RAID

Redundant Array of Independent Disks

Niv Dayan

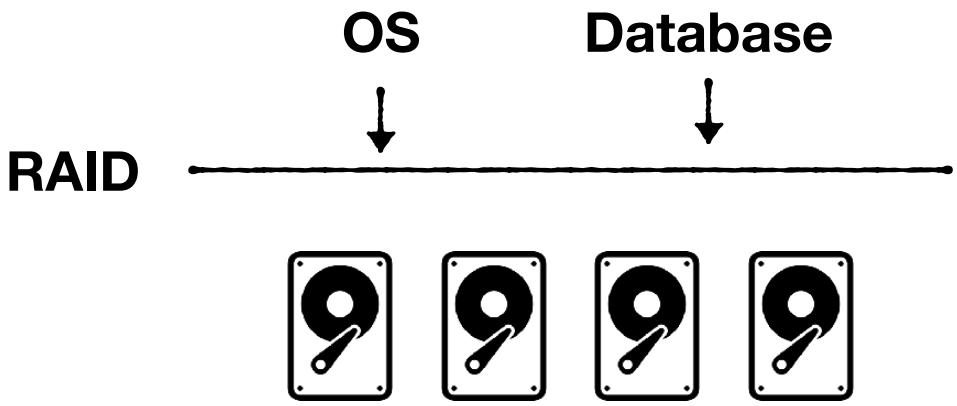
RAID Addresses Three Problems

Our database size exceeds one drive and we need more storage

A drive fails, and we need to recover its data

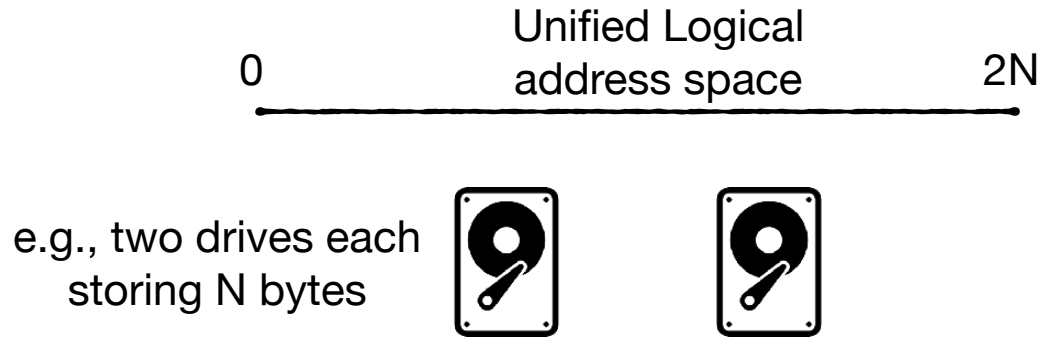
We want to overcome the limits of one storage device speed

RAID divides data along with redundancy on multiple disks



Enables larger data, better performance, and recovery of a drive failure

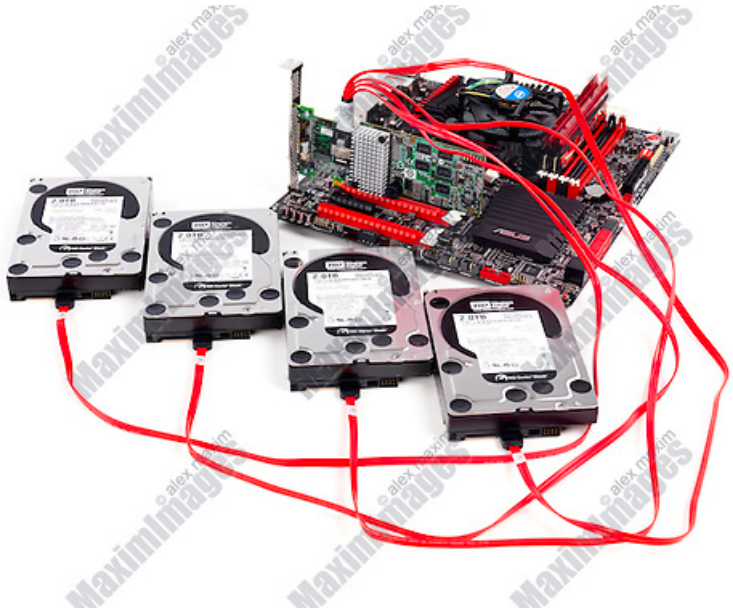
Expose a larger logical address space to OS



Looks to the OS like one drive, though consists of many

Can be implemented in…

Hardware **Or** **Software**

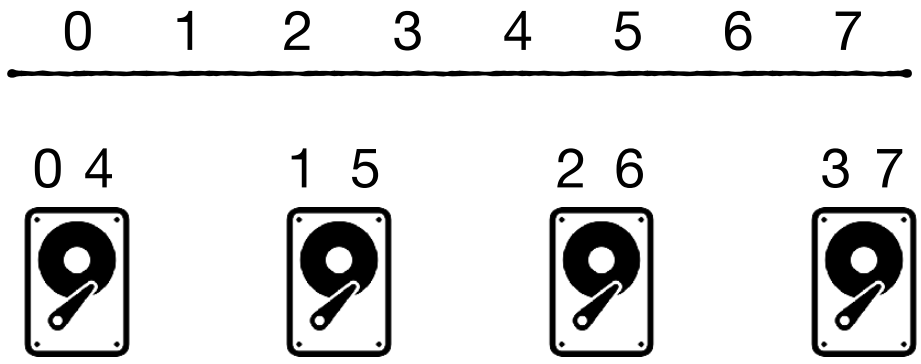


There are many RAID designs, but we'll only cover five

RAID 0 **RAID 1** **RAID 0+1** **RAID 4** **RAID 5**

RAID 0 - Pure striping

Stripe data in the logical address

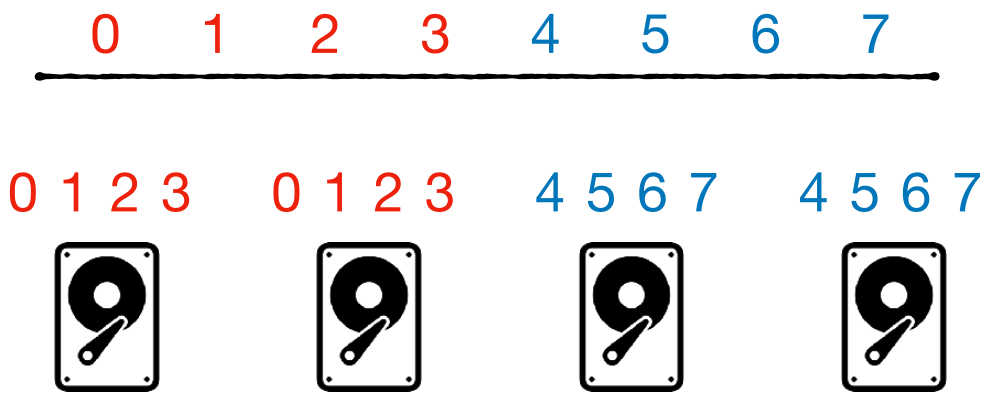


1. Much faster sequential writes and reads
2. Also improvement for random writes and reads due to load balancing
3. No redundancy. If one disk fails, we lose data.

RAID 0 **RAID 1** RAID 0+1 RAID 4 RAID 5

RAID 1 - Mirroring

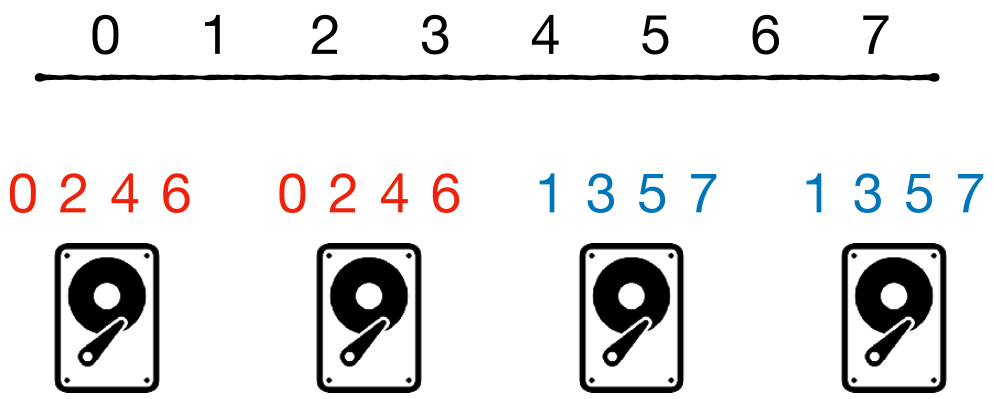
Each drive has one mirror



1. Slower writes as they must make 2 copies
2. Faster reads as we have a choice to read from a non-busy drive
3. Allows recovery of a disk but costs 50% of storage capacity

RAID 0 RAID 1 **RAID 0+1** RAID 4 RAID 5

RAID 0+1 - Striping and Mirroring



1. Faster sequential reads and writes as they are more distributed
2. Writes still require making two copies, and reads still have flexibility
3. Still requires 50% of storage capacity

RAID 0 RAID 1 RAID 0+1 **RAID 4** **RAID 5**

For next time