RELIABILITY IN MODERN CLOUD SYSTEMS

Summer 2025

LOGISTICS

ASSIGNMENT 3

- Grades have been pushed to personal forks

ASSIGNMENT 4

- Check-In #3 on Monday
- Assigned Time Slots for Efficiency:
  - 2:00-2:10: Lukas + Bastien
  - 2:10-2:20: Paritosh
  - 2:20-2:30: Jinhao + Bekhrouz
  - 2:30-2:40: Ali Fahad + Zawayar
  - 2:40-2:50: Aiman + Asim
  - 2:50-3:00: Felix + Marius
  - 3:00-3:10: Talal + Umair

CLOUD INFRASTRUCTURE DISCUSSION

DISCUSSION THEMES

- Resource utilization is one key goal of cloud infrastructure providers? What are some other goals that cloud providers care about?

- How to decide the resource assignment for different workloads?

DISCUSSION THEMES

- Resource utilization is one key goal of cloud infrastructure providers? What are some other goals that cloud providers care about?

DISCUSSION THEMES

- Resource utilization is one key goal of cloud infrastructure providers? What are some other goals that cloud providers care about?

Application performance should not be affected by utilization goals. Cloud providers also want to be sustainable to not harm the environment (any further)

DISCUSSION THEMES

- How to decide the resource assignment for different workloads?

DISCUSSION THEMES

- How to decide the resource assignment for different workloads?

Use information learned from workloads to make predictions about what would be the best assignment for the new incoming workloads.

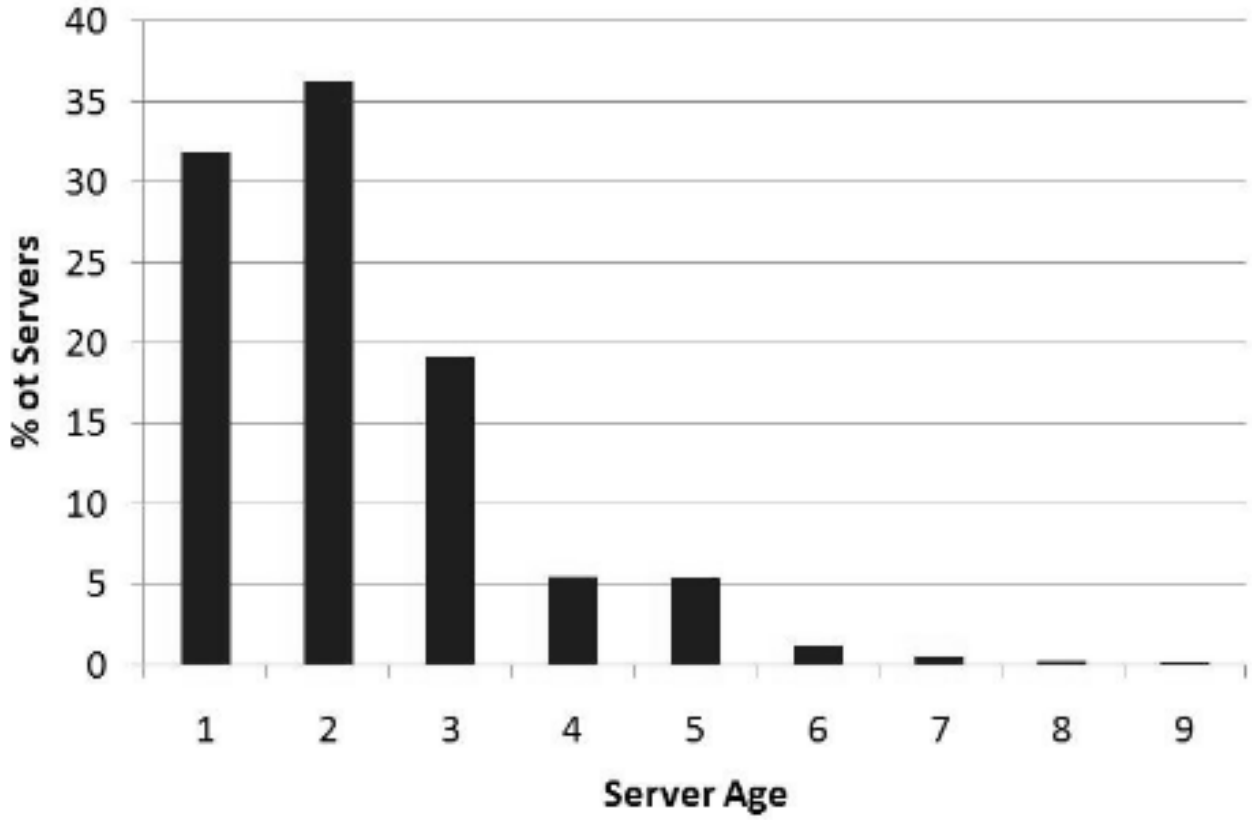
PAPER SUMMARY

PAPER SUMMARY

- Improve resource allocation based on models learned from previous execution data
- Continuously collect metrics from workloads that are running
- Using the metrics, train simple predictive models of how much resources would in reality be needed for a workload
- Inform the schedulers in the infrastructure to make use of this data to perform resource allocation

HARDWARE RELIABILITY

SERVER AGE PROFILE



Microsoft Datacenter, 2010

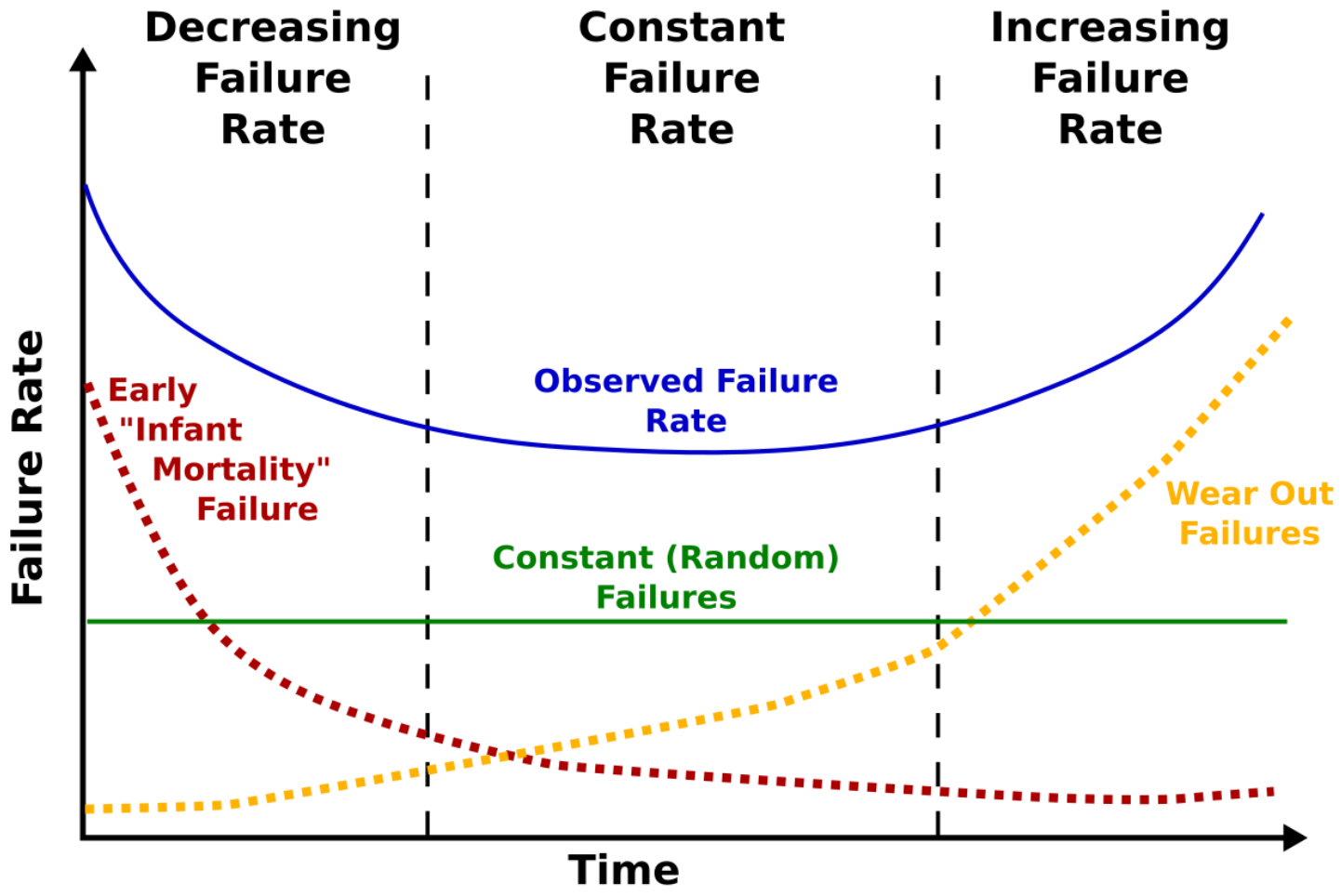
Characterizing Cloud Computing Hardware Reliability, SoCC'10, Vishwanath et al

HARDWARE FAILURES

Two Types

- Repair Events
  - Failure bricks the hardware
  - Usually fixed with a repair
- Silent Data Corruptions
  - Failure does not brick the hardware
  - Not necessarily fixed with repair

INDIVIDUAL COMPONENT FAILURE RATE



https://en.wikipedia.org/wiki/Bathtub_curve

DISK FAILURES

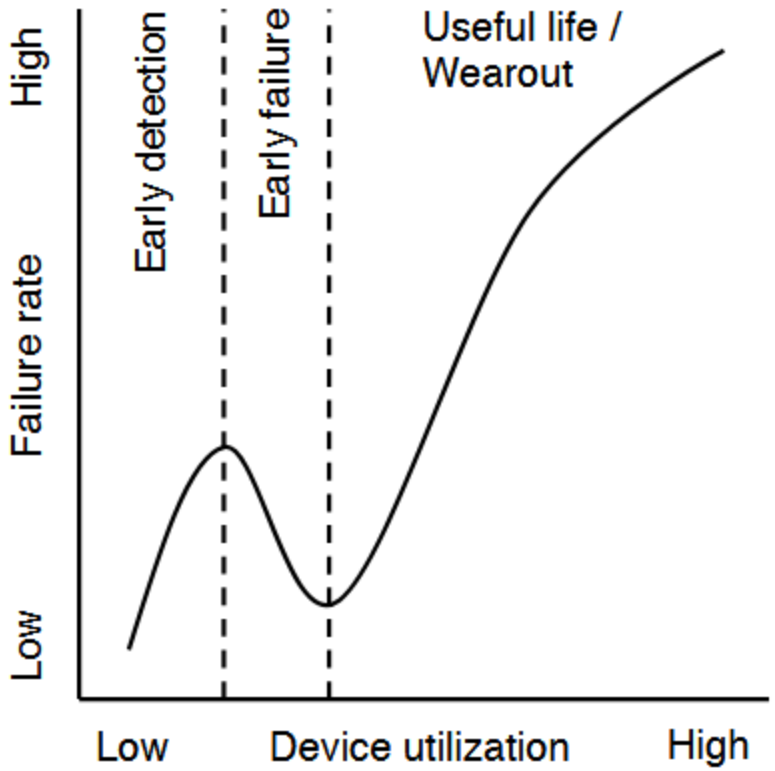
Survey of hardware failures in a Microsoft Datacenter

- 2.7% of all disks are replaced every years
- Majority of repairs were for hard disk failures
- Disk failures follow a a typical bathtub curve

SDD FAILURES

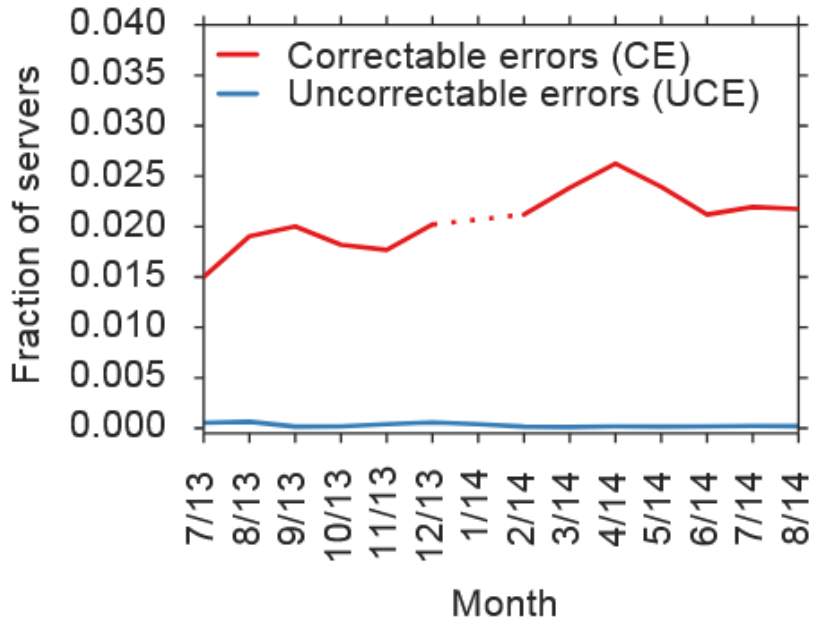
- In addition to the early failure period, SSDs also have an early detection period

- Errors are detected in this period and problematic cells are removed by the controller to improve the reliability



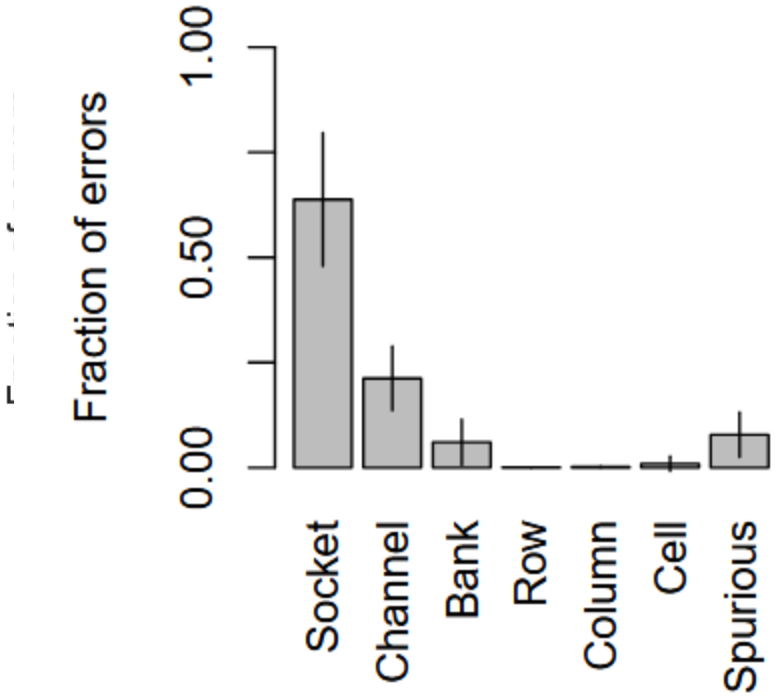
MEMORY FAILURES

- Correctable errors are frequent
- Uncorrectable errors crash the machine; correctable errors don't
  - Uncorrectable error requires repair



MEMORY FAILURES

- Correctable errors are frequent
- Uncorrectable errors crash the machine; correctable errors don't
  - Uncorrectable error requires repair
- Errors caused from DRAM failure or memory controller in the processor
  - Non-DRAM memory errors are not isolated events and cause a denial of service attack on the server



NETWORK FAILURES

Category	Fraction	Description
Maintenance	17%	Routine maintenance (for example, upgrading the software and firmware of network devices).
Hardware	13%	Failing devices (for example, faulty memory modules, processors, and ports).
Misconfiguration	13%	Incorrect or unintended configurations (for example, routing rules blocking production traffic).
Bug	12%	Logical errors in network device software or firmware.
Accidents	11%	Unintended actions (for example, disconnecting or power cycling the wrong network device).
Capacity planning	5%	High load due to insufficient capacity planning.
Undetermined	29%	Inconclusive root cause.

Table 5.1: Common root causes of intra data center network incidents at Facebook from 2011 to 2018.

Large Scale Studies of Memory, Storage, and Network Failures in a Modern Data Center, PhD Thesis, Justin Meza

CPU OVERHEATING

Overheating can cause temperatures for chips to rise

- Operating at a higher temperature can damage the chip
- Reduces the lifespan of the chip

Reasons for overheating

- Broken fan/Inadequate cooling
- Overclocking + High Power Draw to be more efficient
- Dirty environment (eg: dust)

SILENT DATA CORRUPTIONS

Processor faults that do not cause a crash but introduce undesired data

- Usually go undetected immediately
- Data could be lost or modified

~3.61 CPUs out of 10000 exhibit SDCs

Different microarchitectures have different SDC failure rates

Arch	M1	M2	M3	M4	M5	M6	M7	M8	M9	avg
Failure rate	4.619‰	0.352‰	2.649‰	0.082‰	0.759‰	3.251‰	1.599‰	9.29‰	4.646‰	3.61‰

Table 2. Failure rate of different micro-architectures.

MERCURIAL CORES

Cores that produce silent corrupt execution errors

- Corrupt Execution Errors cause Silent Data Corruptions
- These errors are computational in nature
  - In a memory or a disk failure, you are either unable to read or you read what you had not written
  - Requires computing the result multiple times
  - Might only be triggered by certain specific instructions
  - Same instruction may only trigger intermittently

CHECKSUMS

- “Fingerprints” of data being stored or transmitted
- Used for detecting errors in data storage and transmission
  - Only detection (no correction)
- For some unit of data (block, file, packet, etc)
  - Generate a value (checksum) that represents the contents of the data unit
  - After storage or transmission, re-generate the value and compare against the original checksum

ERROR CORRECTING CODES

- Send more redundant information along with the actual data
- Allows you to detect more errors and fix some errors

ERROR CORRECTING CODES

- Send more redundant information along with the actual data
- Allows you to detect more errors and fix some errors
- Eg: (3,1) repetition code
  - Transmit each bit 3 times
  - Majority voting to decide the value of the bit
  - Fixes and detects any 1 bit-flip error

ERROR CORRECTING CODES

- Send more redundant information along with the actual data
- Allows you to detect more errors and fix some errors
- Eg: (3,1) repetition code
  - Transmit each bit 3 times
  - Majority voting to decide the value of the bit
  - Fixes and detects any 1 bit-flip error
- Eg: Hamming Code (7, 4)
  - 4 data bits are encoded with 3 parity bits

ERASURE CODES

- Bits could simply be omitted/erased
- Erasure Code extends original value with extra info
- Using extra info, original value can be reconstructed from any subset of a specific size

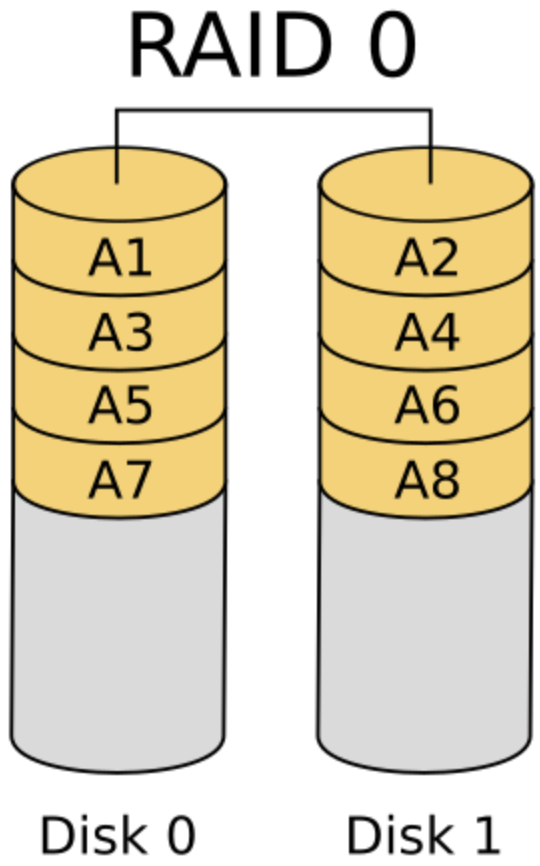
RAID 0 (STRIPING)

Stripe data across 2 disks

- Divide the data into blocks
- Distribute them equally

Removes single point of failure

Some loss of information



https://en.wikipedia.org/wiki/Standard_RAID_levels

RAID 1 (MIRRORING)

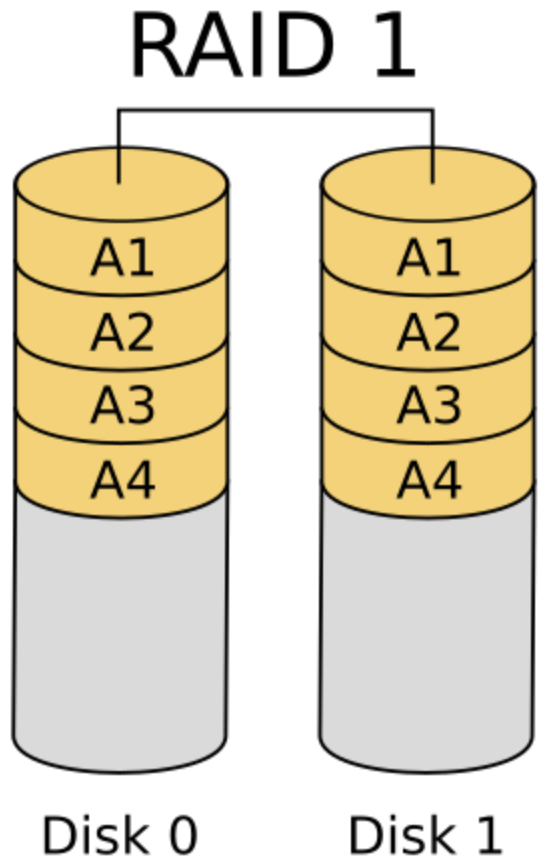
Mirror data across 2 disks

- Full copy of data

No single point of failure

Tolerates 1 disk failure

Uses 2x resources

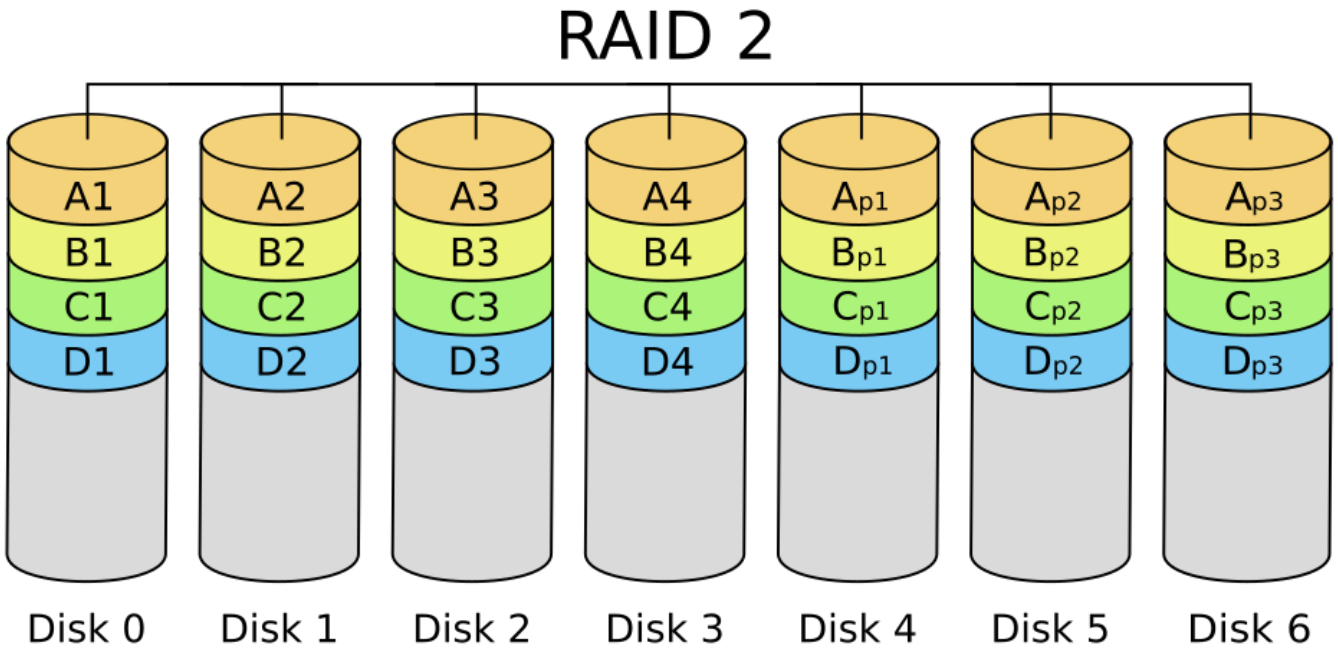


https://en.wikipedia.org/wiki/Standard_RAID_levels

RAID 2 (BIT STRIPING)

Stripe data across multiple disks

- Divide the data into bits
- Error Correction Code (Hamming Code) for each bit



Removes single point of failure

Better bit-level data integrity

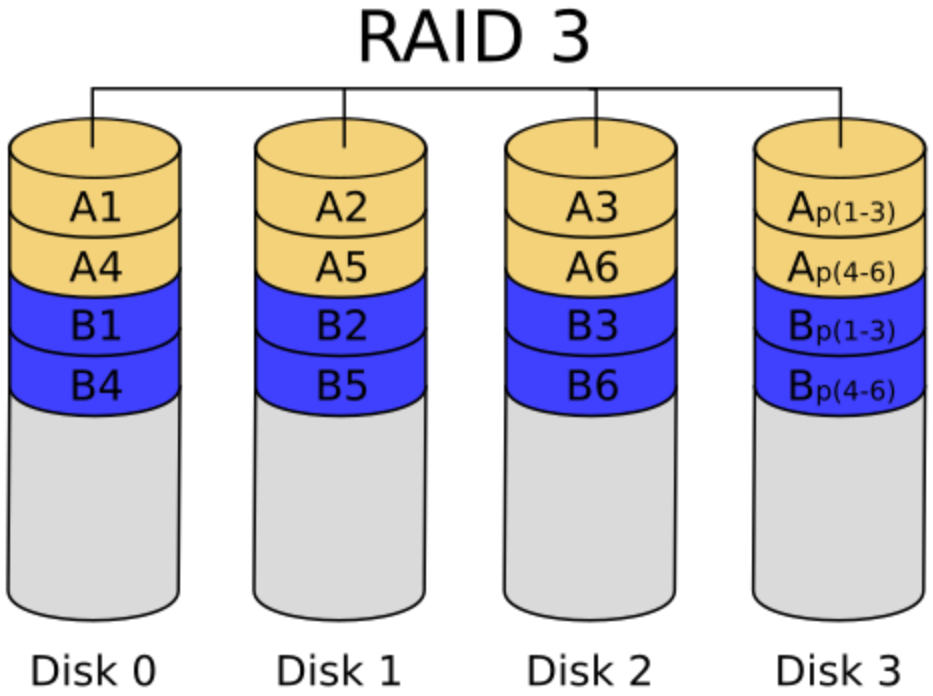
https://en.wikipedia.org/wiki/Standard_RAID_levels

RAID 3 (BYTE STRIPING + PARITY)

Stripe data across multiple disks

- Divide the data into bytes
- Distribute them equally

Parity bytes help with error detection



https://en.wikipedia.org/wiki/Standard_RAID_levels

RAID 4 (BLOCK STRIPING + PARITY)

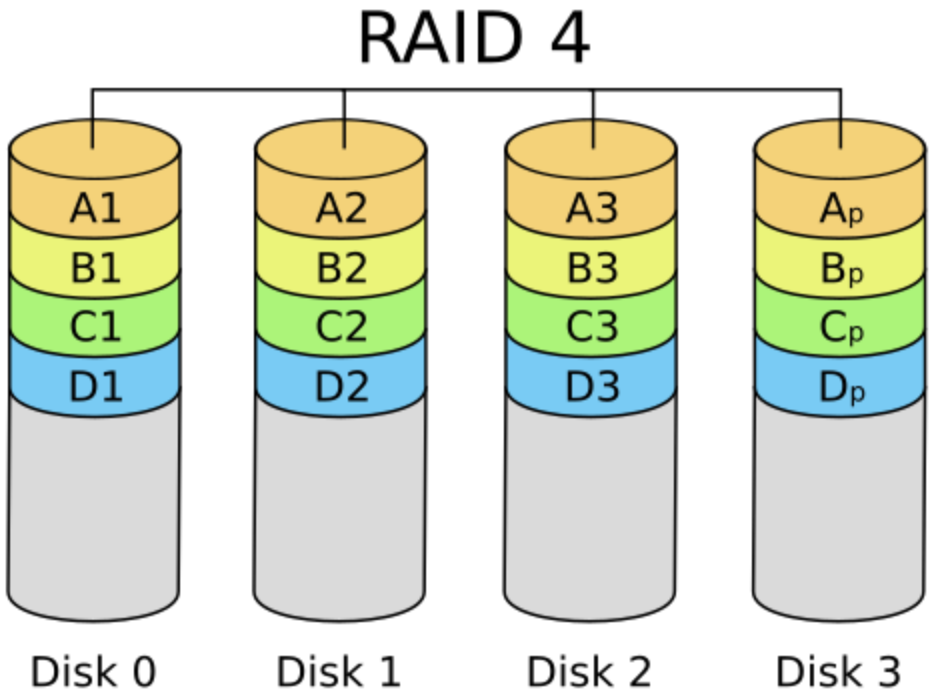
Stripe data across multiple disks

- Divide the data into blocks
- Distribute them equally

1 dedicated disk for parity blocks

Tolerates 1 disk failure

Data Redundancy is lost if parity disk fails



https://en.wikipedia.org/wiki/Standard_RAID_levels

RAID 5 (BLOCK STRIPING + DIST. PARITY)

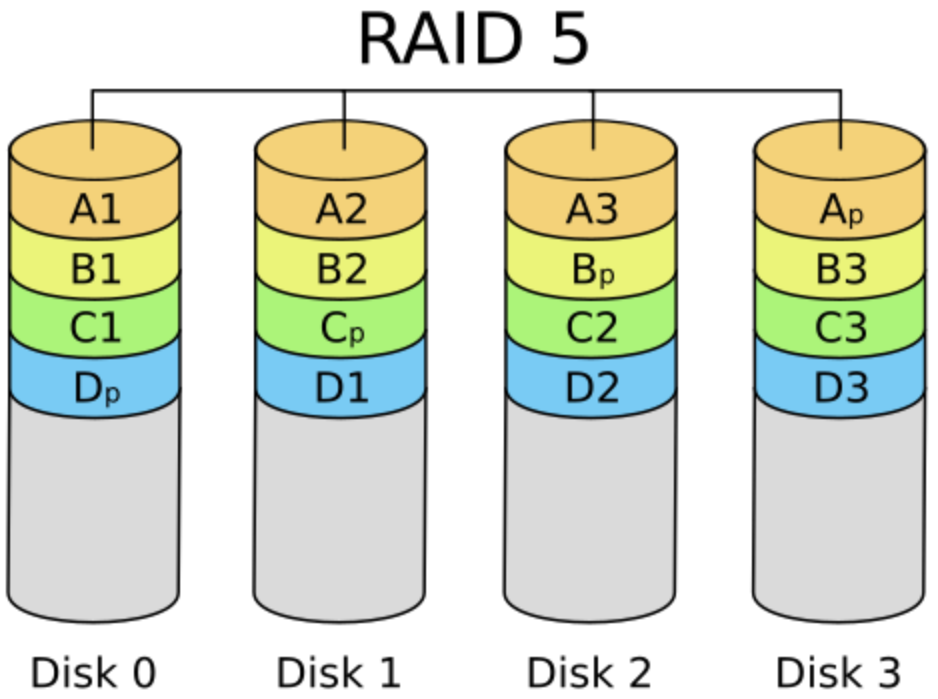
Stripe data across multiple disks

- Divide the data into blocks
- Distribute them equally

Parity blocks spread across disks

Tolerates 1 disk failure

Only partial data redundancy loss

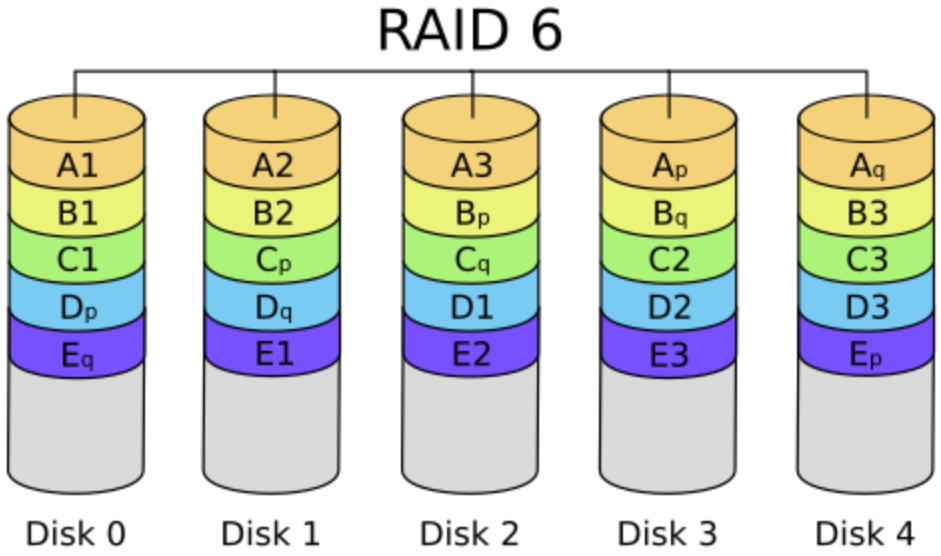


https://en.wikipedia.org/wiki/Standard_RAID_levels

RAID 6 (STRIPING + 2*DIST. PARITY)

Stripe data across multiple disks

- Divide the data into blocks
- Distribute them equally



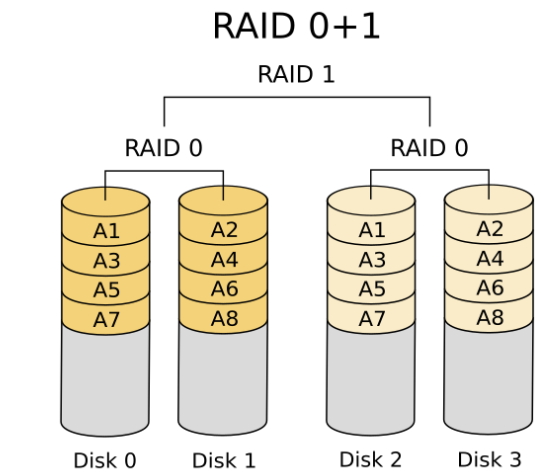
Multiple Parity blocks spread across disks

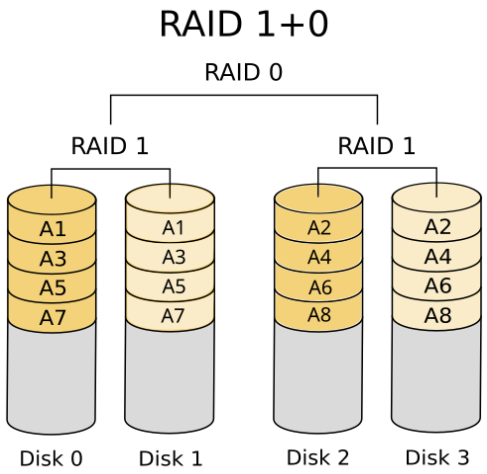
Tolerates 2 disk failures

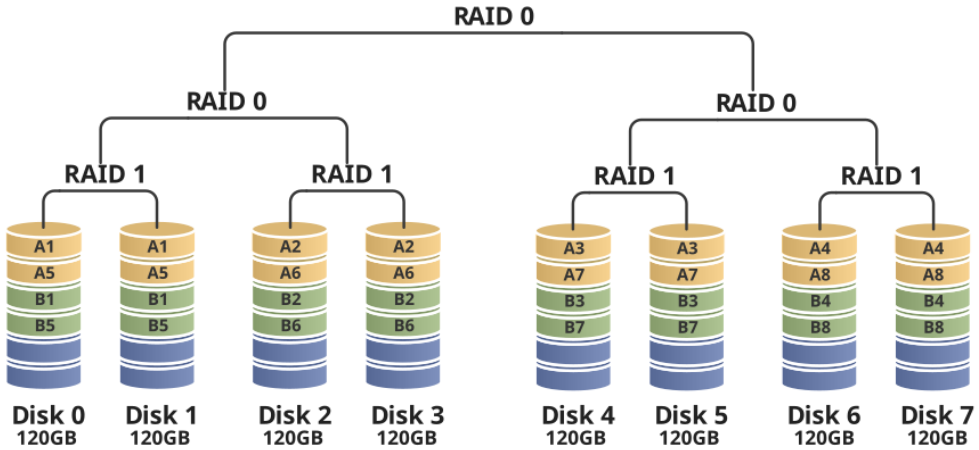
Only partial data redundancy loss

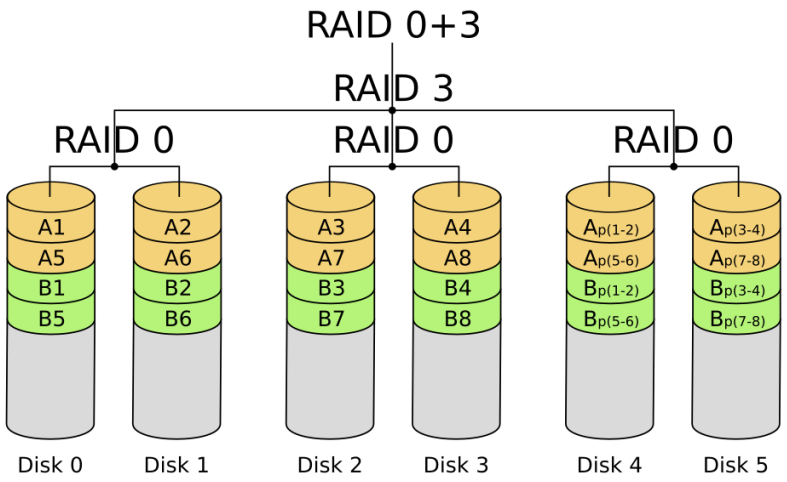
https://en.wikipedia.org/wiki/Standard_RAID_levels

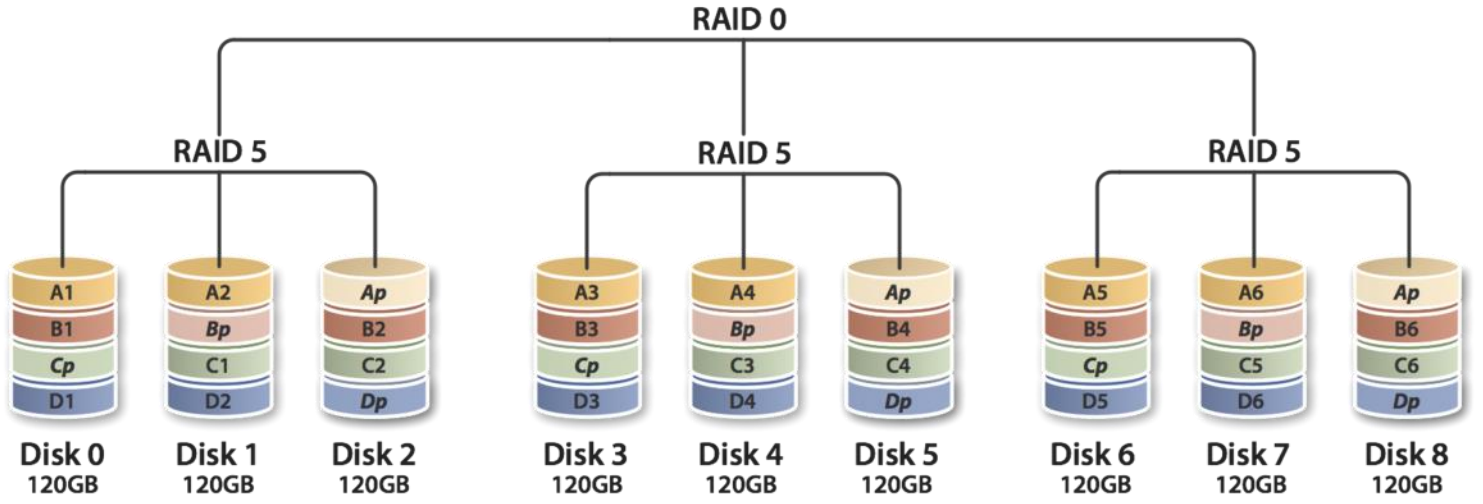
NESTED RAID











https://en.wikipedia.org/wiki/Nested_RAID_levels

THERMAL THROTTLING

- Overheating of CPUs can cause permanent damage to the chip

- Thermal throttling prevents the CPU from damage
  - Throttling kicks in when it measures the chip temperature to be higher than a specific threshold
  - Throttling kicks in to reduce the clock frequency
  - Slows down performance but protects the chip from permanent damage

DISCUSSION THEMES

- How to do resource allocation and utilization in the presence of hardware failures?
- When using RAID, when can you consider written data to be persistent?
- What are some ways you could potentially detect Silent Data Corruption failures during normal workload execution?