Motivation: Why Care About I/O?

- **CPU Performance**: almost 50% per year
- **I/O system performance** limited by *mechanical* delays (e.g., disk I/O)
  - < 10% per year (IO per sec)
- **Amdahl's Law**: system speed-up limited by the slowest part
- **I/O bottleneck**
  - Diminishing fraction of time in CPU
  - Diminishing value of faster CPUs
Disks:
Long-term, nonvolatile storage
Large, inexpensive, also serve as slow level in the storage hierarchy

Disk Terminology

♦ Several platters, with information recorded magnetically on both surfaces (usually)
♦ Bits recorded sequentially in tracks, divided into sectors (e.g., 512 Bytes)
♦ Actuator moves head (end of arm, 1/surface) over track ("seek"), select surface, wait for sector rotate under head, then read or write
  • "Cylinder": all tracks under heads
Disk Device Performance

- **Disk Latency** = Seek Time + Rotation Time + Transfer Time + Controller Overhead

- **Seek Time**: depends on number of tracks arm moves
- **Rotation Time**: depends on speed disk rotates, how far sector is from head's current position
- **Transfer Time**: depends on data rate (bandwidth) of disk (bit density), size of request

---

**Diagram**

- Platter
- Outer Track
- Inner Track
- Sector
- Head
- Spindle
- Arm
- Controller
- Actuator

---

**Rotation Time**: Avg. distance to sector from head
- 1/2 time of a rotation
- 7200 Revolutions Per Minute ⇒ 120 Rev/sec
- 1 revolution = 1 / 120 sec ⇒ 8.33 milliseconds
- 1/2 rotation (revolution) ⇒ 4.17 ms

**Seek Time**: Average number of tracks arm moves
- Sum all possible seek distances from all possible tracks / Total_#_ops
  - Assumes random seek distance
- Disk industry standard benchmark
- Typical: ~8 ms

**Transfer rate**
- 10-40 MByte/sec

**Capacity**: 100s Gbytes to TeraBytes
- Quadruples every 2 years
**Example:**

**Barracuda 180**

- 181.6 GB, 3.5 inch disk
- 12 platters, 24 surfaces
- 24,247 cylinders
- 7,200 RPM; (4.2 ms avg. latency)
- 7.4 ms avg. seek
- 65 MB/s
- 0.1 ms controller time

**Calculate time to read 64 KB (128 sectors) using specs**

- Disk latency = avg. seek time + avg. rotational delay + transfer time + controller overhead
- \( \text{Disk latency} = 7.4 \text{ ms} + 4.2 + 1.0 + 0.1 \text{ ms} = 12.7 \text{ ms} \)

**Source:** www.seagate.com

---

**Disk Performance Example Recalculated**

- Calculate again using 1/3 quoted seek time (not random, mostly to adjacent tracks), 3/4 of internal bandwidth (check bits and gaps between sectors)
- Disk latency = average seek time + average rotational delay + transfer time + controller overhead

\[
\begin{align*}
\text{Disk latency} & = (0.33 \times 7.4 \text{ ms}) + 0.5 \times \frac{1}{7200 \text{ RPM}} + 64 \text{ KB} / (0.75 \times 65 \text{ MB/s}) + 0.1 \text{ ms} \\
& = 2.5 \text{ ms} + 0.5 / (7200 \text{ RPM} / (60000 \text{ ms/M})) + 64 \text{ KB} / (47 \text{ KB/ms}) + 0.1 \text{ ms} \\
& = 2.5 + 4.2 + 1.4 + 0.1 \text{ ms} = 8.2 \text{ ms} (64\% \text{ of } 12.7) \\
\end{align*}
\]
Large Arrays of Disks

Servers and data-centers require large storage capacity but large arrays of disks have low reliability

- Reliability\_disk = \exp(-\lambda t) where \lambda is a constant failure rate
- MTTF = \text{Mean\_Time\_to\_Failure} = 1 / \lambda
- A single disk has MTTF=50,000 hours = 6 years
- Reliability of N disks = [\exp(-\lambda t)]^N = \exp(-N\lambda t)
- MTTF\_array = 1 / N \lambda
- For N=70 disks: 50,000/70= 700 hours = 1 month
- Arrays (without redundancy) too unreliable

RAID

- Redundant Arrays of Independent Disks
- Files are "striped" across multiple disks with redundant data added
- Upon failure: Contents reconstructed from data redundantly stored in the array
- Redundancy yields high data availability
  - service still provided to user, even if some components fail
  - but
  \Rightarrow Capacity penalty to store redundant info
  \Rightarrow Performance penalty to update redundant info
RAID 1: Disk Mirroring/Shadowing

- Each disk is fully duplicated onto its “mirror”
- Very high availability
- Bandwidth sacrifice on write:
  - Logical write = two physical writes
- Reads may be optimized
- Most expensive solution: 100% capacity overhead
- (RAID 2 not interesting, so skip)

RAID 3: Parity Disk

- \( P = \text{sum mod 2 of other disks (parity)} \)
- If disk fails, subtract \( P \) from sum of other disks to find missing information
- Capacity overhead
- Wider arrays reduce capacity overhead, but decrease reliability
Inspiration for RAID 4

- RAID 3 relies on parity disk to detect and correct errors on Read
  - Must read from all disks every time
- But every sector has an error check field
- Rely on error check field to catch errors on read, not on the parity disk
- Use parity disk only for complete disk failure

RAID 4: High I/O Rate

Example:
- Small read D0 & D5
- Large write D12-D15
RAID 5: High I/O Rate Interleaved Parity

Independent writes possible because of interleaved parity

Example: write to D0 & D5 uses disks 0, 1, 3 and 4

System Availability: Orthogonal RAIDs

RAID Group: data redundancy

Common Support Components: fans, power supplies, controller, cables
Summary: RAID Techniques

- **Disk Mirroring, Shadowing (RAID 1)**
  - Each disk is fully duplicated
  - Logical write = two physical writes
  - 100% capacity overhead

- **High I/O Rate Parity Array (RAID 5)**
  - Interleaved parity blocks
  - Independent reads and writes
  - Logical write = 2 reads + 2 writes

Solid state drives - flash technology

- Organization similar to DRAM but basic cell is different
Solid state drives

- Most are currently based on NAND flash
  - Read blocks (4K byte)
  - Write - erasure (multiple blocks) - slower than Read
- SLC - single layer cell; MLC - multi-level cell
  - SLC - low density, read ~25μs write ~250μs; write endurance ~ 100,000 cycles
  - TLC (Triple level) - high density, read ~75μs; write ~1000μs; write endurance ~ 1,000 cycles
- Wear leveling - re-mapping of the contents on every write - avoid the high frequency of re-writes to certain blocks
- Random bit errors due to the noisy environment in the chip and the weak signals from the cells
- Error correction (for transient and permanent bit failures)
  - BCH code (a cyclic code based on finite-fields), e.g., can correct up to 55 bit errors in a 512 byte sector

Solid-state vs. hard disk drives

<table>
<thead>
<tr>
<th></th>
<th>Solid-state Drive</th>
<th>Hard Disk Drive</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start-up time</td>
<td>Instantaneous</td>
<td>Spin-up</td>
</tr>
<tr>
<td>Access latency</td>
<td>0.1ms or less,</td>
<td>3-12 msec, varies with</td>
</tr>
<tr>
<td></td>
<td>random access</td>
<td>relative position</td>
</tr>
<tr>
<td>Transfer rate</td>
<td>100-600 MB/sec</td>
<td>About 150 MB/sec</td>
</tr>
<tr>
<td>Cost</td>
<td>$0.37/GB</td>
<td>$0.05-0.10/GB</td>
</tr>
<tr>
<td>Power consumption</td>
<td>About 50-100%</td>
<td>0.35 - 20Watt</td>
</tr>
<tr>
<td>Capacity</td>
<td>512 GB and up</td>
<td>8 TB</td>
</tr>
<tr>
<td>Reliability</td>
<td>Power outage may</td>
<td>Mechanical failures,</td>
</tr>
<tr>
<td></td>
<td>cause damage</td>
<td>unpowered: long lifetime</td>
</tr>
<tr>
<td>Environment impact</td>
<td>No sensitive and less</td>
<td>Sensitive to shocks and</td>
</tr>
<tr>
<td></td>
<td>noisy</td>
<td>temperature</td>
</tr>
</tbody>
</table>