Japan's growing elderly population will be able to buy companionship in the form of a robot, programmed to provide just enough small talk to keep them from going senile. Snuggling Ifbot, dressed in an astronaut suit with a glowing face, has the conversation ability of a five-year-old, the language level needed to stimulate the brains of seniors.

Magnetic Disks

- **Purpose:**
  - Long-term, nonvolatile, inexpensive storage for files
  - Large, inexpensive, slow level in the memory hierarchy (discuss later)

**Photo of Disk Head, Arm, Actuator**

- Actuator
- Arm
- Head
- Platters (12)

**Disk Device Terminology**

- Several *platters*, with information recorded magnetically on both *surfaces* (usually)
- Bits recorded in *tracks*, which in turn divided into *sectors* (e.g., 512 Bytes)
- *Actuator* moves *head* (end of *arm*) over track ("seek"), wait for *sector* rotate under *head*, then read or write

**Disk Device Performance**

- **Disk Latency = Seek Time + Rotation Time + Transfer Time + Controller Overhead**
  - Seek Time? depends on no. tracks move arm, seek speed of disk
  - Rotation Time? depends on speed disk rotates, how far sector is from head
  - Transfer Time? depends on data rate (bandwidth) of disk (bit density), size of request

**Data Rate: Inner vs. Outer Tracks**

- To keep things simple, originally same # of sectors/track
  - Since outer track longer, lower bits per inch
- Competition decided to keep bits/inch (BPI) high for all tracks ("constant bit density")
  - More capacity per disk
  - More sectors per track towards edge
  - Since disk spins at constant speed, outer tracks have faster data rate
- Bandwidth outer track 1.7X inner track!
**Disk Performance Model /Trends**

- **Capacity**: +100% / year (2X / 1.0 yrs)
  - Over time, grown so fast that number of platters has reduced (some even use only 1 now!)
- **Transfer rate (BW)**: +40%/yr (2X / 2 yrs)
- **Rotation+Seek time**: −8%/yr (1/2 in 10 yrs)
- **Areal Density**
  - Bits recorded along a track: Bits/Inch (BPI)
  - # of tracks per surface: Tracks/Inch (TPI)
  - We care about bit density per unit area: Bits/inch²
  - Called Areal Density = BPI x TPI
- **MB/S**: > 100%/year (2X / 1.0 yrs)
  - Fewer chips + areal density

---

**Disk History (IBM)**

- **Model 3340 hard disk 1973**
  - Capacity: 1.7 Mibytes
  - Data density: 1.7 Mibit/sq. in.

- **Model 2370 1979**
  - Capacity: 7.7 Mibytes
  - Data density: 7.7 Mibit/sq. in.

**Historical Perspective**

- **Form factor and capacity drives market, more than performance**
  - **1970s**: Mainframes ⇒ 14" diam. disks
  - **1980s**: Minicomputers, Servers ⇒ 8", 5.25" diam. disks
  - **Late 1980s/Early 1990s**:
    - Pizzabox PCs ⇒ 3.5 inch diameter disks
    - Laptops, notebooks ⇒ 2.5 inch disks
    - Palmtops didn’t use disks, so 1.8 inch diameter disks didn’t make it

---

**State of the Art: Barracuda 7200.7 (2004)**

- 200 GB, 3.5-inch disk
- 7200 RPM; Serial ATA
- 2 platters, 4 surfaces
- 8 watts (idle)
- 8.5 ms avg. seek
- 32 to 58 MB/s Xfer rate
- $125 = $0.625 / GB

**1 inch disk drive!**

- **2004 Hitachi Microdrive**:
  - 1.7” x 1.4” x 0.2”
  - 4 GB, 3600 RPM, 4-7 MB/s, 12 ms seek
  - Digital cameras, PalmPC
- **2006 MicroDrive?**
  - 16 GB, 10 MB/s!
  - Assuming past trends continue

**source:** www.seagate.com
Use Arrays of Small Disks...

Katz and Patterson asked in 1987:
Can smaller disks be used to close gap in performance between disks and CPUs?

Conventional:
4 disk designs
3.5" 5.25" 10" 14" Low End 

Disk Array:
1 disk design
3.5" High End

Replace Small Number of Large Disks with Large Number of Small Disks! (1988 Disks)

<table>
<thead>
<tr>
<th>IBM 3390K</th>
<th>IBM 3.5&quot; 0061</th>
<th>x70</th>
</tr>
</thead>
<tbody>
<tr>
<td>Capacity</td>
<td>20 GBytes</td>
<td>320 MBytes</td>
</tr>
<tr>
<td>Volume</td>
<td>97 cu. ft.</td>
<td>0.1 cu. ft.</td>
</tr>
<tr>
<td>Power</td>
<td>3 KW</td>
<td>11 W</td>
</tr>
<tr>
<td>Data Rate</td>
<td>15 MB/s</td>
<td>1.5 MB/s</td>
</tr>
<tr>
<td>I/O Rate</td>
<td>600 I/Os/s</td>
<td>55 I/Os/s</td>
</tr>
<tr>
<td>MTTF</td>
<td>250 Khrs</td>
<td>50 Khrs</td>
</tr>
<tr>
<td>Cost</td>
<td>$250K</td>
<td>$2K</td>
</tr>
</tbody>
</table>

Disk Arrays potentially high performance, high MB per cu. ft., high MB per KW,
but what about reliability?

Array Reliability

- Reliability - whether or not a component has failed
  - measured as Mean Time To Failure (MTTF)
- Reliability of N disks = Reliability of 1 Disk ÷ N (assuming failures independent)
  - 50,000 Hours ÷ 70 disks = 700 hour
- Disk system MTTF: Drops from 6 years to 1 month!
- Disk arrays too unreliable to be useful!

Redundant Arrays of (Inexpensive) Disks

- Files are “striped” across multiple disks
- Redundancy yields high data availability
  - Availability: service still provided to user, even if some components failed
- Disks will still fail
- Contents reconstructed from data redundantly stored in the array
  ⇒ Capacity penalty to store redundant info
  ⇒ Bandwidth penalty to update redundant info

Berkeley History, RAID-I

- RAID-I (1989)
  - Consisted of a Sun 4/280 workstation with 128 MB of DRAM, four dual-string SCSI controllers, 28 5.25-inch SCSI disks and specialized disk striping software
  - Today RAID is > $27 billion dollar industry, 80% nonPC disks sold in RAIDs

“RAID 0”: No redundancy = “AID”

- Assume have 4 disks of data for this example, organized in blocks
- Large accesses faster since transfer from several disks at once

This and next 5 slides from RAID.edu. http://www.acm.org/04_01_00.html
RAID 1: Mirror data

- Each disk is fully duplicated onto its "mirror"
  - High availability can be achieved
- Bandwidth reduced on write:
  - 1 Logical write = 2 physical writes
- Most expensive solution: 100% capacity overhead

RAID 3: Parity

- Parity computed across group to protect against hard disk failures, stored in P disk
  - Logically, a single high capacity, high transfer rate disk
  - 25% capacity cost for parity in this example vs. 100% for RAID 1 (5 disks vs. 8 disks)

Inspiration for RAID 5

- Small writes (write to one disk):
  - Option 1: read other data disks, create new sum and write to Parity Disk (access all disks)
  - Option 2: since P has old sum, compare old data to new data, add the difference to P:
    1 logical write = 2 physical reads + 2 physical writes to 2 disks

"And in conclusion..."

- Magnetic Disks continue rapid advance: 60%/yr capacity, 40%/yr bandwidth, slow on seek, rotation improvements, MB/S improving 100%/yr?
  - Designs to fit high volume form factor
- RAID
  - Higher performance with more disk arms per $ per disk
  - Adds option for small # of extra disks
  - Today RAID is > $27 billion dollar industry, 80% nonPC disks sold in RAIDs; started at Cal