# Multimedia Storage Performance Simulation Using RAID


Team Members :

1-Saeed Mamdouh 23030086

2-Seif-eldeen Ahmed Lotfy 23030241

3-Youssef Ahmed Hassan 23030218

4-Omar Ahmed Omar 23030122

5-Khaled Atef Mahmoud 23030064

6- Mohamed Waleed Abdelgawad 23030176

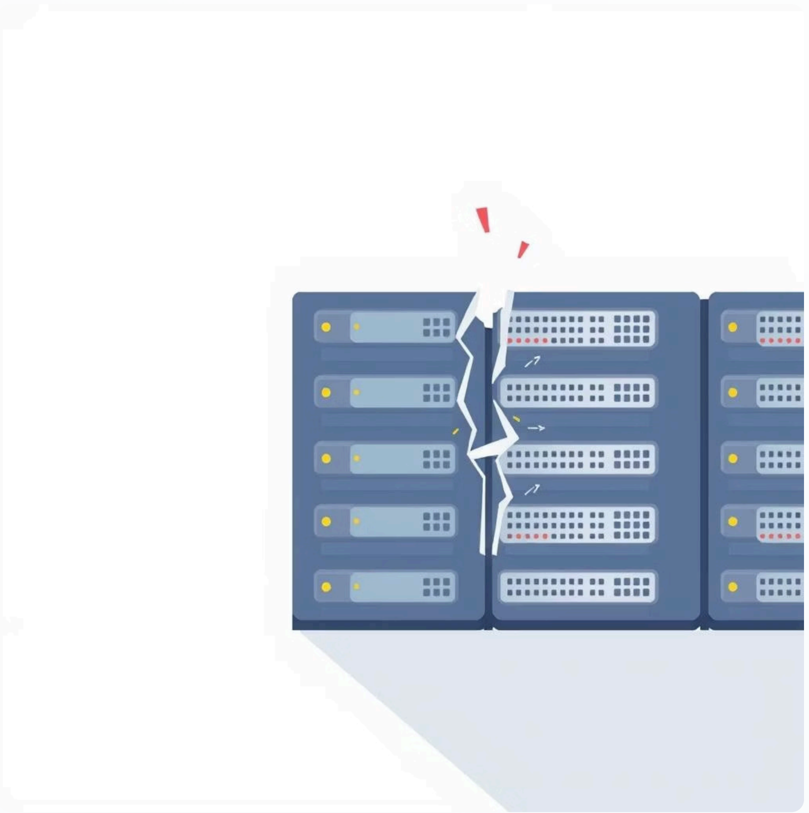
7-Mayada Salah 23030198


Course: Information Storage & Management

# Project Overview and Problem Statement

Modern multimedia applications demand robust and efficient storage solutions. Traditional storage systems often struggle with the high I/O throughput and low latency requirements of streaming video, large image libraries, and interactive content.

This project investigates the performance implications of various RAID configurations for multimedia storage, aiming to identify optimal strategies for data redundancy and access speed through simulation.



# Project Objectives

### Evaluate Performance

Quantify the I/O performance of RAID 0, RAID 1, and RAID 5 in multimedia storage scenarios.

### Analyze Storage Efficiency

Assess the storage space utilization and overheads associated with different RAID levels.

### Identify Optimal Configurations

Determine the most suitable RAID levels for specific multimedia storage workloads.

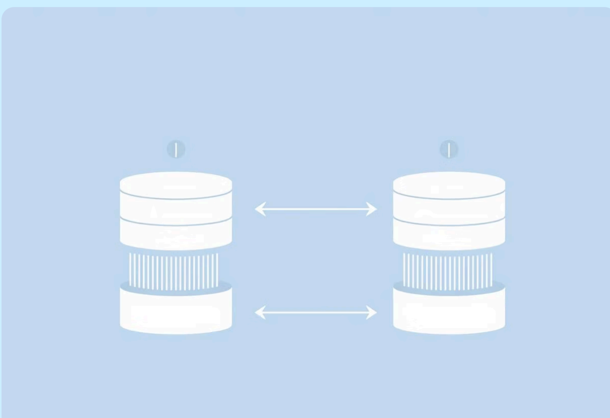
### Develop Simulation Model

Create a Python-based simulation framework for extensible RAID performance analysis.

# RAID Levels Overview

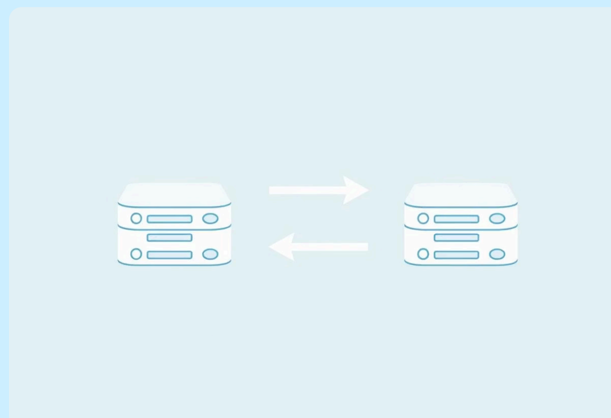
### RAID 0 (Striping)

Data is striped across multiple disks, offering high performance but no fault tolerance.



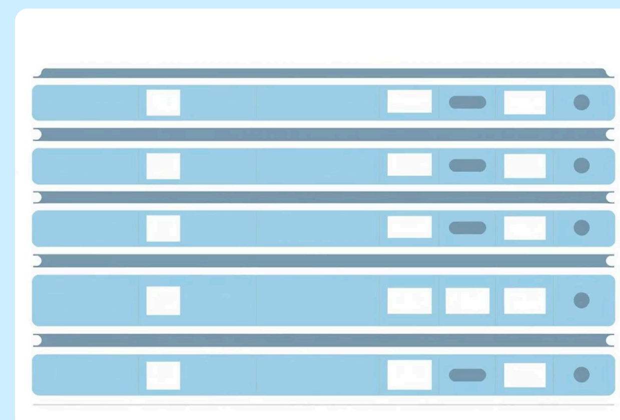
### RAID 1 (Mirroring)

Data is duplicated across two disks, providing excellent fault tolerance at the cost of storage efficiency.



### RAID 5 (Striping with Parity)

Data is striped with distributed parity, balancing performance and fault tolerance efficiently.



# Simulation Methodology Using Python

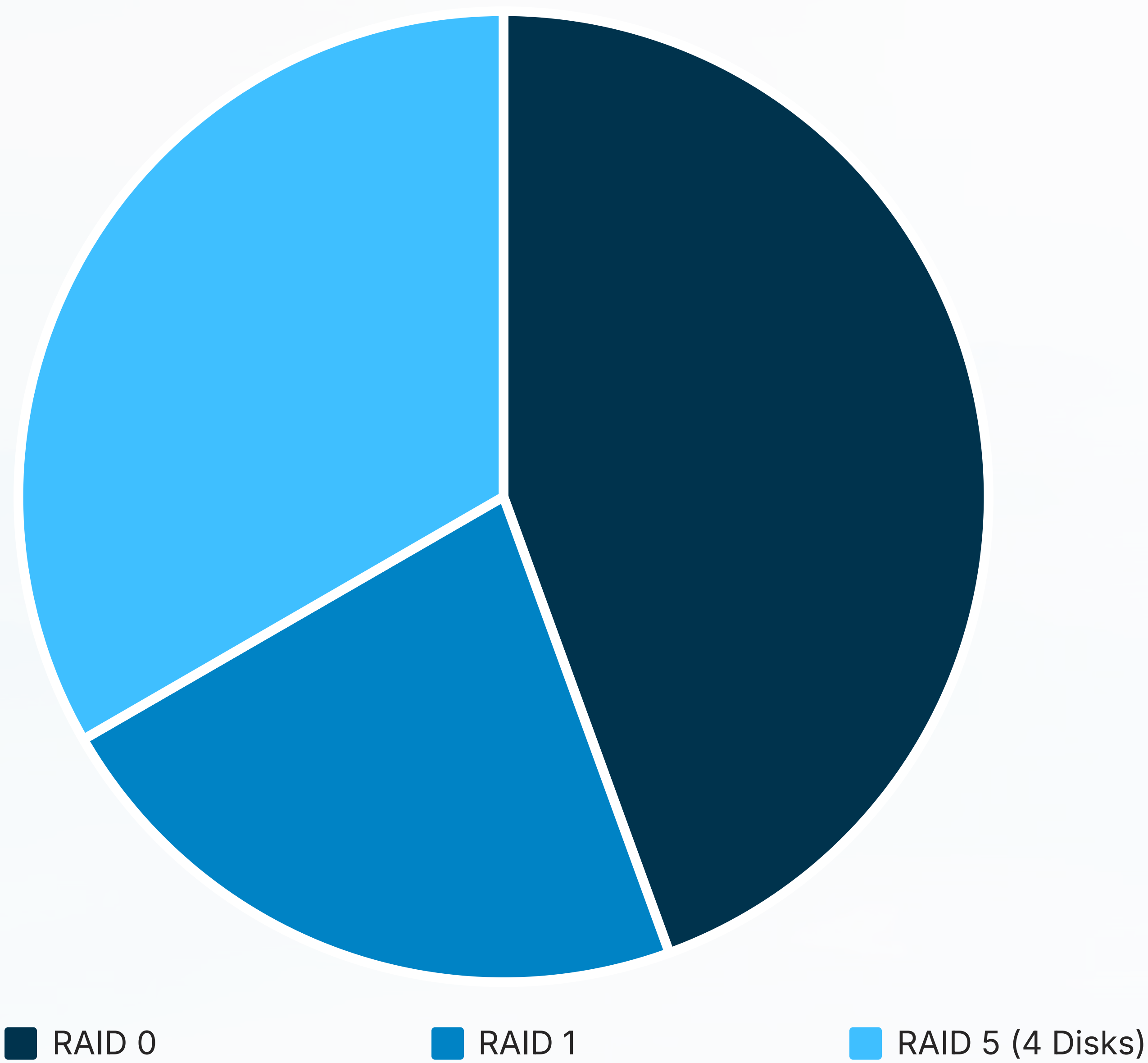
Our simulation framework, developed in Python, models disk I/O operations, data distribution, and error handling for various RAID configurations. It allows for configurable parameters such as disk latency, throughput, and seek times to accurately reflect real-world scenarios.

- Disk I/O operations modeled.
- Configurable disk parameters.
- Workload simulation for multimedia access patterns.
- Performance metrics captured: IOPS, latency, throughput.

```python
def simulate_raid_read(raid_level):
    """
    Simulates RAID read operation time.
    Returns read time in milliseconds.
    """

    base_speed = 150  # Base read speed in MB/s

    # RAID 1 has faster reads due to mirroring
    if raid_level == "RAID 1":
        base_speed *= 1.2

    # RAID 5 has slightly improved read speed
    elif raid_level == "RAID 5":
        base_speed *= 1.1

    # Calculate read time for 500 MB
    read_time = 500 / base_speed

    # Artificial delay to simulate real I/O
    time.sleep(read_time / 10)

    # Return time in milliseconds
    return read_time * 1000


def simulate_raid_write(raid_level):
    """
    Simulates RAID write operation time.
    Returns write time in milliseconds.
    """

    base_speed = 120  # Base write speed in MB/s
    overhead = 1.0    # Write overhead factor
```

# Storage Space Distribution by RAID Level

Understanding how each RAID level utilizes physical disk space is crucial for cost-effective storage planning, especially for large multimedia archives.




This chart illustrates the effective storage capacity for a hypothetical 4-disk array, highlighting the varying overheads.

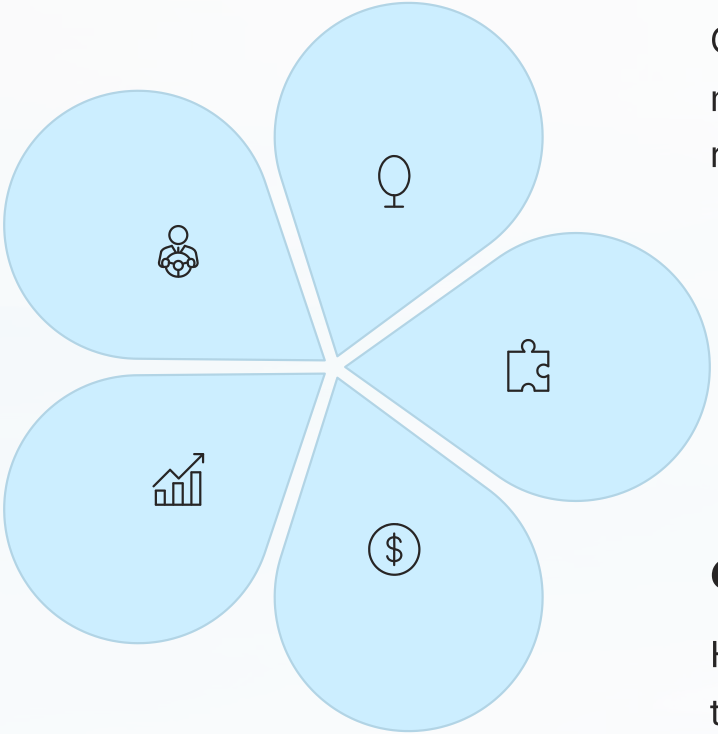
# Pie Chart Analysis and Interpretation

### RAID 0

Achieves 100% efficiency, as all disk space is used for data, but lacks redundancy.

### Data Integrity

Efficiency trade-offs are directly linked to the level of data protection against disk failures.



### RAID 1

Offers 50% efficiency due to mirroring, meaning half the space is for redundancy.

### RAID 5

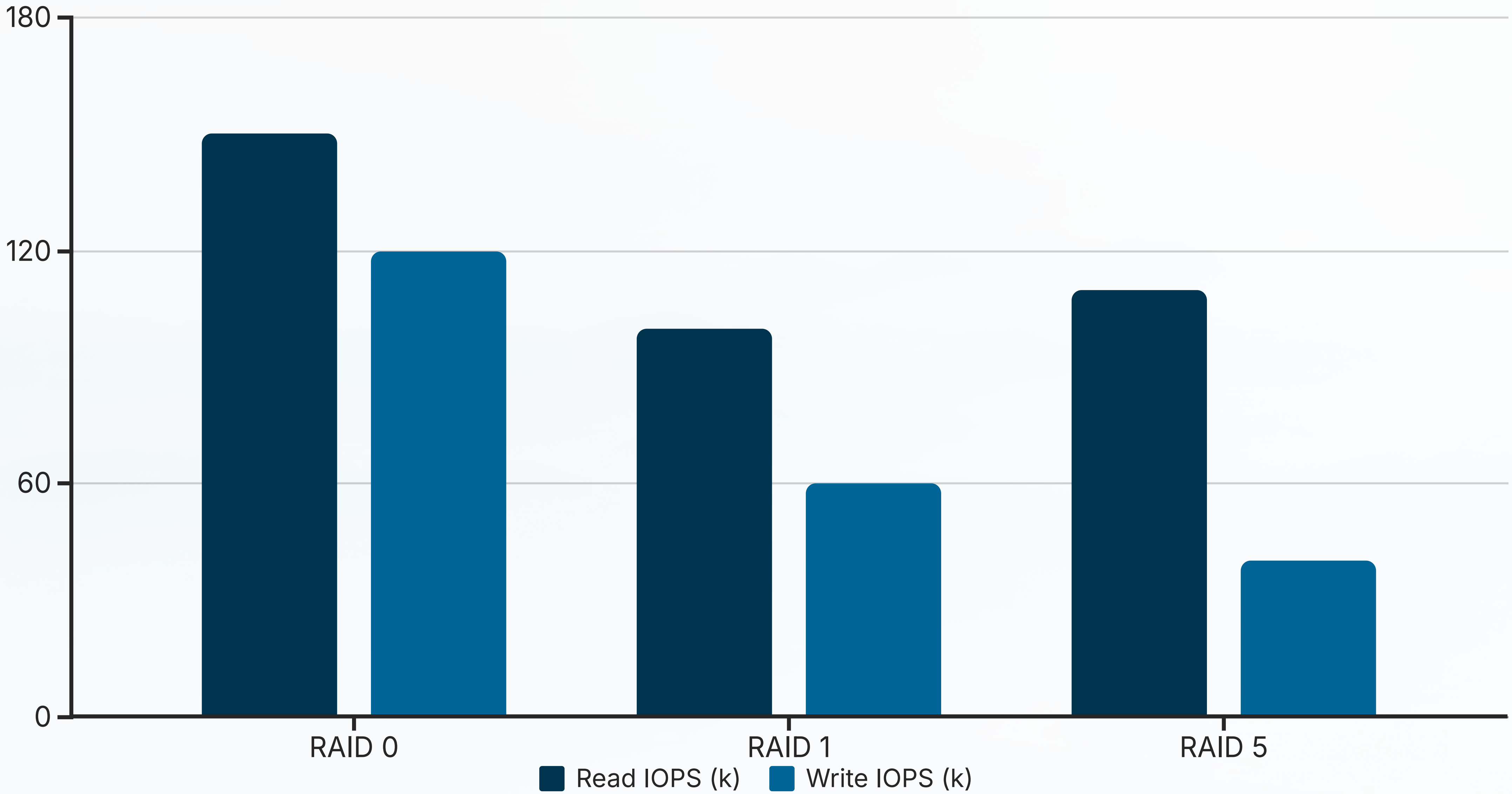
For a 4-disk array, achieves 75% efficiency (N-1 disks), balancing space and fault tolerance.

### Cost Implications

Higher redundancy (RAID 1) directly translates to higher hardware costs per usable GB.

# RAID Performance Comparison: Multimedia Workload

Performance metrics, specifically IOPS (Input/Output Operations Per Second) and throughput, are critical for multimedia applications. This bar chart compares the simulated read/write performance across different RAID levels.




Simulated performance for a mixed multimedia workload, showing relative strengths.

# Bar Chart Analysis and Interpretation

01

## RAID 0 Dominance

Exhibits the highest read and write IOPS due to parallel data access without parity overhead, ideal for performance-critical, non-redundant data.


02

## RAID 1 Read Performance

Shows strong read performance, often comparable to RAID 0, as data can be read from either mirrored disk. Write performance is lower due to dual writes.



03

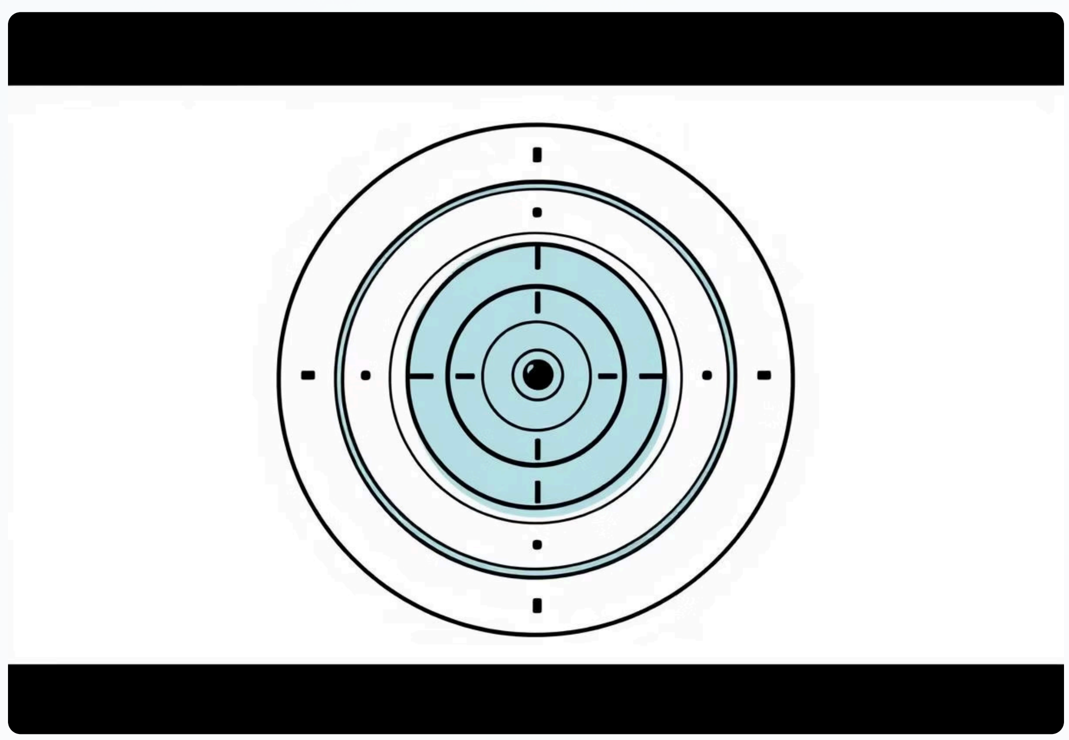
## RAID 5 Trade-offs

Provides good read performance but significantly lower write performance due to parity calculations and writes, making it less suitable for heavy write multimedia workloads.

04

## Workload Matching

The optimal RAID choice depends heavily on the specific multimedia workload – read-heavy vs. write-heavy applications have different requirements.



# Tools and Technologies, Key Findings, and Future Work

## Tools & Technologies

- Python: Core simulation logic
- NumPy: Numerical operations
- Matplotlib: Data visualization
- Jupyter Notebooks: Interactive analysis

## Key Findings

- RAID 0 offers peak performance but zero fault tolerance, suitable for scratch disks.
- RAID 1 provides high data availability and good read performance, ideal for critical, smaller datasets.
- RAID 5 balances capacity and protection, but write performance can be a bottleneck for multimedia editing.

## Future Work

- Integrate caching mechanisms into the simulation.
- Model more complex RAID levels (e.g., RAID 6, RAID 10).
- Validate simulation results against physical hardware benchmarks.
- Incorporate network storage (NAS/SAN) performance factors.