Naman Jain Vimal Kumar
1001670153

1) *Discuss the characteristics that a data model for conceptual schema design should possess.*

The characteristics that a data model for a conceptual schema design should possess are:

1. *Expressiveness* -
   The data model should be expressive to find different types of entities, relationships, and respective constraints.

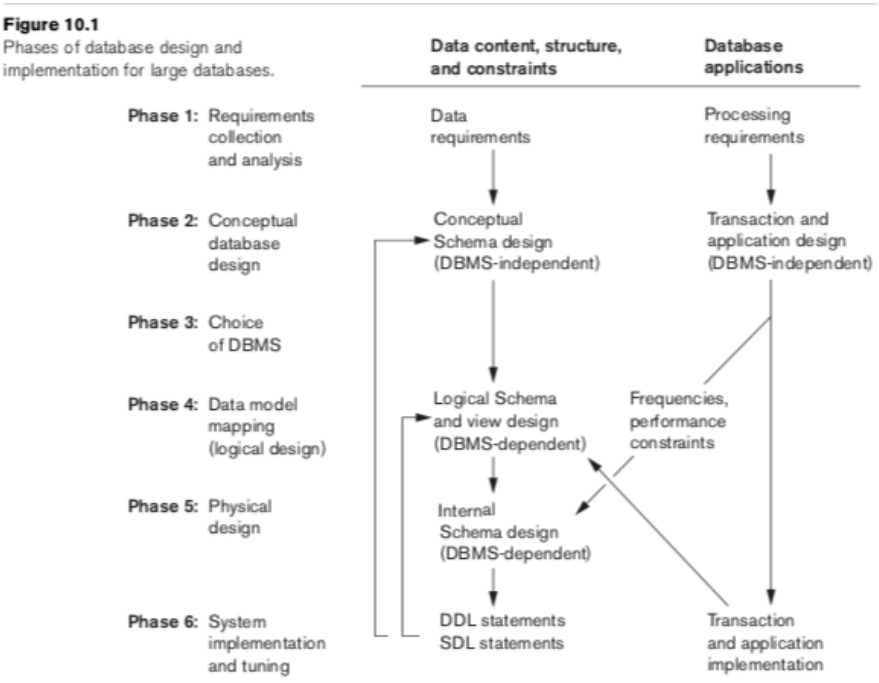
2. *Simplicity and understandability* -
   The data model should be simple for a user to understand and should be able to use its concepts.

3. *Minimality* -
   The data model should have small basic concepts that are distinct and simple to understand.

4. *Diagrammatic representation* -
   The data model should possess a diagrammatic notation that is easy to understand.

5. *Formality* -
   The data model must represent a formal unambiguous specification of the data. That is, a proper schema has to be followed.

2) *What are the six phases of database design? Discuss each phase.*



There are six phases of database design. They are:

i. <u>*Requirements collection and Analysis*</u>
   It is most crucial part to design the database. Here the information is gathered based on the intended use of the database.

The activities in Requirement Collection are:

a. To identify the application areas and user groups that use the database.
b. Studying the existing documentation and analyse the same.
c. The current operating environment is studied. That is the frequency of transactions and other details.
d. Collect written responses from user groups using the database.

These information is collected and analysed. Requirements Specification Techniques are used to transform requirements into a structured representation. Some of them are:

- Object Oriented Analysis (OOA)
- Data Flow Diagrams (DFD)
- Refinement of application goals
- Computer-aided

ii. *Conceptual Database Design*

The activities in this Phase are:

- To examine the data requirements generated by Phase 1 and producing a **conceptual schema diagram** for the same.
- To produce high level specifications like **transaction and application design** by examining the database application analysed in Phase 1.

The goal of Conceptual Schema Diagram is to completely understand the structure of the database, its entities, their inter relationships and constrains. To design the same, we use the following strategies:

a. Top – Down Strategy
b. Bottom – Up Strategy
c. Inside – Out Strategy
d. Mixed Strategy

The goal of Transaction and Application Design is to design the characteristics of known database transactions in a DBMS independent way. The most common technique for specifying transactions is by their input/output and functional behaviour.

iii. *Choice of a Database Management System*

As said, a DBMS is selected. But there are different factors or costs that govern the selection of the database system. They are:

a. Software Acquisition Cost
b. Maintenance Cost
c. Hardware Acquisition Cost
d. Database Creation and Conversion Cost
e. Personal Cost
f. Training Cost
g. Operation Cost

iv. *Data Model Mapping or Logical Database Design*

At this Phase, a conceptual schema needs to be constructed along with external schemas which is dependent on the DBMS selected. These schemas are mapped

with the conceptual schema generated in Phase 2. The mapping can proceed in couple of stages. They are:

a. System-independent Mapping.
b. Tailoring schemas to a specific DBMS

v. *Physical Database Design*

It is defined as a process of choosing file storage structures and access paths for the database files to achieve best performance for various database applications. To achieve the same, the following parameters are used a guide to choose:

- Response Time
- Space Utilization
- Transaction Throughput

vi. *Database System Implementation and Tuning*

Here, the database system is implemented. This is typically the responsibility of the DBA. His responsibilities include:

- Composition of DDL Statements.
- Load Database.
- Convert Data from Earlier Systems.

Database programs are implemented by application programmers, and systems are tuned to monitor the utility to collect performance statistics.

3) What is system-independent data model mapping? How is it different from system-dependent data model mapping?

System-independent data model mapping can be defined as a mapping process that does not consider any characteristics of a specific Database Management System for the implementation of the data model. They use generic mapping process like the ones in ER and EER.

	System Independent Data Model Mapping	**System Dependent Data Model Mapping**
Representation	ER or EER or other schema	Actual DDL Statements
Dependent on Hardware	No	Yes
Dependent on Software	No	Yes
Data Model	Conceptual Data Model	Logical Data Model

4) Explain UML approach and give its advantages.

- UML Approach is a widely used standard approach to cover all the aspects that is requirements analysis, modelling, design, implementation and deployment of databases or applications.
- It stands for Unified Modelling Language Approach.
- It can be used to model any type of application running on any type of operating system, programming language or network.
- It consists many types of diagrams. They are:
    - Structural Diagrams
        - Class Diagrams

- Object Diagrams
- Component Diagrams
- Deployment Diagrams
- Use Case Diagrams
- Sequence Diagrams
- Collaboration Diagrams
- Statechart Diagrams
- Activity Diagrams

o Behavioural Diagrams

Advantages of UML Approach are:

- Widely used
- Flexible
- Visual Representation
- Readability
- Reusability
- You Need to Know Only a Fraction of the Language to Use It
- Abundance of UML Tools

5) A file has r=20,000 STUDENT records of fixed-length. Each record has the following fields: NAME (30 bytes), SSN (9 bytes), ADDRESS (40 bytes), PHONE (9 bytes), BIRTHDATE (8 bytes), GENDER (1 byte), MAJORDEPTCODE (4 bytes), MINORDEPTCODE (4 bytes), CLASSCODE (4 bytes, integer), and DEGREEPROGRAM (3 bytes). An additional byte is used as a deletion marker

average rotational delay rd = 12.5msec,
block size B=512 bytes,
average seek time s = 30 msec
bulk transfer rate btr = 409.6 bytes/msec
Transfer rate tr = 512 bytes/msec

a) Calculate the record size R in bytes.

$$R = Size(Name) + Size(SSN) + Size(Address) + Size(Phone) + Size(BirthDate) + Size(Gender) + Size(MajorDeptCode) + Size(MinorDeptCode) + Size(ClassCode) + Size(DegreeProgram) + Size(Deletion\ Marker)$$

$$R = (30 + 9 + 40 + 9 + 8 + 1 + 4 + 4 + 4 + 3 + 1)\ bytes$$

$$R = 113\ bytes$$

b) Calculate the blocking factor bfr and the number of file blocks b assuming an un-spanned organization.

- $bfr = INT(B/R)$ for un-spanned organization,
  Where 'B' – Block Size, 'R' – Record Size and 'bfr' – blocking factor

$$bfr = INT\left(\frac{512}{113}\right)$$

$$bfr = INT(4.53)$$

$$\boldsymbol{bfr = 4}$$

- b = (R/bfr) for un-spanned organization,
  Here 'bfr' – Blocking Factor, 'R' – Record Size and 'b' – number of file blocks

  Given:
  $bfr = 4$ [From Above]
  $b = \frac{20000}{4}$
  $\boldsymbol{b = 5000}$

c) Calculate the average time it takes to find a record by doing a linear search on the file if:

i) The file blocks are stored contiguously, and double buffering is used

To compute the average time, we need to take best case and worst case situations.
- In the best case, the record is found in 0th Block.
- In the worst case, the record is found in 4999th Block.
- Thus, average case would be in 2500th Block.

Since it is continuous data, we can use bulk transfer rate. Additionally, there would be a single seek time and rotational delay.
Therefore,

$$Time = s + rd + (b/2 * B/br)$$
$$Time = 30ms + 12.5\ ms + (2500 * (512/409.6))$$
$$Time = (30 + 12.5 + 3125)ms$$
$$Time = 3167.5ms$$
$$\boldsymbol{Time = 3.1675\ s}$$

ii) The file blocks are not stored contiguously.

To compute the average time, we need to take best case and worst case situations.
- In the best case, the record is found in 0th Block.
- In the worst case, the record is found in 4999th Block.
- Thus, average case would be in 2500th Block.

Transfer Time 'tt' = (B / tr) = (512 / 512) = 1 ms

Since they are not stored contiguously, there would be a one seek delay and rotational delay for all the blocks
Therefore,

$$Time = (s + rd + tt) * b/2$$
$$Time = (30ms + 12.5ms + 1ms) * 2500$$
$$Time = 43.5ms * 2500$$
$$Time = 108750\ ms$$
$$\boldsymbol{Time = 108.75\ s}$$

6) Explain data striping in RAID and 7 different RAID based on Granularity and Pattern.

Data Striping is a technique where data is distributed equally over multiple disks to make them appear as a single large, fast disk. This improves overall I/O performance as it allows multiple I/O services in parallel, there by insuring high overall transfer rates. Below figure helps to understand Data Striping better.

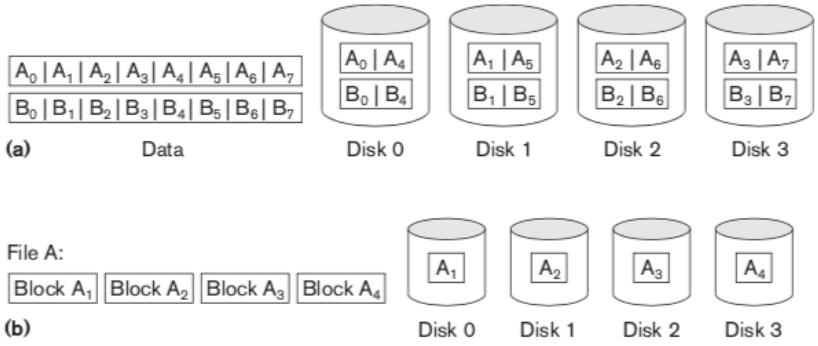


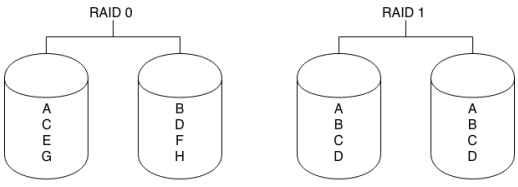
Figure 17.13
Striping of data across multiple disks. (a) Bit-level striping across four disks. (b) Block-level striping across four disks.

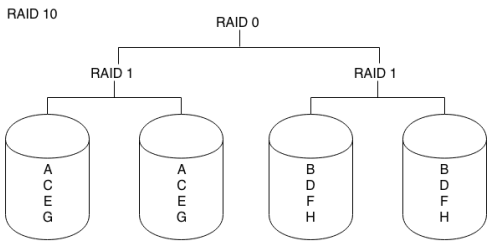
Different Types of RAID (Redundant Arrays of Inexpensive Disks) are:

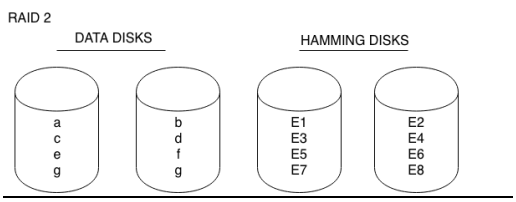
- *RAID 0*
  - It uses data striping.
  - It has no redundant data, thus has best write performance.
  - It splits data evenly across multiple disks.

- *RAID 1*
  - It uses mirrored disks, thus has best read performance.

- *RAID 10*
  - Combination of mirroring from RAID 1 and striping from RAID 0.
  - Redundancy from RAID 1 and performance from RAID 0.







- *RAID 2*

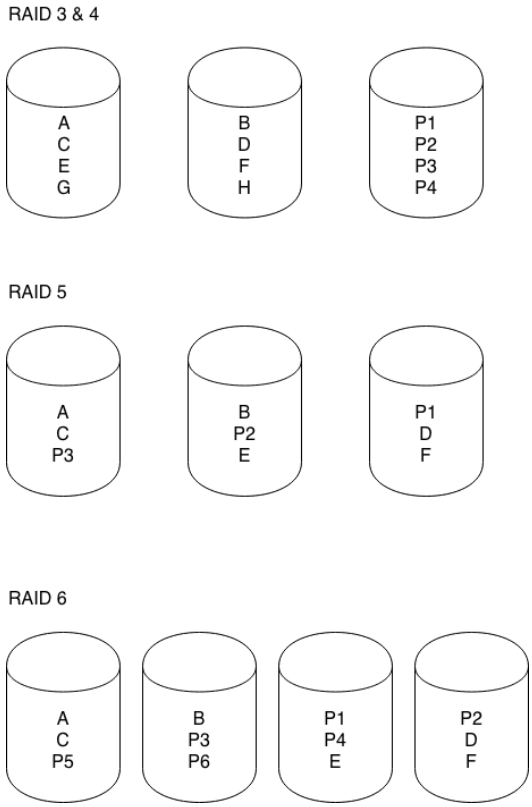
  - Included both error detection and correction.
  - This is with the help of parity bits maintained on separate disks which use Hamming error correction codes.
  - It uses bit level data striping.

- *RAID 3*
  - It uses byte level data striping.
  - Maintains separate disk for Parity for error detection and correction.

- *RAID 4*
  - Same as RAID 3, but has Block level Data striping.

- RAID 5
  - Same as RAID 4, but every disk comprises of one parity block.

- RAID 6
  - Same as RAID 5, with dual parity.
  - It can handle two disk failure.



7) What is collision in Hashing and explain various methods to in collision resolution.

Collision is a scenario where a record which is being inserted generates a hash key whose bucket is already full. To resolve such a collision, we need to insert the following record into a different location. There are many ways to resolve the collision. Some of them are here:

- *Open Addressing –*

The program checks for subsequent empty slot to insert the record proceeding from the collision slot. Thus the next empty slot will be chosen to add the incoming record.

Example, A = {1, 2, 5, 12, 11}
And our Hash(x) = x % 5,
Then,

Bucket 0	**Bucket 1**	**Bucket 2**	**Bucket 3**	**Bucket 4**
5	1	2	12	11

- *Chaining –*
We can say that, every Bucket is added with a link that points to next elements. Thus every bucket follows the pattern of linked lists, and the elements are added based on their hash values.

Example, A = {1, 2, 5, 12, 11}
And our Hash(x) = x % 5,
Then,

Bucket 0	**Bucket 1**	**Bucket 2**	**Bucket 3**	**Bucket 4**
5	1	2	NULL	NULL
NULL	11	12	-	-
-	NULL	NULL	-	-

- *Multiple Hashing –*
Here multiple hash functions are maintained such that if first hash function results with collision, then second one is used. In case, even second fails then a third hash function is chosen if defined or could go ahead with open addressing.

Example, A = {1, 2, 5, 12, 11}
And our Hash1(x) = x % 5,
And our Hash2(x) = (xˆ2) % 5,
Then,

Bucket 0	**Bucket 1**	**Bucket 2**	**Bucket 3**	**Bucket 4**
5	1	2	11	12

Here 12 is added to Bucket 4, with Hash2(12) = 144%5 = 4, Thus there.
But 11 fails with second hash too, thus follows open addressing and is pushed in Bucket 3.

8) Why are disks, not tapes, used to store online database files?

- Tapes are usually used to store backups of database, as they are cheap when compared to disks.
- Access to tapes is very slow when compared to disks.
- To load the data of the tapes, there should be an intervention by an operator to make the data available.
- This cannot be affordable in online database files.
- Thus disks are used to store online database files.

9) Why is accessing a disk block expensive? Discuss the time components involved in accessing a disk block.

Accessing a disk block is expensive due to seek time, rotational delay and block transfer time.

- Seek time (s):
  The time required by the read/write header to position on correct track.

- Rotational Delay (rd):
  The time spent waiting idle for the rotation to reach position where the read/write head starts reading or writing the data in sectors.
  $$rd = \frac{1}{2} * \frac{1}{p}\ min$$
  Where 'p' – revolutions per minute (rpm)

- Block Transfer Time (btt):
  The time spent to read the complete data of a sector.
  $$btt = \frac{B}{tr}\ ms$$
  Where 'B' – Block Size and 'tr' – Transfer rate

10) Explain Reverse engineering and Forward engineering in UML.

Reverse Engineering using UML –

- Creating a conceptual data model with the help of an existing database schema, that is DDL Statements specified.
- Basically reading the DDL file and generating a conceptual data model using the same.
- Rational Rose Data Modeler has a reverse engineering wizard for the same.

Forward Engineering using UML –

- With the help of data model, build the database from the scratch for a specified Database Management System.
- Rational Rose Data Modeler has a forward engineering wizard for the same.

References:

- *https://www.techwalla.com/articles/list-of-advantages-of-uml*
- *https://creately.com/blog/diagrams/advantages-and-disadvantages-of-uml/*
- *www.just.edu.jo/~amerb/teaching/1-9-10/cs728/ch13.ppt*
- *https://www.thegeekstuff.com/2011/11/raid2-raid3-raid4-raid6*
- *Fundamentals of Database Systems, Sixth Edition by Elmasri/Navathe.*