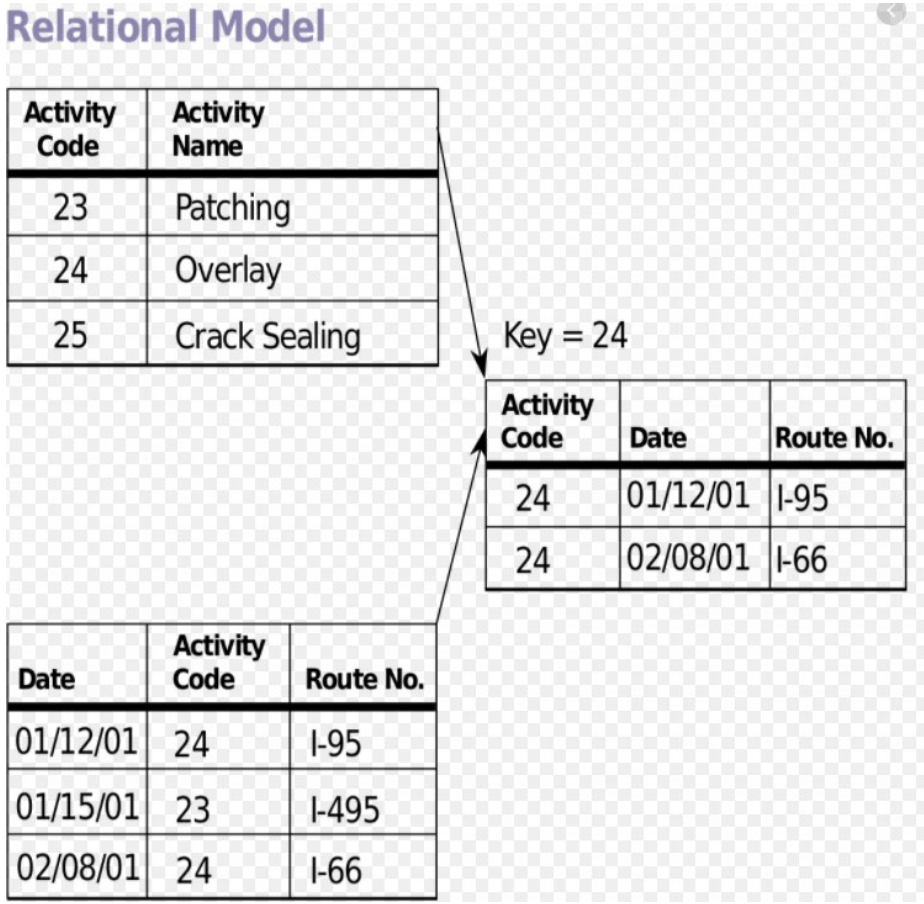
- There are two different machines where machine A has a smaller cache with on average 50% cache hit ratio (H) and the other machine (machine B) has a much larger cache with on average 90% cache hit ratio. However, the memory access time of machine A is 100C and the memory access time of machine B is 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?

Question 2 (Answer)

- 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.

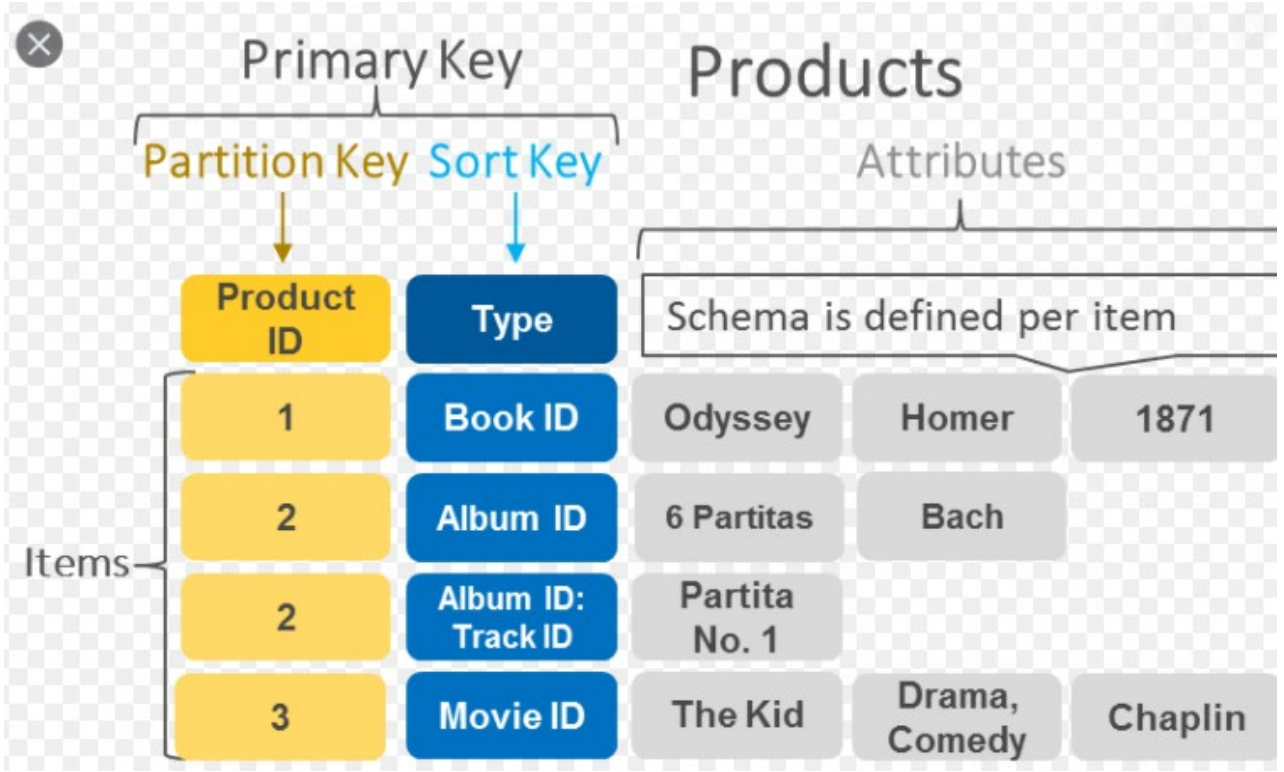
About different forms of Databases

1. Relational Databases
   - Tables are structured related to each other
   - Tables in database are related using primary/foreign key relationship.



Key-Value Databases

2. 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 access to the database are done via the key. Both keys and values can be complex.



Document Storage

3. 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.

Key	Document
1001	{ "CustomerID": 99, "OrderItems": [{ "ProductID": 2010, "Quantity": 2, "Cost": 520 }, { "ProductID": 4365, "Quantity": 1, "Cost": 18 }], "OrderDate": "04/01/2017" }
1002	{ "CustomerID": 220, "OrderItems": [{ "ProductID": 1285, "Quantity": 1, "Cost": 120 }], "OrderDate": "05/08/2017" }

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.

Applications of different forms of Databases

- Can you name applications of different forms of databases?

Applications of different forms of Databases

- 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 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 – news articles.
- Applications for graph databases – well suited for connection data: social network connections (e.g., who are my friends of friends) , spatial data (e.g., route planning – which ways can I go now to reach destination).

Take a Break (10 Minutes)

- Take a break for 10 minutes.
- Deep breath and some stretches would help.

Advanced Database Systems Winter Semester

Week 1- Part Three (Database Architectures/ Cloud Databases)

Ahmad

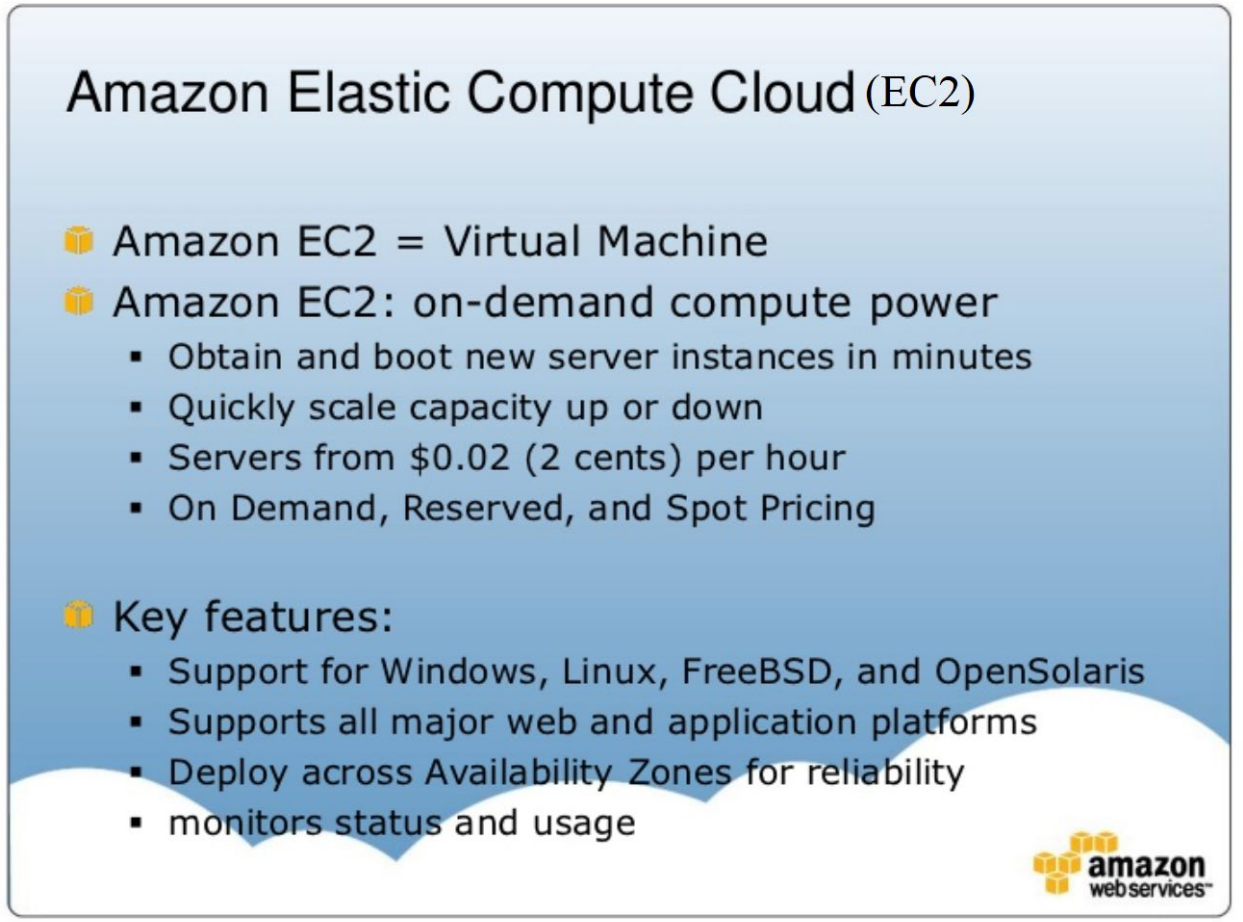
More on the Cloud DBS and Cloud Services

- The life for developers and programs has become much easier by cloud computing services.
- Cloud service providers have taken the responsibilities to do the dirty work, and provide an easy to use service to programmers.
- Programmers no longer need to worry about the maintenance issues, or about the required hardware capabilities.
- There are a number of cloud-service providers, and Amazon is one of the leading brands.

Amazon services

- Amazon offers a number of different services including:
  - virtual computers, block storage, simple storage, and Relational database
  - In addition to these services, amazon also offers NoSQL database services (Amazon DynamoDB)
- Here we review some of the most important Amazon cloud-services

Amazon Elastic Compute Cloud (EC2)

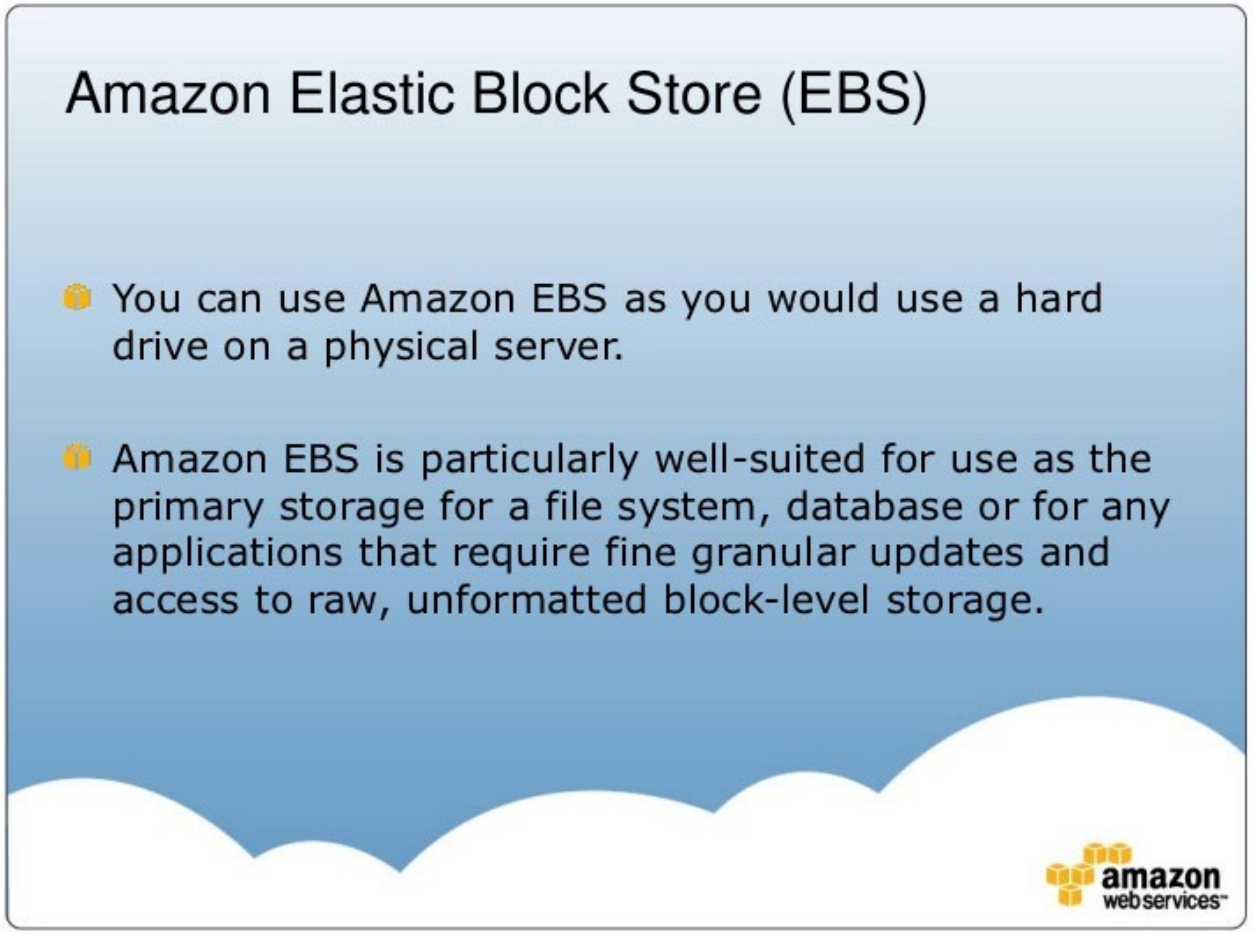


Amazon Storage Services

- Amazon Elastic Block Storage
- Amazon Simple Storage

- The difference between elastic block storage (EBS) and simple storage service (S3) is EBS is only accessible from a single EC2 instance, while you can use S3 across multiple instances.

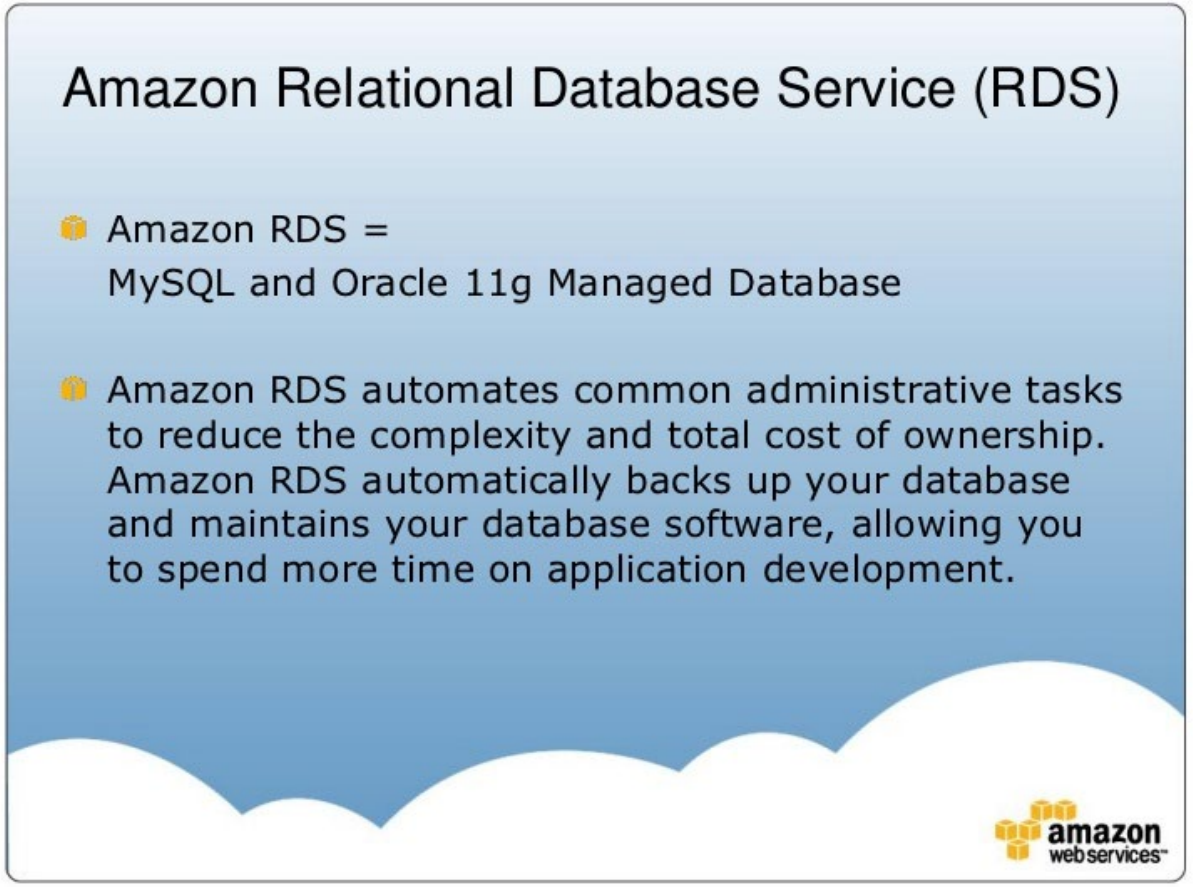
Amazon Elastic Block Storage



Amazon Simple Storage

Creating Virtual Private Cloud

Amazon Relational Database



Different Database Architectures

- Properties of Different DBS Architectures

Centralized	Distributed	WWW	Grid	P2P	Cloud
• Data stored in one location	• Data distributed across several nodes, can be in different locations	• Stored all over the world, several owners of the data	• Like distribu-ted, but each node manages own resource; system doesn't act as a single unit.	• Like grid, but nodes can join and leave network at will (unlike Grid)	• Generalization of grid, but resources are accessed on-demand

Question 1

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

Answers in Short

- A. Centralised – suitable for simple applications, easy to manage; may not scale well
- B. Distributed – Scalable, suitable for large applications and applications that need data access from different physical locations; System administration and crash recovery is difficult, usually have some data inconsistency
- C. WWW – Very convenient to access and share data; security issues, no guarantee on availability or consistency
- D. Grid – Less used now-a-days, very similar to distributed systems with administration done locally by each owner
- E. P2P- Suitable when the nodes of the network cannot be planned in advance, or some may leave and join frequently. For example, sensor network
- F. Cloud database – on-demand resources, cost-effective, maintenance done externally by the cloud provider; some privacy and confidentially issue – but most trusted providers well-address them

Question 3-I (Breakout Rooms)

- 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

Question 3-l (Answer)

- Answer is a.

- As FriendBook do not know a certain number of users and the exact timeline on the growth, the on-demand cloud storage is more cost-effective, can scale with the growth, can server clients across the globe, and Friendbook can focus more on developing their product instead of managing hardware resources.

Question 3-ll.

- 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

Question 3-II (Answer).

- Answer is c.

- 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.