Errors and Failures
Feb 3, 2016
Types of Errors

- **Hard errors**: The component is dead.

- **Soft errors**: A signal or bit is wrong, but it doesn’t mean the component must be faulty.

- **Note**: You can have recurring soft errors due to faulty, but not dead, hardware.
Examples

• DRAM errors

  • Hard errors: Often caused by motherboard - faulty traces, bad solder, etc.

  • Soft errors: Often caused by cosmic radiation or alpha particles (from the chip material itself) hitting memory cell, changing value. (Remember that DRAM is just little capacitors to store charge... if you hit it with radiation, you can add charge to it.)
Some fun #s

- Both Microsoft and Google have recently started to identify DRAM errors as an increasing contributor to failures... Google in their datacenters, Microsoft on your desktops.

- We’ve known hard drives fail for years, of course. :)

## Replacement Rates

<table>
<thead>
<tr>
<th>Component</th>
<th>HPC1 %</th>
<th>COM1 Component</th>
<th>COM1 %</th>
<th>COM2 Component</th>
<th>COM2 %</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hard drive</td>
<td>30.6</td>
<td>Power supply</td>
<td>34.8</td>
<td>Hard drive</td>
<td>49.1</td>
</tr>
<tr>
<td>Memory</td>
<td>28.5</td>
<td>Memory</td>
<td>20.1</td>
<td>Motherboard</td>
<td>23.4</td>
</tr>
<tr>
<td>Misc/Unk</td>
<td>14.4</td>
<td>Hard drive</td>
<td>18.1</td>
<td>Power supply</td>
<td>10.1</td>
</tr>
<tr>
<td>CPU</td>
<td>12.4</td>
<td>Case</td>
<td>11.4</td>
<td>RAID card</td>
<td>4.1</td>
</tr>
<tr>
<td>motherboard</td>
<td>4.9</td>
<td>Fan</td>
<td>8</td>
<td>Memory</td>
<td>3.4</td>
</tr>
<tr>
<td>Controller</td>
<td>2.9</td>
<td>CPU</td>
<td>2</td>
<td>SCSI cable</td>
<td>2.2</td>
</tr>
<tr>
<td>QSW</td>
<td>1.7</td>
<td>SCSI Board</td>
<td>0.6</td>
<td>Fan</td>
<td>2.2</td>
</tr>
<tr>
<td>Power supply</td>
<td>1.6</td>
<td>NIC Card</td>
<td>1.2</td>
<td>CPU</td>
<td>2.2</td>
</tr>
<tr>
<td>MLB</td>
<td>1</td>
<td>LV Pwr Board</td>
<td>0.6</td>
<td>CD-ROM</td>
<td>0.6</td>
</tr>
<tr>
<td>SCSI BP</td>
<td>0.3</td>
<td>CPU heatsink</td>
<td>0.6</td>
<td>Raid Controller</td>
<td>0.6</td>
</tr>
</tbody>
</table>
Measuring Availability

- Mean time to failure (MTTF)
- Mean time to repair (MTTR)
- MTBF = MTTF + MTTR  (mean time between failure)

- Availability = MTTF / (MTTF + MTTR)
  - Suppose OS crashes once per month, takes 10 min to reboot.
  - MTTF = 720 hours = 43,200 minutes
    MTTR = 10 minutes
  - Availability = 43200 / 43210 = 0.997 (~“3 nines”)
## Availability

<table>
<thead>
<tr>
<th>Availability %</th>
<th>Downtime per year</th>
<th>Downtime per month*</th>
<th>Downtime per week</th>
</tr>
</thead>
<tbody>
<tr>
<td>90% (&quot;one nine&quot;)</td>
<td>36.5 days</td>
<td>72 hours</td>
<td>16.8 hours</td>
</tr>
<tr>
<td>95%</td>
<td>18.25 days</td>
<td>36 hours</td>
<td>8.4 hours</td>
</tr>
<tr>
<td>97%</td>
<td>10.96 days</td>
<td>21.6 hours</td>
<td>5.04 hours</td>
</tr>
<tr>
<td>98%</td>
<td>7.30 days</td>
<td>14.4 hours</td>
<td>3.36 hours</td>
</tr>
<tr>
<td>99% (&quot;two nines&quot;)</td>
<td>3.65 days</td>
<td>7.20 hours</td>
<td>1.68 hours</td>
</tr>
<tr>
<td>99.50%</td>
<td>1.83 days</td>
<td>3.60 hours</td>
<td>50.4 minutes</td>
</tr>
<tr>
<td>99.80%</td>
<td>17.52 hours</td>
<td>86.23 minutes</td>
<td>20.16 minutes</td>
</tr>
<tr>
<td>99.9% (&quot;three nines&quot;)</td>
<td>8.76 hours</td>
<td>43.8 minutes</td>
<td>10.1 minutes</td>
</tr>
<tr>
<td>99.95%</td>
<td>4.38 hours</td>
<td>21.56 minutes</td>
<td>5.04 minutes</td>
</tr>
<tr>
<td>99.99% (&quot;four nines&quot;)</td>
<td>52.56 minutes</td>
<td>4.32 minutes</td>
<td>1.01 minutes</td>
</tr>
<tr>
<td>99.999% (&quot;five nines&quot;)</td>
<td>5.26 minutes</td>
<td>25.9 seconds</td>
<td>6.05 seconds</td>
</tr>
<tr>
<td>99.9999% (&quot;six nines&quot;)</td>
<td>31.5 seconds</td>
<td>2.59 seconds</td>
<td>0.605 seconds</td>
</tr>
<tr>
<td>99.99999% (&quot;seven nines&quot;)</td>
<td>3.15 seconds</td>
<td>0.259 seconds</td>
<td>0.0605 seconds</td>
</tr>
</tbody>
</table>

For a reliable component, may have to wait a long time to determine its availability/downtime!
Availability in practice

• Carrier airlines (2002 FAA fact book)
  • 41 accidents, 6.7M departures
  • 99.9993% availability

• 911 Phone service (1993 NRIC report)
  • 29 minutes per line per year
  • 99.994%

• Standard phone service (various sources)
  • 53+ minutes per line per year
  • 99.99+% 

• End-to-end Internet Availability
  • 95% - 99.6%
Real Devices

PRODUCT OVERVIEW

Cheetah 15K.4
Mainstream enterprise disc drive

Simply the best price/performance, lowest cost of ownership disc drive ever

KEY FEATURES AND BENEFITS
- The Cheetah™ 15K.4 is the highest-performance drive ever offered by Seagate®, delivering maximum IOPS with lower drive-to-yield house TDs.
- The Cheetah 15K.4 gives conformance with the breakthrough benefits of some attuned SAS/SATA models for the optimal 2.5-inch drive for the solid enterprise storage.
- Reaction self-diagnosed background management functions improve media integrity, enhance drive efficiency, reduce incidence of integration failures and improve fault resiliency.
- The Cheetah 15K.4 inherits its mechanics architecture and features from the Cheetah 15K.7 to ensure greater factory consistency and reduced time to market.

KEY SPECIFICATIONS
- 144, 72, and 36 drive capacities
- 3.0 max average rest and 3.6 max average write seek times
- Up to 60 Mbyte/s sustained transfer rates
- 1.4 million hours full duty cycle MTBF
- Serial Attached SCSI (SAS), Ultra320 SCSI and 2 Gbit/sec Fibre Channel interfaces
- 3 year warranty.

For more information on why 15K is the industry’s best price/performance disc drive for use in mainstream storage applications, visit http://seagate.com/15K
Real Devices – the small print

- The Cheetah 15K.4 is the highest-performance drive ever offered by Seagate®, delivering maximum IOPS with fewer drives to yield lower TCO.
- The Cheetah 15K.4 price-per-performance value united with the breakthrough benefits of serial attached SCSI (SAS) make it the optimal 3.5-inch drive for rock solid enterprise storage.
- Proactive, self-initiated background management functions improve media integrity, increase drive efficiency, reduce incidence of integration failures and improve field reliability.
- The Cheetah 15K.4 shares its electronics architecture and firmware base with Cheetah 10K.7 and Savvio™ to ensure greater factory consistency and reduced time to market.

**KEY SPECIFICATIONS**

- 146-, 73- and 36-Gbyte capacities
- 3.3-msec average read and 3.8-msec average write seek times
- Up to 96 Mbytes/sec sustained transfer rate
- **1.4 million hours full duty cycle MTBF**
- Serial Attached SCSI (SAS), Ultra320 SCSI and 2 Gbits/sec Fibre Channel interfaces
- 5-year warranty

For more information on why 15K is the industry’s best price/performance disc drive for use in mainstream storage applications, visit [http://specials.seagate.com/15k](http://specials.seagate.com/15k)
Disk failure conditional probability distribution - Bathtub curve

Infant mortality

Burn out

Stable failure period

1 / (reported MTTF)

Expected operating lifetime

Rate of failure

Time

0
Other Bathtub Curves

Human Mortality Rates (US, 1999)

So, back to disks...

• How can disks fail?
  • Whole disk failure (power supply, electronics, motor, etc.)
  • Sector errors - soft or hard
    • Read or write to the wrong place (e.g., disk is bumped during operation)
    • Can fail to read or write if head is too high, coating on disk bad, etc.
  • Disk head can hit the disk and scratch it.
Coping with failures...

- A failure
  - Let’s say one bit in your DRAM fails.

- Propagates
  - Assume it flips a bit in a memory address the kernel is writing to. That causes a big memory error elsewhere, or a kernel panic.
  - Your program is running one of a dozen storage servers for your distributed filesystem.
  - A client can’t read from the DFS, so it hangs.
  - A professor can’t check out a copy of your assignment, so he gives you an F :-(
Recovery Techniques

- We’ve already seen some: e.g., retransmissions in TCP and in your RPC system
- Modularity can help in failure isolation: preventing an error in one component from spreading.
  - Analogy: The firewall in your car keeps an engine fire from affecting passengers
- Redundancy and Retries
  - Later lectures: Specific techniques used in file systems, disks
  - This time: Understand how to quantify reliability
  - Understand basic techniques of replication and fault masking
What are our options?

1. Silently return the wrong answer.

2. Detect failure.

3. Correct / mask the failure