61C In the News

HOLIDAY MONEY

Malls track shoppers' cell phones on Black Friday

By Annalyn Censky @CNNMoneyTech November 22, 2011: 11:48

Starting on Black Friday and running through New Year's Day, two U.S. malls -- Promenade Temecula in southern California and Short Pump Town Center in Richmond, Va. -- will track guests' movements by monitoring the signals from their cell phones.

The goal is for stores to answer questions like: How many Nordstrom shoppers also stop at Starbucks? How long do most customers linger in Victoria's Secret? Are there unpopular spots in the mall that aren't being visited?

While the data that's collected is anonymous, it can follow shoppers' paths from store to store.

Consumers can opt out by turning off their phones.

CS 61C: Great Ideas in Computer Architecture (Machine Structures)

Lecture 38: IO, Networking & Disks

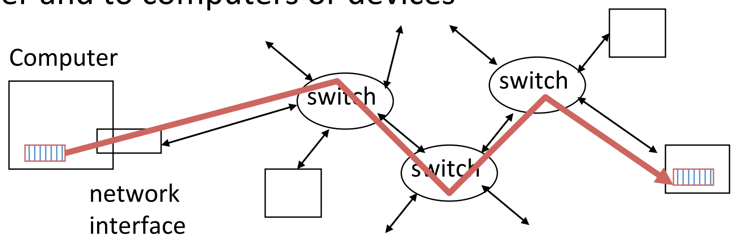
Instructors:
Mike Franklin
Dan Garcia
http://inst.eecs.Berkeley.edu/~cs61c/Fa11

Review

- Exceptions are “Unexpected” events
- Interrupts are asynchronous
  - can be used for interacting with I/O devices
- Need to handle in presence of pipelining, etc.
  - Logic similar to that of Branch mis-prediction
- Networks are another form of I/O
- Internet – 1962
  - Started with 4 hosts, growing exponentially since
- WWW – 1986
  - “Vague but Exciting” proposal at CERN
- Shared vs. Switched networks

What makes networks work?

- links connecting switches and/or routers to each other and to computers or devices



- **ability to name the components and to route packets of information - messages - from a source to a destination**
- **Layering, redundancy, protocols, and encapsulation as means of <u>abstraction</u> (61C big idea)**

CISCO SYSTEMS

Software Protocol to Send and Receive

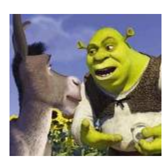
- SW Send steps
  1: Application copies data to OS buffer
  2: OS calculates checksum, starts timer
  3: OS sends data to network interface HW and says start
- SW Receive steps
  3: OS copies data from network interface HW to OS buffer
  2: OS calculates checksum, if OK, send ACK; if not, delete message (sender resends when timer expires)
  1: If OK, OS copies data to user address space, & signals application to continue

Dest Net ID	Src Net ID	Len	ACK INFO	CMD/ Address /Data	Checksum
Header				Payload	Trailer

11/28/11 Fall 2011 -- Lecture #38 5

Protocol for Networks of Networks?

- **Abstraction to cope with complexity of communication**
- **Networks are like onions**
  - **Hierarchy of layers:**
    - **Application (chat client, game, etc.)**
    - **Transport (TCP, UDP)**
    - **Network (IP)**
    - **Physical Link (wired, wireless, etc.)**



Networks are like onions.
They stink?
Yes. No!
Oh, they make you cry.
No!… Layers. Onions have layers. Networks have layers.

11/28/11 Fall 2011 -- Lecture #38 6

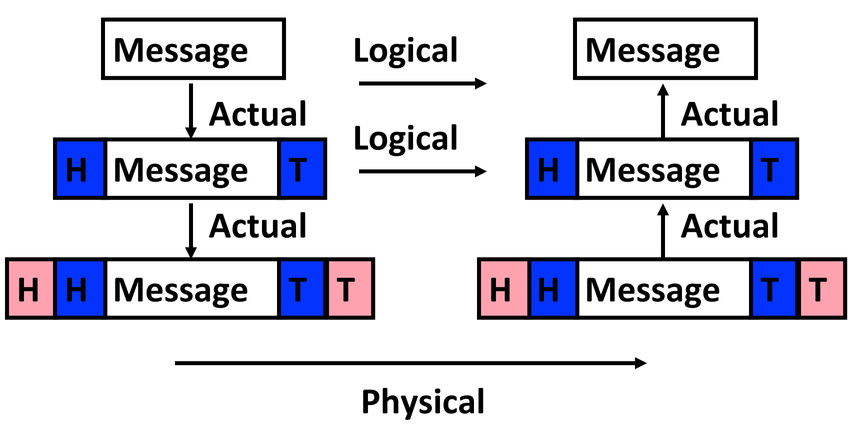
Protocol Family Concept

- Key to protocol families is that communication occurs logically at the same level of the protocol, called peer-to-peer…

  …but is implemented via services at the next lower level

- Encapsulation: carry higher level information within lower level "envelope"
- Fragmentation: break packet into multiple smaller packets and reassemble

Protocol Family Concept



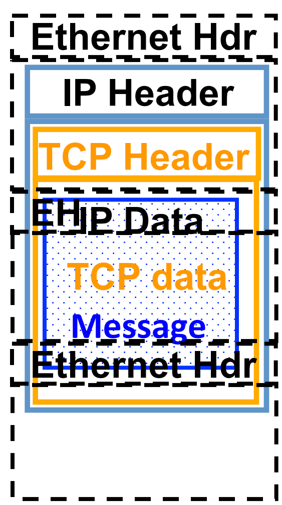
Protocol for Network of Networks

- Transmission Control Protocol/Internet Protocol (TCP/IP)
  (TCP :: a Transport Layer)
  - This protocol family is the basis of the Internet, a WAN protocol
  - IP makes best effort to deliver
    - Packets can be lost, corrupted
  - TCP guarantees delivery
  - TCP/IP so popular it is used even when communicating locally: even across homogeneous LAN

11/28/11 Fall 2011 -- Lecture #38 9

TCP/IP packet, Ethernet packet, protocols

- Application sends message
- **TCP breaks into 64KiB segments, adds 20B header**
- **IP adds 20B header, sends to network**
- **If Ethernet, broken into 1500B packets with headers, trailers (24B)**
-



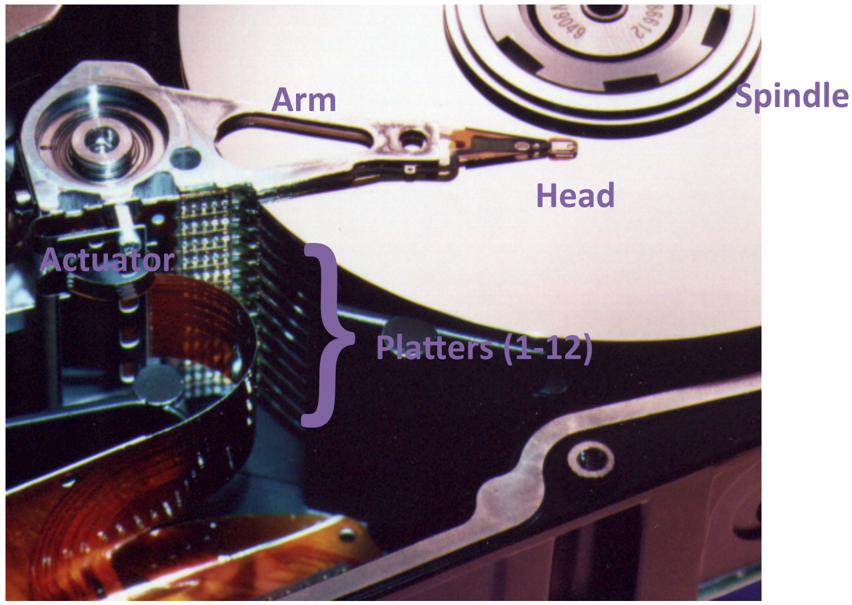
11/28/11 Fall 2011 -- Lecture #38 10

disks

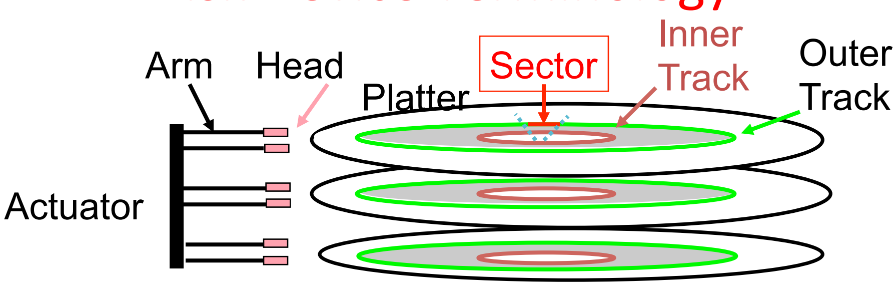
Magnetic Disk – common I/O device

- A kind of computer memory
  - Information stored by magnetizing ferrite material on surface of rotating disk
    - similar to tape recorder except digital rather than analog data
- Nonvolatile storage
  - retains its value without applying power to disk.
- Two Types
  - Floppy disks – slower, less dense, removable.
  - Hard Disk Drives (HDD) – faster, more dense, non-removable.
- Purpose in computer systems (Hard Drive):
  - Long-term, inexpensive storage for files
  - "Backup" for main-memory. Large, inexpensive, slow level in the memory hierarchy (virtual memory)

Photo of Disk Head, Arm, Actuator



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**

Where does Flash memory come in?

- Microdrives and Flash memory (e.g., CompactFlash going head-to-head
  - Both non-volatile (no power, data ok)
  - Flash benefits: durable & lower power (no moving parts, need to spin µdrives up/down)
  - Flash limitations: finite number of write cycles (wear on the insulating oxide layer around the charge storage mechanism). Most ≥ 100K, some ≥ 1M W/erase cycles.
- How does Flash memory work?
  - NMOS transistor with an additional conductor between gate and source/drain which "traps" electrons. The presence/ absence is a 1 or 0.

en.wikipedia.org/wiki/Flash_memory

en.wikipedia.org/wiki/Ipod www.apple.com/ipod

What does Apple put in its iPods?

Toshiba flash	Samsung flash	Toshiba 1.8-inch HDD	Toshiba flash
2GB	8, 16 GB	80, 160GB	8, 32, 64 GB

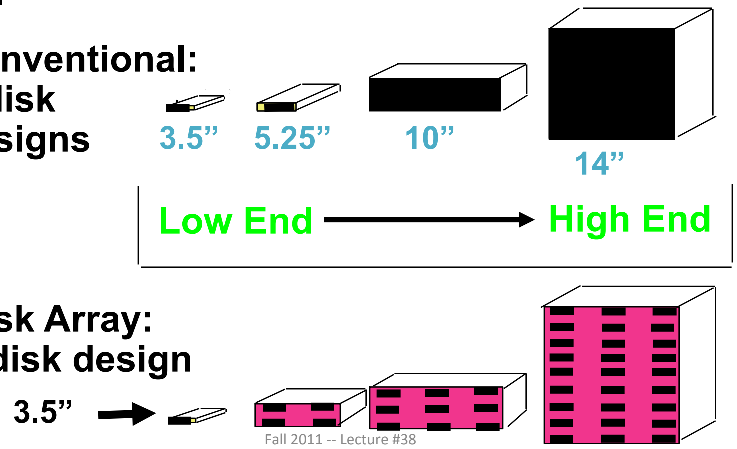
shuffle, nano, classic, touch

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

Disk Array: 1 disk design



Replace Small # of Large Disks with Large # of Small!

(1988 Disks)

	IBM 3390K	IBM 3.5" 0061
Capacity	20 GBytes	320 MBytes
Volume	97 cu. ft.	0.1 cu. ft.
Power	3 KW	11 W
Data Rate	15 MB/s	1.5 MB/s
I/O Rate	600 I/Os/s	55 I/Os/s
MTTF	250 KHrs	50 KHrs
Cost	$250K	$2K

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

Replace Small # of Large Disks with Large # of Small!

(1988 Disks)

	IBM 3390K	IBM 3.5" 0061	x70	
Capacity	20 GBytes	320 MBytes	23 GBytes	
Volume	97 cu. ft.	0.1 cu. ft.	11 cu. ft.	9X
Power	3 KW	11 W	1 KW	3X
Data Rate	15 MB/s	1.5 MB/s	120 MB/s	8X
I/O Rate	600 I/Os/s	55 I/Os/s	3900 I/Os/s	6X
MTTF	250 KHrs	50 KHrs	??? Hrs	
Cost	$250K	$2K	$150K	

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!

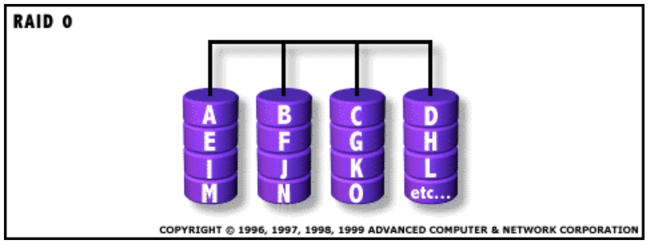
Redundant Arrays of (Inexpensive) Disks

- 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

RAID : Redundant Array of Inexpensive Disks

- Invented @ Berkeley (1989)
- A multi-billion industry
  80% non-PC disks sold in RAIDs
- Idea:
  - Files are "striped" across multiple disks
  - Redundancy yields high data availability
    - 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

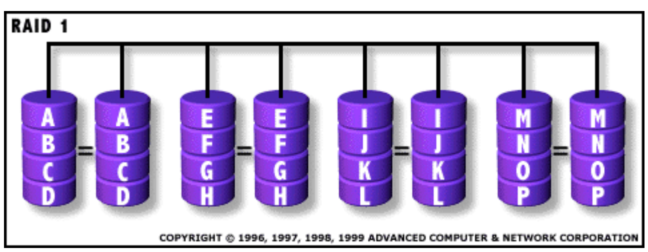
"RAID 0": No redundancy = "AID"



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

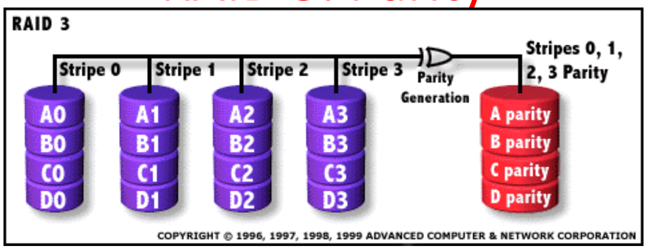
This and next 5 slides from **RAID.edu**, **http://www.acnc.com/04_01_00.html**
http://www.raid.com/04_00.html also has a great tutorial

RAID 1: Mirror data



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

RAID 3: Parity

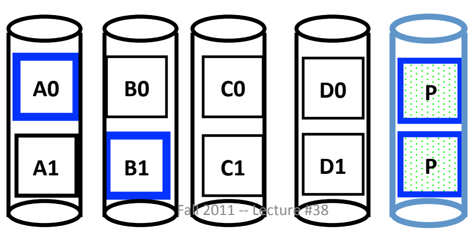


- Spindles synchronized, each sequential byte on a diff. drive
- Parity computed across group to protect against hard disk failures, stored in P disk
- Logically, a single high capacity, high transfer rate disk
- 25% capacity cost for parity in this example vs. 100% for RAID 1 (5 disks vs. 8 disks)
- Q: How many drive failures can be tolerated?

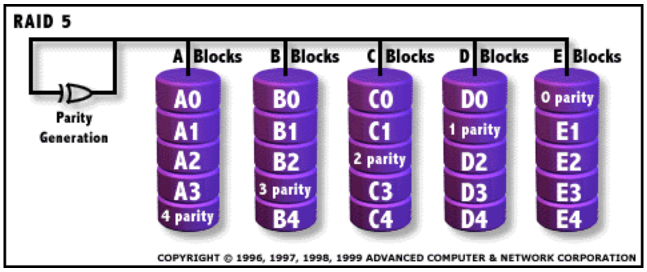
Inspiration for RAID 5 (RAID 4 block-striping)

- 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
- Parity Disk is bottleneck for Small writes: Write to A0, B1 ➔ both write to P disk

(RAID 4 block-striping)



RAID 5: Rotated Parity, faster small writes



- Independent writes possible because of interleaved parity
  - Example: write to A0, B1 uses disks 0, 1, 4, 5, so can proceed in parallel
  - Still 1 small write = 4 physical disk accesses

en.wikipedia.org/wiki/Redundant_array_of_independent_disks

"And in conclusion…"

- I/O gives computers their 5 senses
- I/O speed range is 100-million to one
- Processor speed means must synchronize with I/O devices before use: Polling vs. Interrupts
- Networks are another form of I/O
- Protocol suites allow networking of heterogeneous components
  - Another form of principle of abstraction
- RAID
  - Higher performance with more disk arms per $
  - More disks == More disk failures
  - Different RAID levels provide different cost/speed/reliability tradeoffs

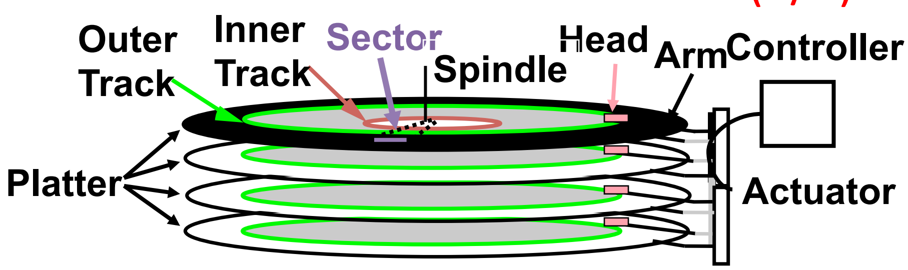
Peer Instruction

1. RAID 1 (mirror) and 5 (rotated parity) help with performance and availability
2. RAID 1 has higher cost than RAID 5
3. Small writes on RAID 5 are slower than on RAID 1

	ABC
0:	FFF
1:	FFT
2:	FTF
3:	FTT
4:	TFF
5:	TFT
6:	TTF
7:	TTT

BONUS SLIDES

Bonus: Disk Device Performance (1/2)



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

Bonus: Disk Device Performance (2/2)

- Average distance of 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
- Average no. tracks to move arm?
  - Disk industry standard benchmark:
    - Sum all time for all possible seek distances from all possible tracks / # possible
    - Assumes average seek distance is random
- Size of Disk cache can strongly affect perf!
  - Cache built into disk system, OS knows nothing

BONUS : Hard Drives are Sealed. Why?

- The closer the head to the disk, the smaller the “spot size” and thus the denser the recording.
  - Measured in Gbit/in2. ~60 is state of the art.
- Disks are sealed to keep the dust out.
  - Heads are designed to “fly” at around 5-20nm above the surface of the disk.
  - 99.999% of the head/arm weight is supported by the air bearing force (air cushion) developed between the disk and the head.