Last Time

• Pipelining
  – Can we perform computations more efficiently than CPI = 1.0?
  – What complications arise from trying to do this?

Today

• Swinging back around to performance
  – How fast will our computer access stored data?
  – How does HDD speed affect computation?
  – How can we improve HDD access times?
System Characteristics

• Dependability is important
• Performance measures:
  – Latency (response time)
  – Throughput (bandwidth)
• Desktops and embedded systems look for response time
• Servers look for throughput
  – Both “in general”
Dependability

- **Fault** – failure of a component
- **Reliability** – mean time to failure (MTTF)
- **Service Interruption** – mean time to repair (MTTR)
- Mean time between failures (MTBF)
  - \( MTBF = MTTF + MTTR \)
- **Availability** – \( \frac{MTTF}{MTTF + MTTR} \)
- Improving availability:
  - Reduce MTTR – improved tools and processes for diagnosis and repair
  - Increase MTTF – fault avoidance, fault tolerance, fault forecasting
Improving Availability

• Increase MTTF
  – **Fault avoidance** – Proactive measures taken to minimize occurrence of faults (backups, replication).
  – **Fault tolerance** – The ability of a system to continue operation in the event of a fault.
  – **Fault forecasting** – Predict likely faults so that they can be removed or their effects can be circumvented.
Let’s Talk about Hard Drives
Let’s Talk about Hard Drives

• Nonvolatile, rotating magnetic storage
• Each sector records:
  – Sector ID
  – Data (512 bytes, 4096 proposed)
  – Error Correcting Code (ECC) – a method of fault ______________
  – Synchronization fields and gaps
• Access to a sector involves:
  – Queueing delay to see if other accesses are pending
  – Seek latency – move the read/write head
  – Rotational latency – let the platter rotate to the head
  – Data transfer – read the data from the sector
  – Controller overhead
Disk Access Example

• 512 bytes/sector, 15000 RPM platter, 4ms average seek time, 100MB/s transfer rate, 0.2ms controller overhead, idle disk at start

• Average read time =
  – 4ms seek time
  – $\left(\frac{1}{2}\left/\frac{15000}{60}\right.\right) = 2$ms rotational latency
  – $512 / 100\text{MB/s} = 0.005$ms transfer time
  – 0.2 controller delay
  = 6.205ms
Disk Performance Issues

• Manufacturers quote average seek time, not average read time
  – Based on all possible seeks
  – Locality and OS scheduling lead to smaller actual average seek times

• Smart disk controller allocates physical sectors on the disk

• Disk drives include caches
  – Prefetch sectors in anticipation of access
  – Avoid seek and rotational delay
Flash Storage

- Nonvolatile semiconductor storage
  - 100x–1000x faster than disk
  - Smaller, lower power, more robust
  - More expensive
Flash Types

- **NOR flash** – bit cell like a NOR gate
  - Random read/write access
  - Used for instruction memory in embedded systems

- **NAND flash** – bit cell like a NAND gate
  - Denser (bits/area), but accessed in blocks
  - Cheaper per GB
  - Used for USB drives, media storage, etc.

- **Flash bits wear out after 1000s of accesses**
  - Not suitable replacement for RAM or HDD
  - **Wear leveling** – remap data to less-used blocks
RAID

- **RAID** – Redundant Array of Independent Disks
  - Uses multiple smaller disks
  - Parallelism improves performance
  - Plus, extra disk(s) for redundancy
- Provides fault tolerant storage system
  - Data stored on multiple disks
RAID Levels

• RAID 0
  – No redundancy (“AID”?), just stripe data over multiple disks (improves performance)
  – N disks

• RAID 1
  – Mirroring (simple replication of data)
  – Write data to both; read from primary, read from mirror if failure
  – N+N disks

• RAID 2 (not used in practice, too complex)
  – ECC (generate E-bit error correcting code)
  – N+E disks
RAID Levels

• RAID 3
  – Bit-interleaved parity (one extra disk stores a parity bit)
  – On read, read all disks
  – On write, generate new parity and update all disks
  – On failure, use parity to reconstruct missing data
  – N+1 disks

• RAID 4
  – Similar to RAID 3, but parity done by blocks instead of by bytes
RAID Levels

• RAID 5
  – Distributed parity
  – Like RAID 4, but now parity blocks are distributed across all disks
    • Avoids a write bottleneck on a parity drive
  – Widely used
  – N+1 disks
RAID Levels

• RAID 6
  – P+Q redundancy
  – Like RAID 5, but two disks of parity blocks
  – Better fault tolerance through more redundancy
  – N+2 disks

• In summary:
  – RAID Improves both performance and availability
    • High availability requires hot-swapping
  – Assumes independent disk failures (not useful if the building burns down)
Fallacy: Disk Dependability

• If a disk manufacturer quotes MTTF as 140 years (1,200,000 hours), all disks will work that long.

• **Wrong:** this is the mean time to failure
  – What if you have 1000 disks; how many will fail per year?
  – **AFR** – Annual Failure Rate

\[
AFR = \frac{1000 \text{ disks} \times 8760 \text{ hrs/disk}}{1200000 \text{ hrs/failure}} = 0.73\%
\]
Any Questions?