Database Security

Database Security ~ Scope

- More and more sensitive data is stored on computers.
- As sophistication and the number of users grow, the data becomes more vunerable to unwanted access or corruption.
- In this lecture we will look at:
  - Threats to security
  - Threats to integrity
  - Resolution of Problems

Database Security : Introduction

- Data is a valuable resource that must be strictly controlled and managed.
  - Part or all of the corporate data may have strategic importance to an organization.
- Data therefore represents an essential corporate resource that should be properly secured using appropriate tools, i.e. must be kept **secure** and **confidential.**
- Among the services of a DBMS include **authorization services** which are mechanisms to ensure that only authorised users access the database.

Database Security

<u>SECURITY</u>

- Refers to protection of the database from **<u>unauthorised</u>** users
  - How can we stop people accessing data when they shouldn't?
- Ensures that users are **<u>allowed</u>** to do the things they are trying to do
- Example: Use of passwords:
  - A password should be like a toothbrush.
  - Use it every day; change it regularly; and DON'T share it with anyone (O'Reilly and Associates)

Database Security

- **Database Security** refers to the mechanisms that protect the database against intentional or accidental threats.
  - Computer security starts with a set of well-designed set of controls.
- Database security aims to minimize losses caused by anticipated events in a cost effective manner without unduly constraining the users

Database Security

- We consider security in relation to the following situations:
  - Theft and fraud
  - Loss of confidentiality
  - Loss of privacy
  - Loss of integrity
  - Loss of availability.

Database Security

- **Theft and fraud** do not necessarily alter data, as is the case with loss of confidentiality and loss of integrity.
- **Confidentiality** refers to the need to maintain secrecy over data, usually only that which is critical to the organisation.
  - Loss of confidentiality could result to loss of competitiveness.
- **Privacy** refers to the need to protect data about individuals.
  - Loss of privacy could lead to legal action being taken against the organisation.

Database Security

- **Loss of integrity** results to invalid or corrupted data which may seriously affect the operations of the organisation.
  - Integrity involves pprotecting the database from authorised users by ensuring that what users are trying to do is correct.
- **Loss of availability** means that data, or the system, or both cannot be accessed, which can seriously affect an organisations financial performance.

Database Security

Threat

- Threat is any situation or event, whether intentional or accidental, that may adversely affect a system and consequently the organization.
- The harm/threat may be tangible, such as loss of hardware, software, or data, or intangible, such as loss of credibility or client confidence.

Database Security

Threat	Theft and Fraud	Loss of confidentiality	Loss of privacy	Loss of Integrity	Loss of availability
Using another persons' means to access	√	√	√		
Unauthorized amendment or copying of data	√			√	
Program alteration	√			√	√
Illegal entry by hacker	√	√	√		
Blackmail	√	√	√		
Wire tapping	√	√	√		
Creating a `trapdoor` into system	√	√	√		
Theft of data, program and equipment	√	√	√		

Database Security

Threat	Theft and Fraud	Loss of confidentiality	Loss of privacy	Loss of Integrity	Loss of availability
Failure of security mechanism giving greater access than normal		√	√	√	
Staff shortages or strikes				√	√
Inadequate staff training		√	√	√	√
Viewing and disclosing unauthorized data	√	√	√		
Electronic interference and radiation				√	√
Data corruption due to power loss or surge (hardware)				√	√
Fire, flood, bomb (Hardware)				√	√

Database Security

Threat	Theft and Fraud	Loss of confidentiality	Loss of privacy	Loss of Integrity	Loss of availability
Physical damage of equipment (Hardware)				√	√
Breaking cables or disconnection of cables				√	√
Introduction of viruses				√	√
Inadequate policies and procedures that allow a mix of confidential and normal output.					√

Database Security

- Recovery from the threats listed previously depend upon a number of factors such as when the last backups were taken and the time needed to restore the system.

Database Security

Summary of Potential threats to computer systems

Hardware

- Fire, flood, bombs
- Data corruption due to power loss or surge
- Failure of security mechanisms giving greater access
- Theft of equipment Physical damage to equipment
- Electronic interference and radiation

Database Security

Summary of Potential threats to computer systems

DBMS and Application software

- Failure of security mechanism giving greater access
- Program alteration
- Theft of programs

Communication networks

- Wire tapping
- Breaking or disconnection of cable

Database Security

Summary of Potential threats to computer systems

Databases

- ~Unauthorized amendment or copying of data
- ~Theft of data
- ~Data corruption due to power loss or surge

Data/ Database Administrator

- Inadequate security policies and procedure.

Database Security

Summary of Potential threats to computer systems

Programmers / Operators

- Creating trapdoors
- Program alteration (such as creating software that is insecure)
- Inadequate staff training
- Inadequate security policies and procedures
- Staff shortages or strikes

Database Security

Summary of Potential threats to computer systems

Users

- Using another persons' means of access
- Viewing and disclosing unauthorized data
- Inadequate staff training
- Illegal entry by hacker
- Blackmail
- Introduction of viruses

COUNTER MEASURES

- Range from physical controls to administrative procedures.

1. **Authorization**

- This is the granting of a right or privileges that enables a subject to have legitimate access to a system or a systems' object.
- The authentication controls (access controls) can be built into the software and govern not only what system or object a specified user can access but also what the user may do with it.

COUNTER MEASURES

a. **Authentication** – is a mechanism that determines whether a user is who he / she claim to be.

- The system administrator is responsible for allowing users to have access to the computer systems by creating individual user accounts.
- This procedure allows authorized use of a computer system but does not necessarily authorize access to the DBMS or any associated application programs.
  - Database administrators set up individual user accounts and passwords using DBMS to give users the right to use the DBMS.

COUNTER MEASURES

b. **Privileges** – Once a user is given permission to use a DBMS, various privileges may also be automatically associated with it; such as right to access or create certain database objects like relations, views etc.

c. **Ownership and privileges** – Some objects in the DBMS are owned by the DBMS itself, usually in the form of a specific super user such as a DBA.

   - This gives the owner all appropriate privileges on the objects owned and can assign appropriate privileges for the object.
   - Example: although a user owns a view, he/she may be authorized only to query the view.

COUNTER MEASURES

2. Views (Sub-schemas)

- A view is a dynamic result of one or more relational operations operating on the base relations to produce another relation.
- A view is a virtual relation that does not actually exist in the database, but is produced upon request by a particular user, at the time of the request.
- The view mechanism provides a powerful and flexible security mechanism by hiding parts of the database from certain user.

COUNTER MEASURES

- The user is not aware of the existence of any attributes or rows that are missing from the view.
- A view can be defined over several relations with a user being granted the appropriate privilege to use it, but not to use the base relations.
- Using view is more restrictive than simply having certain privileges granted to a user on the base relation(s).

COUNTER MEASURES

3. Backup and Recovery

- The process of periodically taking a copy of the database and log file (and possibly programs) on to storage media is called backing up.
- DBMS should provide backup facilities to assist with the recovery of a database following failures.
- **Journaling** is the process of keeping and maintaining a log file (or journal or a track) of all changes made to the database (current status of transactions) to enable the DB to be recovered to its last known consistent state using backup copies and log files in the event of failure.

COUNTER MEASURES

4. **Integrity**

- Integrity constraints contribute to maintaining a secure database system by preventing data from becoming invalid., and hence giving misleading or incorrect results.

5. **Encryption**

- This is the encoding of the data by a special algorithm that renders the data unreadable by any program without the decryption key.

COUNTER MEASURES

- If a DB system holds particularly sensitive data , it may be deemed necessary to encode it as a precaution against possible external threats or attempts to access it.
- Some DBMS's provide an encryption facility for that purpose.
- Encryption protects also data transmitted over communication lines.

COUNTER MEASURES

6. **RAID(Redundant Array of Independent Disks)**

- The hardware that the DBMS is running on must be fault tolerant, meaning that the DBMS should continue to operate even if one of the hardware components fails.
- This suggests having redundant components that can be seamlessly integrated into a working system whenever there is one or more components failures.

COUNTER MEASURES

- The main hardware components that should be fault-tolerant include disk drives, disk controllers, CPU, power supplies and cooling fans.
- RAID works on having a large disk array comprising an arrangement of several independent disks that are organized to improve reliability and at the same time increase performance.
- Performance is increased through data striping where data is segmented into equal- size partitions (the striping unit) which are temporarily distributed across multiple disks.

COUNTER MEASURES

- This gives the appearance of a single large, fast disk where in actual fact the data is distributed across several smaller disks.
- **Striping** improves overall I/O performance by allowing multiple I/O's to be serviced in parallel.
- At the same time, data stripping also balances the load among disks.
- Reliability is improved through storing redundant information across the disks using a parity scheme or an error-correcting scheme.

COUNTER MEASURES

- In a **parity** scheme, each byte may have a parity bit associated with it that records whether the number of bits in the byte that are set is even or odd.
  - If the number of bits in the byte becomes corrupted, it will not match the stored parity
- **Error-correcting** schemes store two or more additional bits, and can reconstruct the original data if a single bit becomes corrupt.
- This schemes can be used through stripping bytes across disks.

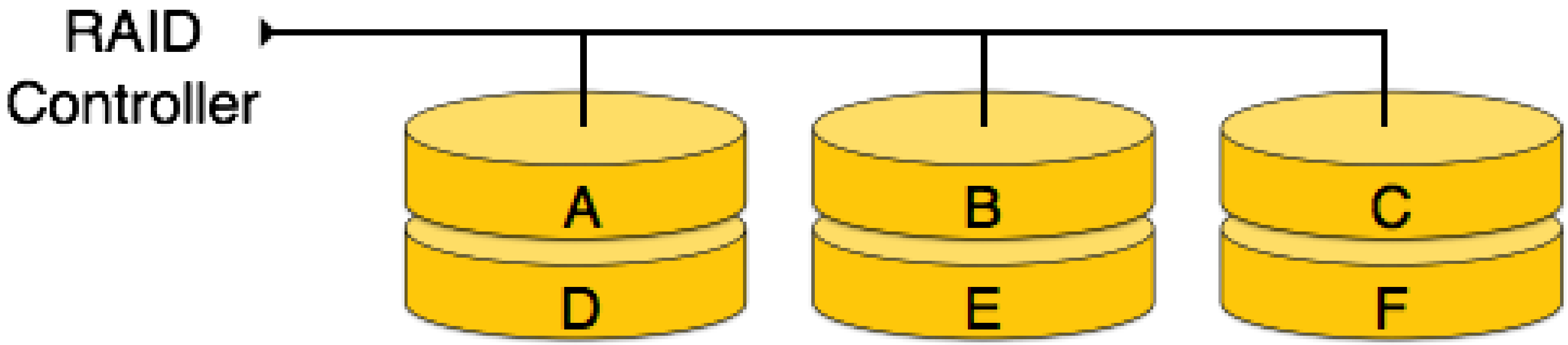
RAID Levels

RAID 0 – Non-redundant – This level maintains no redundant data and so has the best write performance since updates do not have to be replicated.

- Data stripping is performed at the level of blocks.
- In this level, a striped array of disks is implemented.
- The data is broken down into blocks and the blocks are distributed among disks.
- Each disk receives a block of data to write/read in parallel.
- It enhances the speed and performance of the storage device.
- There is no parity and backup in Level 0.

RAID Levels

RAID O – Non-redundant



RAID Levels

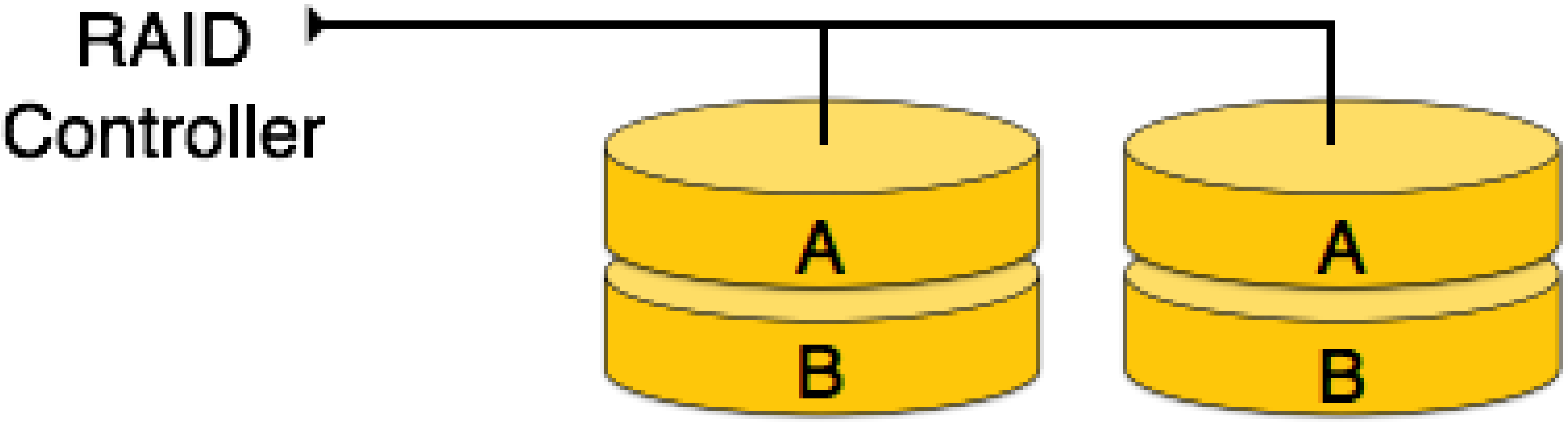
RAID I – Mirrored – Maintains (mirrors) two identical copies of the data across different disks.

- When data is sent to a RAID controller, it sends a copy of data to all the disks in the array.
- RAID level 1 provides 100% redundancy in case of a failure.
- It's the most expensive storage solution

RAID O + I – Non-redundant and Mirrored – Combines striping and mirroring.

RAID Levels

RAID I – Mirrored



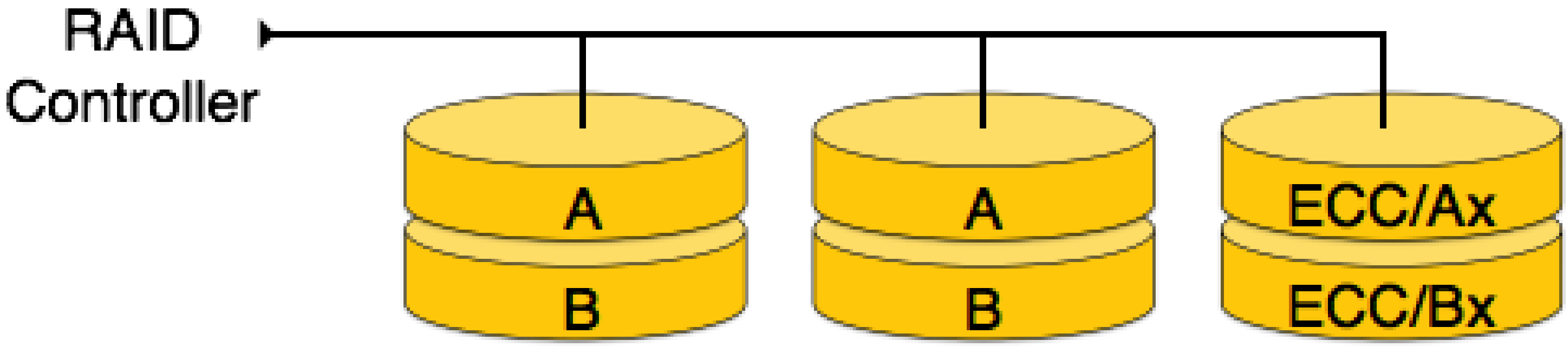
RAID Levels

RAID 2 – Memory-style Error correcting codes:

- The striping unit in this level is a single bit and hamming codes are used as the redundancy scheme.
  - RAID 2 records Error Correction Code using Hamming distance for its data, striped on different disks.
  - Like level 0, each data bit in a word is recorded on a separate disk and ECC codes of the data words are stored on a different set disks.
  - Due to its complex structure and high cost, RAID 2 is not commercially available.

RAID Levels

RAID 2 – Memory-style Error correcting codes:

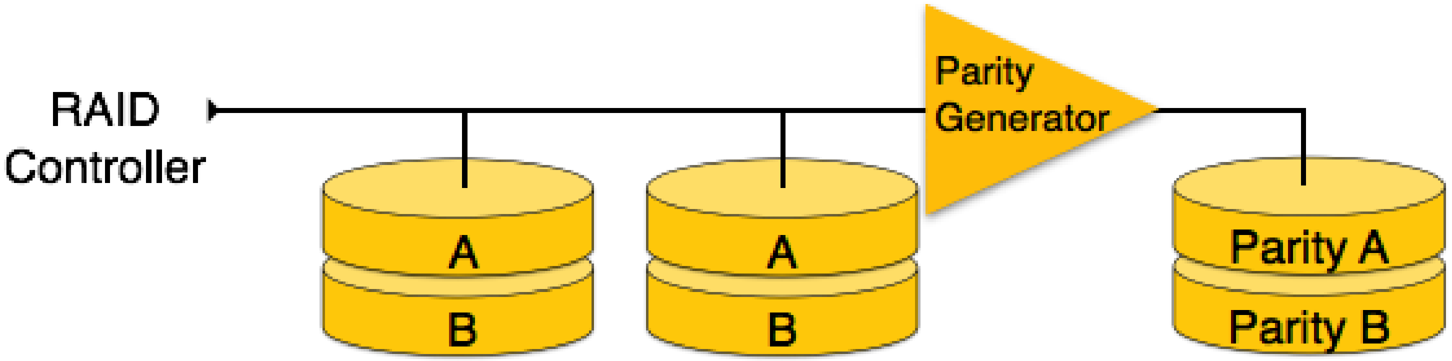


RAID Levels

- **RAID 3 – Bit~ Interleaved Parity** – Provides redundancy by storing parity information on a single disk in the array which can be used to recover the data on the disks should they fail.
  - RAID 3 stripes the data onto multiple disks.
  - The parity bit generated for data word is stored on a different disk.
  - This technique makes it to overcome single disk failures.

RAID Levels

RAID 3 – Bit~ Interleaved Parity

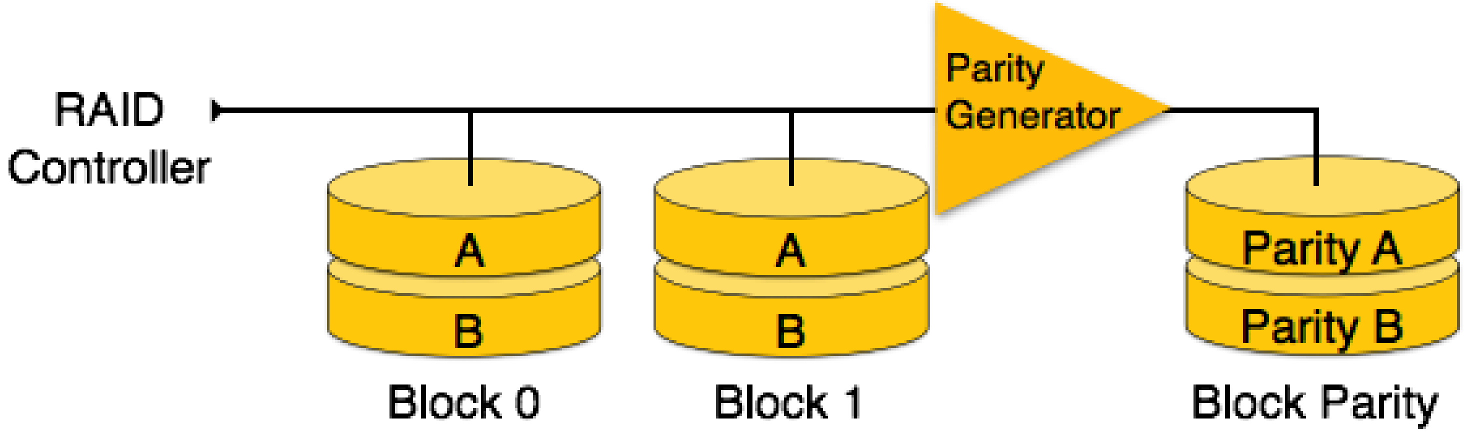


RAID Levels

- **RAID 4 – Block Interleaved Parity** – In this level the striping unit is a disk block and a parity block is maintained on a separate disk for corresponding blocks from a number of other disks.
  - If one disk fails, the parity block can be used with the corresponding blocks from the other disks to restore the blocks of the failed disk.
  - **Process:** an entire block of data is written onto data disks and then the parity is generated and stored on a different disk.
    - Note that level 3 uses byte-level striping, whereas level 4 uses block-level striping. Both level 3 and level 4 require at least three disks to implement RAID.

RAID Levels

- RAID 4 – Block Interleaved Parity



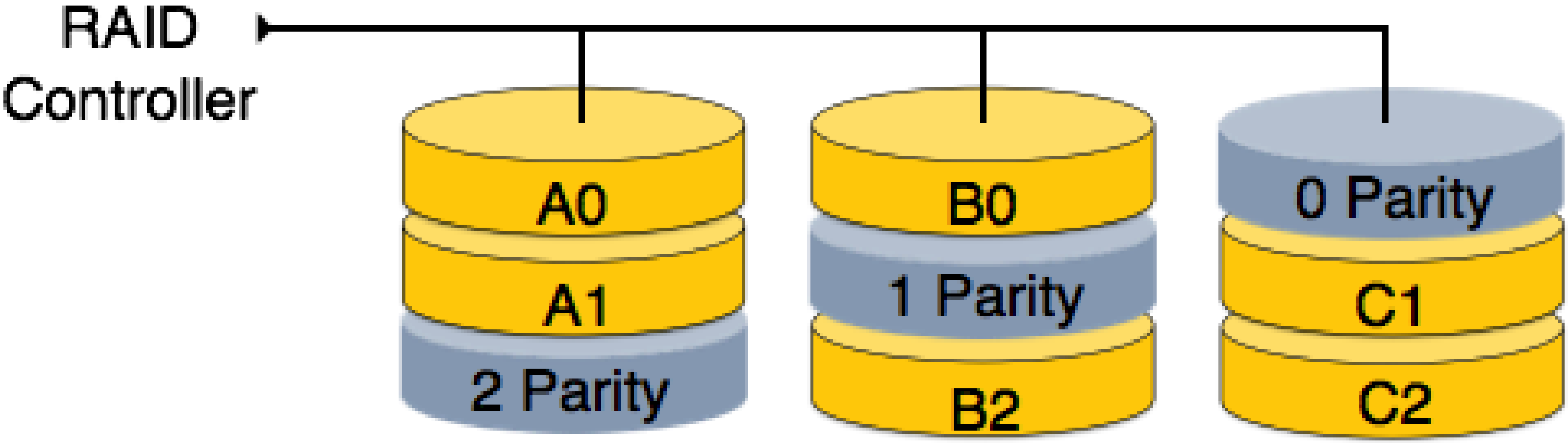
RAID Levels

RAID 5 – Block~ Interleaved Distributed Parity – Uses a parity data for redundancy in a similar way to RAID 3 but strips the parity data across all the disks.

- RAID 5 writes whole data blocks onto different disks, but the parity bits generated for data block stripe are distributed among all the data disks rather than storing them on a different dedicated disk.

RAID Levels

RAID 5 – Block~ Interleaved Distributed Parity

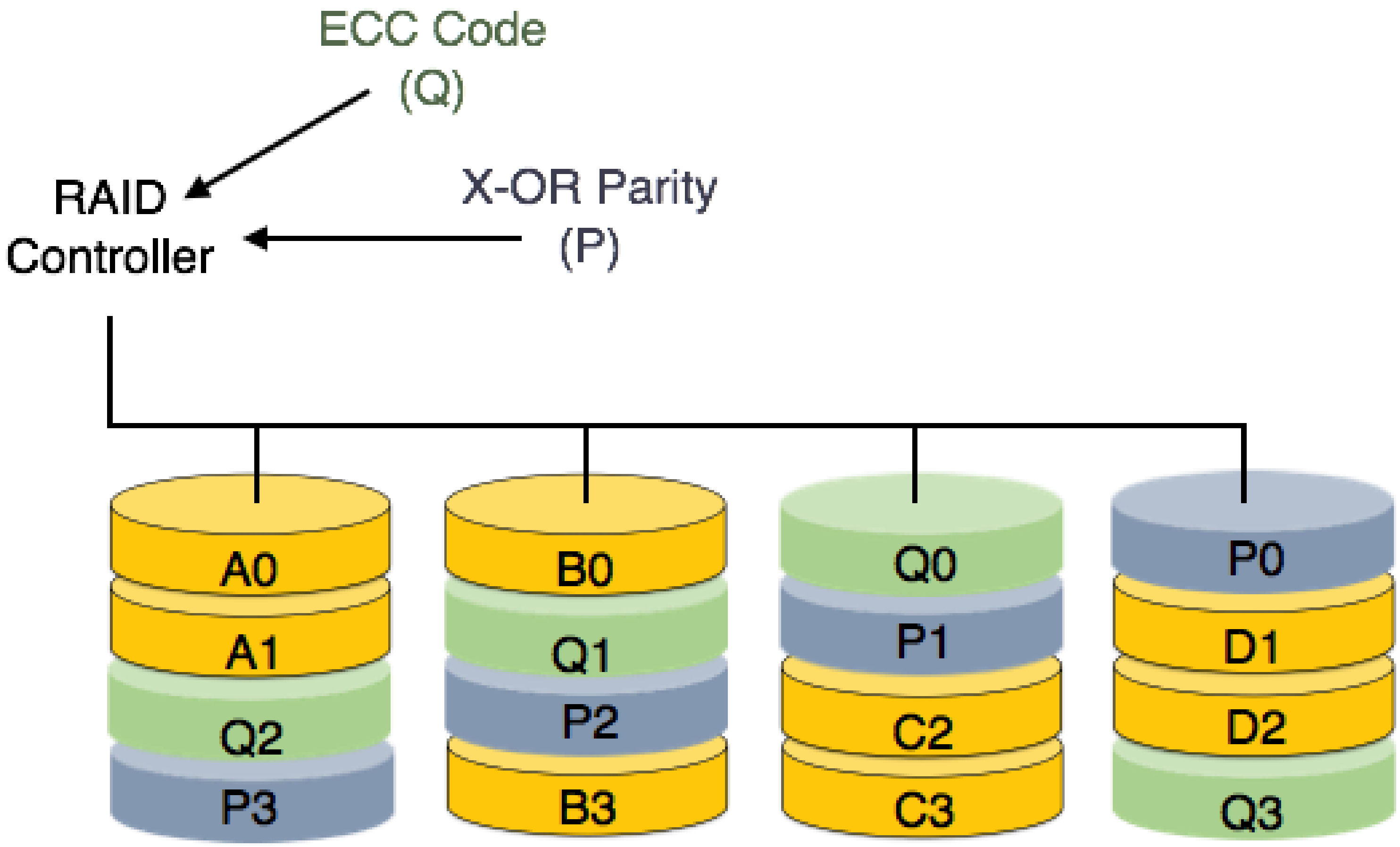


RAID Levels

- **RAID 6 – P+Q Redundancy** – Similar to RAID 5 but additional redundant data is maintained to protect against multiple disk failures.
- Error correcting codes are used instead of using parity.
  - In this level, two independent parities are generated and stored in distributed fashion among multiple disks.
  - Two parities provide additional fault tolerance.
  - This level requires at least four disk drives to implement RAID.

RAID Levels

RAID 6 – P+Q Redundancy


ECC Code
(Q)
RAID
Controller
X-OR Parity
(P)
A0
A1
Q2
P3
B0
Q1
P2
B3
Q0
P1
C2
C3
P0
D1
D2
Q3