CS162
Operating Systems and Systems Programming
Lecture 22

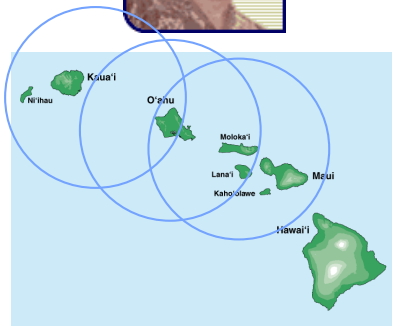
Networking II

April 13, 2010
Ion Stoica
http://inst.eecs.berkeley.edu/~cs162

Multiple Access Algorithm

- Single shared broadcast channel
  - Avoid having multiple nodes speaking at once
  - Otherwise, collisions lead to garbled data
- Multiple access mechanism
  - Distributed algorithm for sharing the channel
  - Algorithm determines which node can transmit
- Classes of techniques
  - Channel partitioning: divide channel into pieces
  - Taking turns: scheme for trading off who gets to transmit
  - Random access: allow collisions, and then recover
    - » *Optimizes for the common case of only one sender*

Random Access Protocol: AlohaNet



- Norm Abramson left Stanford in search of surfing
- Set up first radio-based data communication system connecting the Hawaiian islands
  - Hub at Alohanet HQ (Univ. Hawaii, Oahu)
  - Other sites spread among the islands
- Had two radio channels:
  - Random access: sites sent data on this channel
  - Broadcast: only used by hub to rebroadcast incoming data

Aloha Transmission Strategy

- When new data arrived at site, send to hub for transmission
- Site listened to broadcast channel
  - If it heard data repeated, knew transmission was rec'd
  - If it didn't hear data correctly, it assumed a collision
- If collision, site waited *random* delay before retransmitting
- Problem: Stability: what if load increases?
  - More collisions ⇒ less gets through ⇒more resent ⇒ more load… ⇒ More collisions…
  - Unfortunately: some sender may have started in clear, get scrambled without finishing

Ethernet

- Bob Metcalfe, Xerox PARC, visits Hawaii and gets an idea!
- Shared medium (coax cable)
- Can "sense" carrier to see if other nodes are broadcasting at the same time
  - Sensing is subject to time-lag
  - Only detect those sending a short while before
- Monitor channel to detect collisions
  - Once sending, can tell if anyone else is sending too

Ethernet's CSMA/CD

- CSMA: Carrier Sense Multiple Access
- CD: Collision detection
- Sense channel, if idle
  - If detect another transmission
    - » Abort, send jam signal
    - » Delay, and try again
  - Else
    - » Send frame
- Receiver accepts:
  - Frames addressed to its own address
  - Frames addressed to the broadcast address (broadcast)
  - Frames addressed to a multicast address, if it was instructed to listen to that address
  - All frames (promiscuous mode)

Ethernet's CSMA/CD (more)

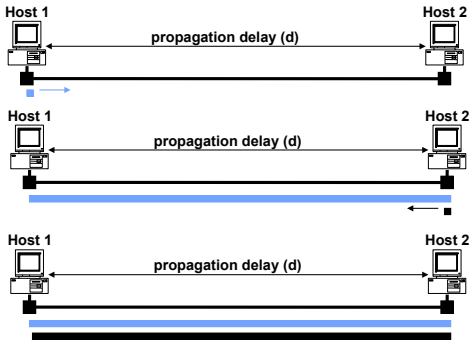
- Exponential back-off
  - Goal: adapt retransmission attempts to estimated current load
  - Heavy load: random wait will be longer
  - First collision: choose K from {0,1}; delay is K x 512 bit transmission times
  - After second collision: choose K from {0,1,2,3}…
  - After ten or more collisions, choose K from {0,1,2,3,4,…,1023}
- Minimum packet size
  - Give a host enough time to detect collisions
  - In Ethernet, minimum packet size = 64 bytes
  - What is the relationship between minimum packet size and the length of the LAN?

Minimum Packet Size (more)

a) Time = t; Host 1 starts to send frame

b) Time = t + d; Host 2 starts to send a frame, just before it hears from host 1's frame

c) Time = t + 2*d; Host 1 hears Host 2's frame → detects collision



d = LAN_length/ligh_speed = min_frame_size/(2*bandwidth) →

LAN_length = (min_frame_size)*(light_speed)/(2*bandwidth) =

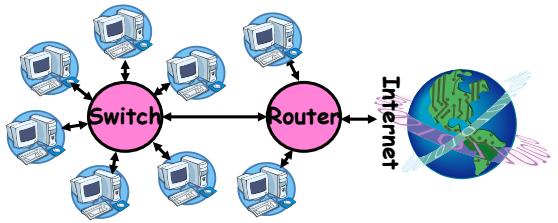
= (8*64b)*(2.5*10^8mps)/(2*10^7 bps) = 6400m approx

What about 100 mbps? 1 gbps? 10 gbps?

Goals for Today

- Networking
  - Network layer
  - Transport layer (start)
- MapReduce primer (project 4)

Review: Point-to-point networks



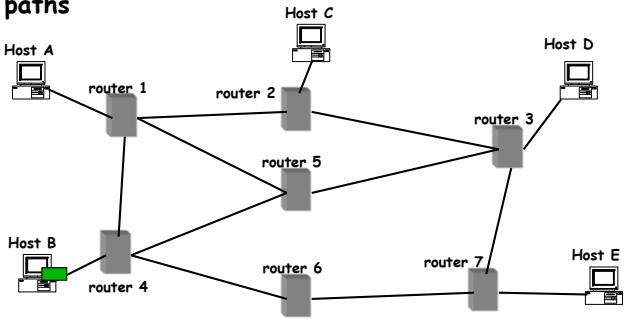
- Point-to-point network: a network in which every physical wire is connected to only two computers
- Switch: a bridge that transforms a shared-bus (broadcast) configuration into a point-to-point network.
- Hub: a multiport device that acts like a repeater broadcasting from each input to every output
- Router: a device that acts as a junction between two networks to transfer data packets among them.

Network (IP) Layer

- Deliver a packet to specified network destination
  - Packet forwarding & routing
- Perform segmentation/reassemble
- Others:
  - packet scheduling
  - buffer management
- Packet forwarding: the process of selecting outgoing link (next hop) to forward a packet
  - Usually done based on destination address
- Routing: the process of computing paths between end-points and building forwarding tables at routers

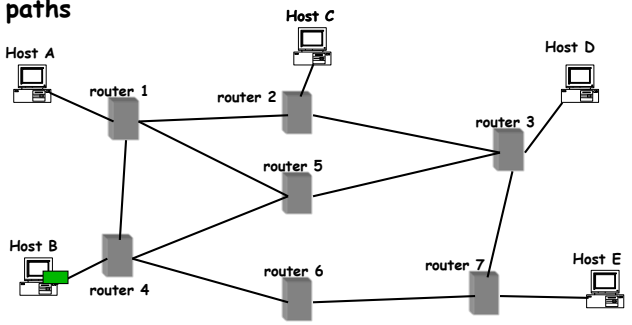
IP Routing

- Each packet is routed individually (like a letter)
- Packets of same connection may take different paths



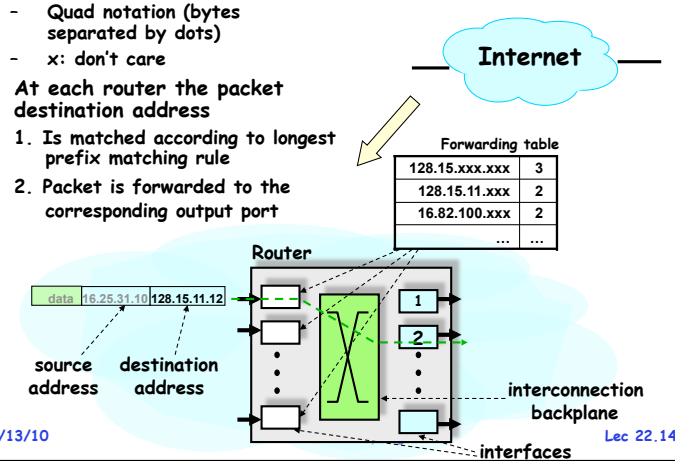
IP Routing

- Each packet is routed individually (like a letter)
- Packets of same connection may take different paths



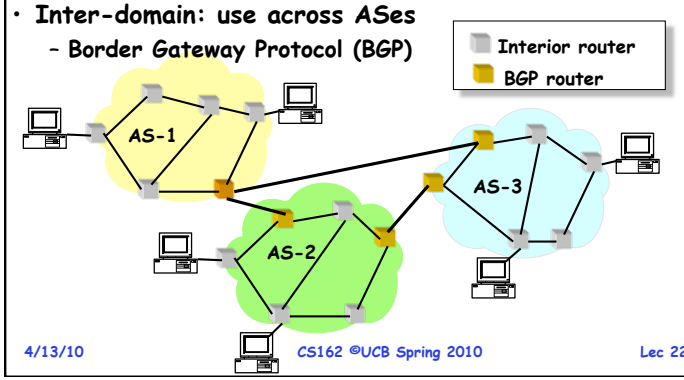
Packet Forwarding

- IP v4 addresses (32b)
  - Quad notation (bytes separated by dots)
  - x: don't care
- At each router the packet destination address
  1. Is matched according to longest prefix matching rule
  2. Packet is forwarded to the corresponding output port



Internet Routing: Two Level Hierarchy

- Autonomous system (AS): network owned by one admin. entity (e.g., ATT, Comcast)
- Intra-domain: routing within an AS
  - e.g., link state, distance vector protocols
- Inter-domain: use across ASes
  - Border Gateway Protocol (BGP)



Administrivia

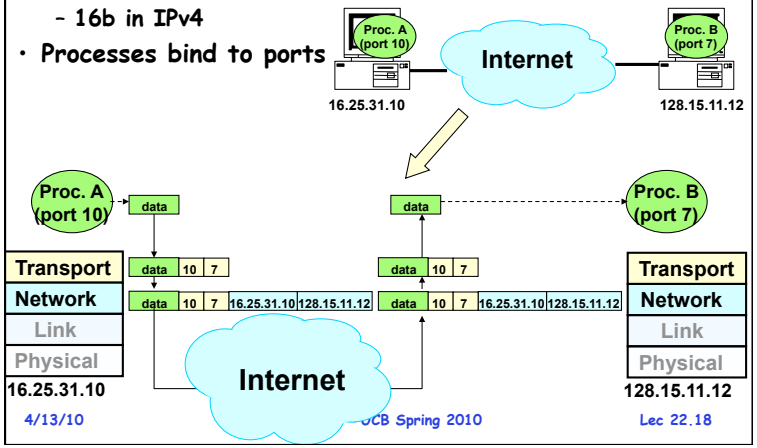
- I'll be away Wednesday-Friday (Eurosys)
  - Thursday's lecture will be taught by Ben
  - No office hour on Thursday, April 15
- Matei and Andy will be away as well
  - Ben will teach the discussion sections of both Matei and Andy
  - No office hours for Andy and Matei next week
- Project 4
  - Initial design, Wednesday (4/21), will give you two discussion sections before deadline
  - Code deadline, Wednesday (5/5), two weeks later

Transport Layer

- Demultiplex packets at the receiver: decide to which process to deliver a packet
- Others:
  - Flow control: protocol to avoid over-running a slow receiver
  - Congestion control: protocol to avoid over-running (congesting) the network
  - Reliability: recover packet losses
  - In-order delivery: deliver packets in the same order they were sent out
- Examples:
  - UDP (User Datagram Protocol): only demulteplexing
  - TCP (Transport Control Protocol): demultiplexing, flow & congestion control, reliability, in-order delivery

Transport Layer: Demultiplexing

- Ports: end-points at receiver to which packets are delivered
  - 16b in IPv4
- Processes bind to ports



What is Cloud Computing?

- "Cloud" refers to large Internet services that run on 10,000's of machines (Google, Yahoo!, etc)
- More recently, "cloud computing" refers to services by these companies that let external customers rent cycles
  - Amazon EC2: virtual machines at 8.5¢/hour, billed hourly
  - Amazon S3: storage at 15¢/GB/month
  - Windows Azure: special applications using Azure API
- Attractive features:
  - Scale: 100's of nodes available in minutes
  - Fine-grained billing: pay only for what you use
  - Ease of use: sign up with credit card, get root access

What is MapReduce?

- Data-parallel programming model for clusters of commodity machines
- Pioneered by Google
  - Processes 20 PB of data per day
- Popularized by open-source Hadoop project
  - Used by Yahoo!, Facebook, Amazon, …
- Hadoop: open source version of MapReduce
  - http://hadoop.apache.org/

What is MapReduce Used For?

- At Google:
  - Index building for Google Search
  - Article clustering for Google News
  - Statistical machine translation
- At Yahoo!:
  - Index building for Yahoo! Search
  - Spam detection for Yahoo! Mail
- At Facebook:
  - Data mining
  - Ad optimization
  - Spam detection

MapReduce Goals

- Scalability to large data volumes:
  - Scan 100 TB on 1 node @ 50 MB/s = 24 days
  - Scan on 1000-node cluster = 35 minutes
- Cost-efficiency:
  - Commodity nodes (cheap, but unreliable)
  - Commodity network
  - Automatic fault-tolerance (fewer admins)
  - Easy to use (fewer programmers)

Typical Hadoop Cluster

Challenges

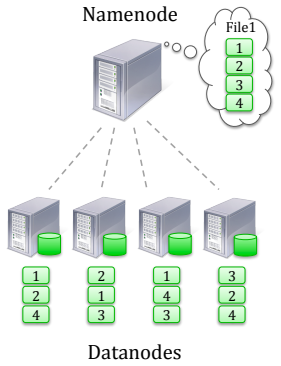
- Cheap nodes fail, especially if you have many
  - Mean time between failures for 1 node = 3 years
  - MTBF for 1000 nodes = 1 day
  - Solution: Build fault-tolerance into system
- Commodity network = low bandwidth
  - Solution: Push computation to the data
- Programming distributed systems is hard
  - Solution: Users write data-parallel "map" and "reduce" functions, system handles work distribution and failures

Hadoop Components

- Distributed file system (HDFS)
  - Single namespace for entire cluster
  - Replicates data 3x for fault-tolerance
- MapReduce framework
  - Runs jobs submitted by users
  - Manages work distribution & fault-tolerance
  - Colocated with file system

Hadoop Distributed File System

- Files split into 128MB blocks
- Blocks replicated across several datanodes (usually 3)
- Namenode stores metadata (file names, locations, etc)
- Optimized for large files, sequential reads
- Files are append-only



MapReduce Programming Model

- Data type: key-value *records*
- Map function:

  $(K_{in}, V_{in}) \rightarrow list(K_{inter}, V_{inter})$
- Reduce function:

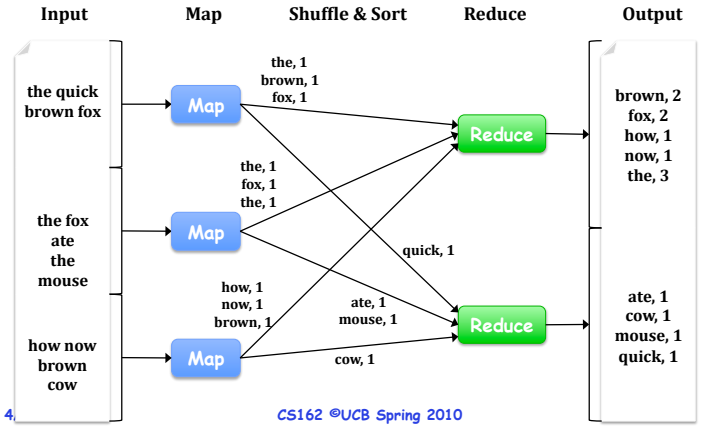
  $(K_{inter}, list(V_{inter})) \rightarrow list(K_{out}, V_{out})$

Example: Word Count

```
def mapper(line):
    foreach word in line.split():
        output(word, 1)


def reducer(key, values):
    output(key, sum(values))
```

Word Count Execution



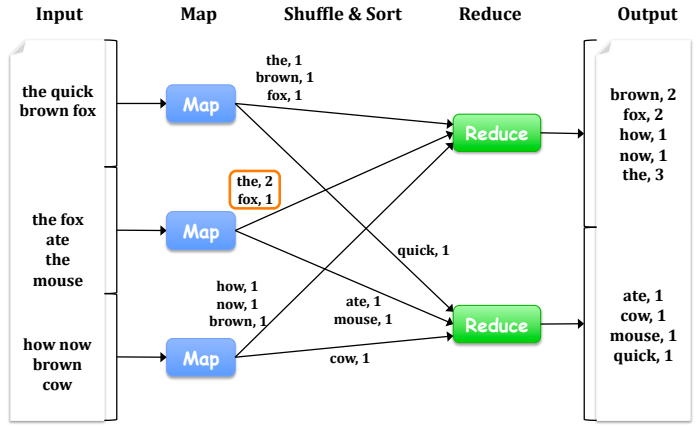
An Optimization: The Combiner

- Local reduce function for repeated keys produced by same map
- For associative ops. like sum, count, max
- Decreases amount of intermediate data

- Example: local counting for Word Count:

```
def combiner(key, values):
    output(key, sum(values))
```


Word Count with Combiner



MapReduce Execution Details

- Mappers preferentially scheduled on same node or same rack as their input block
  - Push computation to data, minimize network use

- Mappers save outputs to local disk before serving to reducers
  - Allows running more reducers than # of nodes
  - Allows recovery if a reducer crashes

Conclusion

- Network layer
  - IP packet forwarding: based on longest-prefix match
- Transport layer
  - Multiplexing and demultiplexing via port numbers
  - UDP gives simple datagram service
  - TCP gives reliable byte-stream service
- MapReduce (Hadoop)
  - Data-parallel programming model for clusters of commodity machines