Computer Networks
CS 552

Badri Nath
Rutgers University
badri@cs.rutgers.edu
SPRING 2017
About us: Management

- **Professor: Badri Nath**
  - http://www.cs.rutgers.edu/~badri
  - badri@cs.rutgers.edu
  - Office Hours: Wednesday 1:30-3:30 PM; CORE 320

- **TA: Long T Le**
  - longtle@cs.rutgers.edu
  - Office Hours: Tuesday 10:00-11:00 AM; Hill 202

- **Course info**
  - http://www.cs.rutgers.edu/~badri/552.html
Course Web Page

- Course schedule
- Reading list
- Lecture notes
- Announcements
- Assignments
- Project ideas
- Exams
Sakai Web page

- [https://sakai](https://sakai)
- Will submit reviews or answer to mini-quizzes (based on the paper) online
- Course announcements
- Written Homeworks
Course Goals

- Understand the basic principles of computer networks, in particular the Internet
- Study new concepts, design principles in network protocols and design
- How to do network research
  - How to determine what is important
  - What are the trends
    - Internet of things (IoT), Datacenter, cloud, SDN,
    - Cloud vs edge:
    - connected devices/home, connected vehicles
    - If software is eating the world, networking is enabling it
  - What are the economics, technology that is driving innovation
    - Cost, performance, energy, availability, security
    - Sharing Economy: airbnb, uber
Course Materials

- Research papers
  - Links to pdf on Web page
  - Combination of classic and recent work
  - ~30 papers
  - Optional readings

- Recommended textbooks
  - For students not familiar with networking
  - Peterson & Davie (4th edition)
  - Alternative: Kurose & Ross
Reading papers

- Understand /identify the basic idea
  - What is the problem that the paper tackles?
- What kind of a paper?
  - Performance, vision, new direction/protocol paper
- Summarize key idea
- +ve aspects of the paper
  - New, breakthrough, incremental,
- -ve aspects of the paper
  - Readability, Assumptions (valid?), scaling issues (does it scale), implementation (has it been implemented), measurements (problems?)


• **TCP/IP Illustrated, Volume 1: The Protocols** by W. Richard Stevens.

• **Unix Network Programming: Networking APIs: Sockets and XTI (Volume 1)** by W. Richard Stevens.


• I or 2 recommended
Grading

- 15% Paper summaries/reviews/HWs based on Papers
  - A subset of the papers will be assigned for submitting summary/critique
  - All papers assigned should be read as quizzes/Hws will be based on these papers
- 35% Programming project (two-person)
- 20% Mid term
- 30% Final
- Honor code
  - All submitted work should be yours
  - You are all grad students!!
Class Coverage

- Quick overview of undergraduate networking
- Pre requisite: 352 or equivalent
- Students expected to know
  - Link layer, basic IP routing, TCP,
- Focus on Advanced topics in networking (from papers in recent SIGCOMM, NSDI)
- Course will deal with:
  - Services and Protocols
  - Investigate protocol trade-offs, new cost models
  - New Workloads, new technologies, new services
  - Scheduling, QoS, Load balancing
  - Software defined networking
Class etiquette

- Cell phones in off position
- No FB status updates in class
- If you need to surf while in class (I prefer not), do not disturb your neighbors
- Stop me anytime to ask questions
  - Prof may not know the answer!!
- This is a graduate class, student participation in class is important
  - Challenge the class, the prof, and ideas in papers
Lecture Topics

Traditional
- Layering
- Internet architecture
- Routing (IP, BGP)
- Transport (TCP)

Recent Topics
- Energy, $$$ Considerations
- Measurements
- CDN/Video/DASH
- Datacenter networking
- Cloud(IaaS)
- Software defined networking

Trends: Desktop/Internet ➔ Mobile/Cloud ➔ Edge computing
What is a Network?

- Carrier of information between 2 or more entities
- Some carry objects/people (postal, air, surface transport)
- Most important is the services offered
- User expectation of service
  - Latency, cost, reliability, service interface, others
- We focus on computer networks
- Interconnection may be any medium capable of communicating information:
  - copper wire
  - Lasers (optic fibre)
  - Microwave
  - Cable (coax)
  - wireless
  - satellite link
- Example: Ethernet, Wifi, 3G, LTE
Why Networks?

- Availability of Resources
  - Resources become available regardless of the user’s physical location (server based, peer2peer)

- Load Sharing/utilization
  - Jobs processed on least crowded machine
  - Resources can be shared

- High Reliability
  - Alternative source of supply (multiple copies)

- Human-to-Human Communication
  - e.g., on-line world, e-commerce
Domain name growth

Hobbes' Internet Timeline Copyright ©2016 Robert H Zakon
http://www.zakon.org/robert/internet/timeline/
What is Internet Technology?

- What is an internet?
  - Network of networks
- What is the Internet?
  - A global internet based on the IP protocol
- To what does “Internet technology” refer?
  - Architecture, services, interfaces, and protocols
DATA NEVER SLEEPS 3.0

How much data is generated every minute?

Data is being created all the time without us even noticing it. Much of what we do every day now happens in the digital realm, leaving an ever-increasing digital trail that can be measured and analyzed. Just how much data do our tweets, likes and photo uploads really generate? For the third time, Domo has the answer—and the numbers are staggering.

**UBER**
- 110,040 calls
- 694 passengers take rides

**Facebook**
- 1.466 million users like
- 347,222 tweets

**Twitter**
- 3.17 million users send

**Snapchat**
- 284,722 snaps

**Skype**
- 34,150 videos

**Buzzfeed**
- 1.041 million videos

**Snapchat**
- 284,722 shares

**Tinder**
- 590,278 swipes

**Vine**
- 1.041 million videos

**Amazon**
- 4,310 unique visitors

**Netflix**
- 77,160 hours of video

**Apple**
- 34,000 apps

**Reddit**
- 18,327 votes

**Google**
- 2.4 million queries

**YouTube**
- 300 hours of new video

**Instagram**
- 1.736 million photos

**Pinterest**
- 9,722 images

**YouTube**
- 200,000,000

Number of monthly active WhatsApp users worldwide from April 2013 to September 2015 (in millions)

Sources:
WhatsApp, various sources (language) & Statista 2015

Additional Information:
Worldwide: WhatsApp, April 2013 to September 2015

**20 M** msgs/minute

**2.4 M** Queries/minute

Scale Scale Scale

THE GLOBAL INTERNET POPULATION CREW
18.5% FROM 2013–2015 AND NOW REPRESENTS
3.2 BILLION PEOPLE.

With each click, share and like, the world’s data pool is expanding faster than we can comprehend. Businesses today are paying attention to scores of data sources to make crucial decisions about the future. The team at Domo can help your business make sense of this endless stream of data by providing executives with all their critical information in one intuitive platform. Domo delivers the insights you need to transform the way you run your business. Learn more at www.domo.com.

Sources:
FACEBOOK, TWITTER, YOUTUBE, INSTAGRAM, PINTEREST, APPLE, NETFLIX, REDDIT, AMAZON, TINDER, BUZZFEED, STATISTA, INTERNET LIVE STATS, STATISTORIAN.COM
Impact on society

- Net neutrality
  - ISPs should not “discriminate”
  - For and against case
- Nations’ laws and Internet
  - Regulation across national boundaries
- Privacy
  - Ears and Eyes in your house connected to the net.
  - Who can ask for that?
  - Is Alexa’s (Amazon) recordings available to others?
- Content creation, ownership, distribution, online piracy
- Cyber Warfare – N Korea, Hacking (political objectives)
Internet Players

- Users, people who use the applications
  - Everyone (mom and pop, kids)
  - get something done (hopefully useful)

- Designers
  - You: protocol design and implementation
  - Scale, performance, cost, incremental deployment

- Service Providers
  - Administrators and ISPs
  - Datacenter operators
  - Provider-customer versus peer-to-peer
  - Management, revenue, deployment

- Market/business models for the Internet
  - Consumer to consumer (ebay, match.com, craigslist, airbnb), Business to consumer (amazon, orbitz, google, netflix, hulu), Business to business (getty, harvest, google), Consumer to business (hot jobs, monster, linkedin)
Internet Growth

1995
35MM+ Internet Users
0.6% Population Penetration

2014
2.8B Internet Users
39% Population Penetration

Mary meeker : Internet trends  May 2015
# Internet Growth

## 2.4B Global Internet Users in 2012* – 8% Y/Y Growth*, Driven by Emerging Markets

<table>
<thead>
<tr>
<th>Rank</th>
<th>Country</th>
<th>2008-2012 Internet User Adds (MMs)</th>
<th>2012 Internet Users (MMs)</th>
<th>Y/Y Growth</th>
<th>Population Penetration</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>China</td>
<td>282</td>
<td>538</td>
<td>10%</td>
<td>40%</td>
</tr>
<tr>
<td>2</td>
<td>India</td>
<td>88</td>
<td>137</td>
<td>26</td>
<td>11</td>
</tr>
<tr>
<td>3</td>
<td>Indonesia</td>
<td>39</td>
<td>55</td>
<td>58</td>
<td>23</td>
</tr>
<tr>
<td>4</td>
<td>Iran</td>
<td>35</td>
<td>42</td>
<td>205</td>
<td>55</td>
</tr>
<tr>
<td>5</td>
<td>Russia</td>
<td>33</td>
<td>70</td>
<td>6</td>
<td>49</td>
</tr>
<tr>
<td>6</td>
<td>Nigeria</td>
<td>31</td>
<td>48</td>
<td>15</td>
<td>30</td>
</tr>
<tr>
<td>7</td>
<td>Philippines</td>
<td>28</td>
<td>34</td>
<td>32</td>
<td>35</td>
</tr>
<tr>
<td>8</td>
<td>Brazil</td>
<td>27</td>
<td>88</td>
<td>6</td>
<td>45</td>
</tr>
<tr>
<td>9</td>
<td>Mexico</td>
<td>19</td>
<td>42</td>
<td>9</td>
<td>37</td>
</tr>
<tr>
<td>10</td>
<td>USA</td>
<td>18</td>
<td>244</td>
<td>3</td>
<td>78</td>
</tr>
<tr>
<td>11</td>
<td>Argentina</td>
<td>17</td>
<td>28</td>
<td>57</td>
<td>68</td>
</tr>
<tr>
<td>12</td>
<td>Egypt</td>
<td>17</td>
<td>30</td>
<td>11</td>
<td>38</td>
</tr>
<tr>
<td>13</td>
<td>Colombia</td>
<td>14</td>
<td>25</td>
<td>39</td>
<td>54</td>
</tr>
<tr>
<td>14</td>
<td>Turkey</td>
<td>13</td>
<td>35</td>
<td>17</td>
<td>47</td>
</tr>
<tr>
<td>15</td>
<td>Vietnam</td>
<td>12</td>
<td>31</td>
<td>7</td>
<td>35</td>
</tr>
<tr>
<td>Top 15</td>
<td></td>
<td>673</td>
<td>1,447</td>
<td>15%</td>
<td>34%</td>
</tr>
<tr>
<td>World</td>
<td></td>
<td>902</td>
<td>2,406</td>
<td>8%</td>
<td>34%</td>
</tr>
</tbody>
</table>

Note: *Data as of 6/12, 2.4B global Internet users and 8% Y/Y growth rate based on the latest available data. Source: United Nations / International Telecommunications Union, internetworldstats.com.
WWW growth

<table>
<thead>
<tr>
<th>DATE</th>
<th>SITES</th>
</tr>
</thead>
<tbody>
<tr>
<td>12/90</td>
<td>1</td>
</tr>
<tr>
<td>12/91</td>
<td>10</td>
</tr>
<tr>
<td>12/92</td>
<td>50</td>
</tr>
<tr>
<td>06/93</td>
<td>130</td>
</tr>
<tr>
<td>09/93</td>
<td>204</td>
</tr>
<tr>
<td>10/93</td>
<td>228</td>
</tr>
<tr>
<td>12/93</td>
<td>623</td>
</tr>
<tr>
<td>06/94</td>
<td>2,738</td>
</tr>
<tr>
<td>12/94</td>
<td>10,022</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DATE</th>
<th>SITES</th>
</tr>
</thead>
<tbody>
<tr>
<td>06/95</td>
<td>23,500</td>
</tr>
<tr>
<td>01/96</td>
<td>100,000</td>
</tr>
<tr>
<td>06/96</td>
<td>252,000</td>
</tr>
<tr>
<td>01/97</td>
<td>646,162</td>
</tr>
<tr>
<td>06/97</td>
<td>1,117,259</td>
</tr>
<tr>
<td>01/98</td>
<td>1,834,710</td>
</tr>
<tr>
<td>06/98</td>
<td>2,410,067</td>
</tr>
<tr>
<td>01/99</td>
<td>4,062,280</td>
</tr>
<tr>
<td>07/99</td>
<td>6,598,697</td>
</tr>
</tbody>
</table>

Hobbes' Internet Timeline Copyright ©2016 Robert H Zakon
http://www.zakon.org/robert/internet/timeline/
80% of Top Ten Global Internet Properties ‘Made in USA’…
81% of Users Outside America
Astounding growth of Google

Average Percentage of End Devices Sending Traffic to Google Servers in North American Consumer Networks during Summer 2013

DEEPFIELD RANK

#1

62.28 %

Percent Users

% of Users

3. Jun  
17. Jun  
1. Jul  
15. Jul
Facebook growth

MAU: Number of unique users in the past 30 days
Social Networking/Messaging

**Gossip beats learning**
US website traffic, % of total visitors

- **Facebook**
- **Google**

Source: Experian Hitwise

**Faster than Facebook**
Active users, m

- **WhatsApp**
- **Facebook**
- **Twitter**

Source: comScore, Facebook, WhatsApp
Messaging continues to grow rapidly...
Leaders = WhatsApp / Facebook Messenger / WeChat

Monthly Active Users on Select Social Networks and Messengers,
Global, 2011 – 2015

- LinkedIn
- Twitter
- Instagram
- WhatsApp
- WeChat
- Facebook Messenger

Initially: Only kids!!

February 2004 to January 2005

Facebook begins at Harvard, and expands to a few universities at a time.
Now: everyone; grandpa, grandma

Facebook is translated into more than 40 languages. The fastest-growing group of members is people over 35.
Mobile Phone usage

1995
80MM+ Mobile Phone Users
1% Population Penetration

2014
5.2B Mobile Phone Users
73% Population Penetration

- 60% Feature Phone
- 40% Smartphone
Mobile Traffic as % of Global Internet Traffic =
Growing 1.5x per Year & Likely to Maintain Trajectory or Accelerate

Global Mobile Traffic as % of Total Internet Traffic, 12/08 – 5/13
(with Trendline Projection to 5/15E)

% of Internet Traffic

- 30%
- 25%
- 20%
- 15%
- 10%
- 5%
- 0%

12/08  12/09  12/10  12/11  12/12  12/13E  12/14E

Trendline

0.9% in 5/09
2.4% in 5/10
6% in 5/11
10% in 5/12
15% in 5/13
Mobile OS trends

Global Smartphone Operating System Market Share (by Units Shipped), 2005 vs. 2010 vs. 2013

- 2005:
  - Other OS: 40%
  - BlackBerry OS: 30%
  - iOS: 20%
  - Android: 10%
  - Windows Phone: 0%

- 2010:
  - Other OS: 30%
  - BlackBerry OS: 25%
  - iOS: 30%
  - Android: 15%
  - Windows Phone: 10%

- 2013:
  - Other OS: 20%
  - BlackBerry OS: 10%
  - iOS: 50%
  - Android: 25%
  - Windows Phone: 5%
Can nodes be selected dynamically?
Video Content Growth

YouTube Hours of Video Uploaded per Minute, 6/07 – 5/13

Mary Meeker Internet trends 2013
Data Producers everywhere

Long Ago, People Danced @ Concerts, Now They Video / Click / Share / Tweet…

1990s

2010s
Re-Imagination of User Interfaces...

THEN...
(Keyboard + Mice = Graphical User Interface)

NOW...
(Touch + Voice + Gesture = Natural User Interface)
Reimagining shared economy

Renting a place

Hailing a cab

Renting your car

Airbnb

Uber

On demand food delivery

Supply/Demand Exchanges
What’s hot?

- Software Defined Networking (SDN)
- Internet
- Social Networking
- Data centers
  - Edge vs Cloud
- Embedded
  - Internet of things (IoT)
  - Internet Bodily of Things (IoB)
- Mobile & Wireless
- Bots

http://a16z.com/2016/12/16/the-end-of-cloud-computing/
Circle of networking Architecture

mainframes

Desktop

Edge computing

Mobile/cloud

Action is all at the edge: cars, drones, robots
Edge is where the action will be

- Car: data center on wheels
- Drones: data center on wings
- Robots: data center with arms and legs
- Flying cars?
- Mainframe to desktop to cloud to edge
- Sense, infer, query, act
  - Watch the Video By Levine: return to the edge
  - [http://a16z.com/2016/12/16/the-end-of-cloud-computing/](http://a16z.com/2016/12/16/the-end-of-cloud-computing/)
Device Growth: connected

Computing Growth Drivers Over Time, 1960 – 2020E

- 1MM+ Units
- 10MM+ Units
- 100MM+ Units
- PC
- Minicomputer
- Mainframe
- Cellphone / Desktop Internet
- Mobile Consumer
- 1B+ Units / Users
- 10B+ Units???
- Car Electronics
  - GPS, ABS, A/V
- Mobile Video
- Home Entertainment
- Wireless Home Appliances
- Smartphone
- Kindle
- Tablet
- MP3

Note: PC installed base reached 100MM in 1993, cellphone / internet users reached 1B in 2002 / 2003 respectively;
Source: ITU, Mark Lipakis, Morgan Stanley Research.
Smart cities/Smart Campuses
Software defined urban platforms

If software is eating the world, networking is enabling it
Objective of networking

- End-hosts to communicate
  - Applications running on end-hosts
- Different technologies
- Different protocols
- Different Services
How to communicate?

- Circuit switching
  - Establish a connection before communicating
    - POTS (plain old telephone system)
    - Dedicated pipe for the duration of the session
- Packet switching
  - Multiplex communication from different sources
  - Every packet is self contained
  - Efficient use of resources
  - NO guarantees on performance
How to handle different networks?

- Many differences between networks
- How to translate between various network technologies?
- Have a common protocol for inter network communication
  - IP
  - A set of rules with a well-defined interface
How to locate a node?

- Naming, discovery and routing
  - Network elements needed to support directory
  - Network elements needed to support forwarding towards destination
- Scalable
- Reliable
How to meet application demands?

- Corruption?
  - Need error detection and correction
- Reliability
  - Data lost?
- Overload
  - Congestion control
- Security
  - Encryption, authentication
Lots of Functions Needed

- Link
- Multiplexing
- Routing
- Addressing/naming (locating peers)
- Reliability
- Flow control
- Fragmentation
- Etc....
ISO OSI Layering Architecture

Host A

- Application Layer
- Presentation Layer
- Session Layer
- Transport Layer
- Network Layer
- Data Link Layer
- Physical Layer

Host B

- Application Layer
- Presentation Layer
- Session Layer
- Transport Layer
- Network Layer
- Data Link Layer
- Physical Layer

Router

Network Protocol
- Application Protocol
- Presentation Protocol
- Session Protocol
- Transport Protocol

Diagram showing the ISO OSI Layering Architecture with data flow between Host A and Host B through routers.
Problems

- Seven layers not widely accepted
- Standardized before implemented
- Top three layers fuzzy
- Internet or TCP/IP layering widespread
TCP/IP Layering Architecture

- A simplified model
- The network layer
  - Hosts drop packets into this layer, layer routes towards destination- only promise- try my best
- The transport layer
  - Reliable/unreliable byte-oriented stream
# Internet design philosophy

- **#1 Functionality at the edge as opposed to core**
- In Telephone network it is the opposite
  - Any new service, the phone company has to provide
  - Edge device is dumb
- **Smart device at the edge means programmability**
- **New services can be supported, drives innovation**
  - VOIP (SIP), IM
- **Cathedral vs Bazaar**

David Clark, The design philosophy of the DARPA internet protocols, 1998
Implications (cathedral vs bazaar)

- **Bazaar**
  - Edge is programmable
  - Nimble, novel applications

- **Cathedral**
  - Core elements still rigid
  - Standards, slow evolution
  - Can't do anything radical

- Where is programmability in the system?
### #2 Multiplexed utilization

- **Best effort, packet switching**
- **Keep the network simple**
  - Packets may be lost, corrupted, out-of-order
  - Let the end host implement any other requirements
  - Want reliability?
    - Retransmit from sender
- **Packets self contained**
  - Can take different routes
  - Different transfers on the same link
- **Stateless in the core**
  - End hosts can maintain state
  - **Fate-sharing** (If I Die, my state will die but will not affect others)
#3 Support multiple networks

- IP over anything, anything over IP
  - Run over any type of link
  - Build any end-to-end protocol over IP
#4 distributed management

- Need only know local information
- Information distributed over different nodes
  - Scalable
  - No single hot spot
- Distributed functionality-- Roles
  - Different entities manage different parts of the system
  - Impact on naming, routing, addressing
- Local and Global management authorities
Hierarchical topology but this too is changing
A New Internet Model

Major elements:
CDN Content distribution networks
IXPs Internet Exchange points
Datacenters
Facebook DC

4x40 G RSW

Internet Design Principles

- **Scale**
  - Protocols should work in networks of all sizes and distances
- **Incremental deployment**
  - New protocols need to be deployed gradually
- **Heterogeneity**
  - Different technologies, autonomous organizations
- **End-to-end argument**
  - Networking functions should be delegated to the edges; application knows best
End-to-end argument

- Saltzer, reed and clark [1984]
  - End-to-end arguments in system design
- Main idea
- If a function can only be completely and correctly implemented with the knowledge and help of the applications standing at the communication end points. Hence providing this function in the subsystem is not possible
  - Complexity at the edges as opposed to the core
- Simply stated, the argument suggests that functions placed at the low levels of a system may be redundant or of little value when compared with the cost of providing them at that low level.
- Don’t force feature, service, restriction on the end points
Communication system

- An end system connected by a communication subsystem
- Questions?
- Who is responsible for a given function
  - Subsystem?
  - End units?
  - Or both (redundant) or jointly?
End-to-end argument

- Functions placed at lower level implies specific problems being solved in a general way
- Best aim:
- Simple lower layer with smart end points
  - Basic and general functions at the lower layers
- Gives flexibility
e2e argument

- Low level system may not have all the information to implement the given functionality
- Implement only for performance (wireless links)
- Low level system shared by all applications – what if the application does not need the feature
- Performance
- Duplicated effort
- Should not impact applications that do not use that functionality
New business models
- Network caching, redirection, proxy transcoding

Wireless Application protocols
- Gateway provides a box for content translation

Network redirection
- Network level switch for load balancing

Balance between performance, layering, e2e argument
New metrics

- Energy/power
  - Always-on system consumes a lot of power
- System Performance - Latency, throughput, Availability
  - How to guarantee performance on shared Infrastructure? IaaS
  - Latency, throughput, availability Request=<L,B, N>
  - How to measure?
- Cost <$>
  - Hardware cost is falling, software lease cost
  - Opex vs capex debate
  - Cloud/Software as a Service (SaaS) models
  - Cost proportional to usage
  - How to bid? How to optimize?
Typical Datacenter networking

The cost of a cloud: Research Problems in data center networks by Albert Greenberg et.al, CCR
Data center traffic (Of Mice and Elephants): Latency, BW, $ as metric

- 50% of flows < 100 KB
- <5% of bytes
- 5% of flows > 10 MB
- 30% of bytes

DC network evolution

- North-South Traffic
- N-Tier Network
- Performance Limit
- Expensive, Hard to Manage

- East-West Traffic
- Flat Network
- Fast Interconnect
- Scalable and Manageable
Datacenter cost

- Servers: 45%
- Power: 15%
- Infrastructure: 25%
- Network: 15%

- 50,000 server @ 3K a pop, 5% cost of money, 3YR
  - 52.5 M/Yr cost
- Power cost
- Power to run the IT equipment
- Power to run cooling, UPS etc – Overhead
- PUE=Total power /IT power
- 1.2 ideal -- 20% overhead
- Typically 2 to 3 PUE – Air conditioning costs enormous
Facts and Figures
[Quereshi09]

- Servers are power hungry (annual electricity bills)

<table>
<thead>
<tr>
<th>Company</th>
<th>Servers</th>
<th>Electricity</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>eBay</td>
<td>16K</td>
<td>$0.6 \times 10^5$ MWh</td>
<td>$3.7M</td>
</tr>
<tr>
<td>Akamai</td>
<td>40K</td>
<td>$1.7 \times 10^5$ MWh</td>
<td>$10M</td>
</tr>
<tr>
<td>Rackspace</td>
<td>50K</td>
<td>$2 \times 10^5$ MWh</td>
<td>$12M</td>
</tr>
<tr>
<td>Microsoft</td>
<td>&gt;200K</td>
<td>$6 \times 10^5$ MWh</td>
<td>$36M</td>
</tr>
<tr>
<td>Google</td>
<td>&gt;500K</td>
<td>$6.3 \times 10^5$ MWh</td>
<td>$38M</td>
</tr>
<tr>
<td>USA (2006)</td>
<td>10.9M</td>
<td>$610 \times 10^5$ MWh</td>
<td>$4.5B</td>
</tr>
<tr>
<td>MIT campus</td>
<td></td>
<td>$2.7 \times 10^5$ MWh</td>
<td>$62M</td>
</tr>
</tbody>
</table>
Energy-proportional metric

- Can we design networks that consume power proportional to utilization?
Figure 1. Average CPU utilization of more than 5,000 servers during a six-month period. Servers are rarely completely idle and seldom operate near their maximum utilization, instead operating most of the time at between 10 and 50 percent of their maximum.
Throughput proportional fabric

- Does throughput offered rise/fall in proportional to traffic sources/sinks?
- Fat-free topologies paper in Hotnets 2016
Figure 2. Server power usage and energy efficiency at varying utilization levels, from idle to peak performance. Even an energy-efficient server still consumes about half its full power when doing virtually no work.

Energy Efficiency = Utilization/Power

<table>
<thead>
<tr>
<th>util</th>
<th>power</th>
<th>EE</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.15</td>
<td>0.6</td>
<td>.25</td>
</tr>
<tr>
<td>0.4</td>
<td>0.7</td>
<td>.57</td>
</tr>
</tbody>
</table>

Doing nothing well? Still power
Energy Proportional Computing

Can we do this for Networking Infrastructure?

Energy Efficiency = Utilization/Power

Figure 4. Power usage and energy efficiency in a more energy-proportional server. This server has a power efficiency of more than 80 percent of its peak value for utilizations of 30 percent and above, with efficiency remaining above 50 percent for utilization levels as low as 10 percent.
Energy/ Power metric

- If Cost/Green Conscious
- Make network elements less power hungry
- Take advantage of lower power rates elsewhere, time-of-day
What is the energy consumption of the internet?

- Energy consumed by networking equipment such as routers, switches, hubs etc
- Does not include hosts
- Internet energy consumption controversial data
### Facts and Figures [Gupta & Singh - 03]

<table>
<thead>
<tr>
<th>Device</th>
<th>Approximate Number Deployed</th>
<th>Total AEC TW-h</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hubs</td>
<td>93.5 Million</td>
<td>1.6 TW-h</td>
</tr>
<tr>
<td>LAN Switch</td>
<td>95,000</td>
<td>3.2 TW-h</td>
</tr>
<tr>
<td>WAN Switch</td>
<td>50,000</td>
<td>0.15 TW-h</td>
</tr>
<tr>
<td>Router</td>
<td>3,257</td>
<td>1.1 TW-h</td>
</tr>
</tbody>
</table>

P=E/T  \( 1W = 1 \text{ J}/\text{S} \)  \( 1 \text{ Kw-H} = 3600000 \text{ joules} = 10 \times 100W \text{ bulbs for 1 Hr} \)
More Numbers...

- Total energy consumed by networking devices annually in 2000 (US): 6.05 TW-h
- Amounts to about 0.07 % of total U.S. energy expenditure
- Expected increase: +1 TW-h by 2005
  - Note: This does not include energy consumed by hosts, UPS supplies or cooling equipment.
Enterprise networks

- Network Switches hubs routers under utilized
- Many units not energy proportional
- Turn off ports in proportion to b/w demands
- Timescales?
3 reasons:

1. Current energy inefficiencies
   a) Wired vs. wireless energy costs
   b) 6 TW-h ~ 1 nuclear reactor
   c) Extrapolate to World ~140 nuclear reactors

2. Enable greater deployment
   a) Similar connectivity in India would require 4.75% of total energy budget

3. Enable longer operation times during events of disaster
   a) Recent Grid failure in NE US/Canada
   b) Frequent power outages in most of the world
How to save energy?

- **Low-Energy Hardware Design:**
  - Use hardware components with low power modes of operation
  - Lower the clock frequency of the components, use DVS and other methods during low demand
  - Energy-Aware APIs - give control to software
  - Architecture that allows selective powering off

- **Energy-Aware Protocol Design:**
  - Node-level algorithms for sleeping
  - Route aggregation and other global techniques to inform devices when and for how long to sleep
Where to save energy in a device?

- Memory
- Main CPU
- Switch fabric or bus backplane
- Line cards (designs ranging from simple to complex with ASICs or network processors to process packets)
- Bang for the buck
Key Questions

- For how long can components sleep?
  - state transition times, energy spike

- How is the decision to sleep taken?
  - traffic activity level, in isolation (uncoordinated), global (coordinated), edge or backbone device, transit or stub network

- How to distinguish sleep vs. failure?
  - should not trigger network reconfiguration in sleep state vs. failure state

- How to wake up a device?
  - at fixed intervals, on packet arrival, account for protocol timers

- Impact on protocol behavior?
  - long sleep times, slower propagation of topology changes
More on sleeping....

**Uncoordinated sleeping:**
- Decision to sleep based on individual traffic levels alone
- Inform nearest neighbors only
- Sleep time limited by protocol hello message timer
- May trigger network reconfiguration in case of missed protocol messages

**Coordinated sleeping:**
- Decision taken on a network-wide basis
- Need algorithm to pre-compute the optimal sleep time, but computation costs increase
- Hello message frequency can be adjusted, may take longer to detect changes in network topology
- Sleep longer intervals, but forwarding tables may be outdated
- Reroute all traffic through one route, shutdown other routes
- Introduces delay and packet loss in case of sudden traffic burst
Computation Placement

- Energy cost varies by location
- Energy cost varies by hour of the day
- Can we push computation to a geographically distant place to save energy?
- Can we exploit time difference?
- Peak vs non-peak power rates
Exploit spatial-temporal price variation

- Prices vary from place to place
  - Coal vs nuclear
- Price varies with time of day
- Peak vs off-peak

<table>
<thead>
<tr>
<th>Location</th>
<th>RTO</th>
<th>Mean *</th>
<th>StDev *</th>
<th>Kurt.*</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chicago, IL</td>
<td>PJM</td>
<td>40.6</td>
<td>26.9</td>
<td>4.6</td>
</tr>
<tr>
<td>Indianapolis, IN</td>
<td>MISO</td>
<td>44.0</td>
<td>28.3</td>
<td>5.8</td>
</tr>
<tr>
<td>Palo Alto, CA</td>
<td>CAISO</td>
<td>54.0</td>
<td>34.2</td>
<td>11.9</td>
</tr>
<tr>
<td>Richmond, VA</td>
<td>PJM</td>
<td>57.8</td>
<td>39.2</td>
<td>6.6</td>
</tr>
<tr>
<td>Boston, MA</td>
<td>ISONE</td>
<td>66.5</td>
<td>25.8</td>
<td>5.7</td>
</tr>
<tr>
<td>New York, NY</td>
<td>NYISO</td>
<td>77.9</td>
<td>40.26</td>
<td>7.9</td>
</tr>
</tbody>
</table>
Modification of protocols at layers 2 and 3 to incorporate sleep modes
Study impact of modifications on end-to-end delay and performance
Develop energy models for routers and switches
Study the algorithmic problems of how/when/how long to sleep
Load migration: tolerable latency vs cost savings
The tail at scale

- Curse of the long tail
- 99% of the requests finish < 10 msec
- 1% of the request > 1 sec
- A large fanout, more requests will need > 1 sec
- Touch 100 servers: Prob not one of them is cursed = $0.99^{100} = 0.37$
- 63% of requests take > 1 sec
Variability in latency

- Shared resources
- Background jobs
- Queuing
- Maintenance jobs (checkpoint)
- Large Fanout: Multiplier effect

The tail at Scale by Dean and Barroso, CACM, Feb 2013, Vol. 56, NO.2
Use redundancy

- Redundancy effectively used to improve availability (RAID, replicated servers)
- Now use to reduce variability in latency
- Response time important in many services
  - Word completion, spellchecking, document selection
- Idea: **Send requests to multiple servers**
- Pro: Hit one server with low delay
- Con: more resources consumed due to duplicate work (increase in queueing delay)
- Challenge: reduce latency without undue overhead
Hedged requests

- Send requests to multiple servers
- Cancel the other after first response
- Techniques to reduce overhead:
  - Send second after some delay ($d > 95$ percentile)
  - Send second at a lower priority
Tied requests

- Send requests to multiple servers (with id of replicas)
- Server cancels other requests when the request scheduled
- Still both requests may be served (queue empty)
- Techniques to reduce overhead:
  - Send second after some delay (d > RTT of message in Network)
Hedged requests: 100 servers
- Send hedged request after 10 msec delay
- Reduced 99.9 percentile latency from 1800 msec to 74 msec
- Sends only 2% more requests

Tied requests

40% reduction at the 99.9 percentile latency

Key takeaway: Predictability from unpredictable parts

Can we do this for network security?
Computing cost declining: own or rent

Rutgers pays about 30K a month for ISPs
Cloud services

- Amazon Web services
- Google cloud
- Microsoft Azure

Infrastructure as a service  Platform as a service
Amazon cloud pricing

- **On-demand:** pay by the hour, no upfront payment (0.023 per hour to 0.094 per hour)
  - # CPUs, memory size, disk space etc varies
- **Spot pricing**
  - Bid for spare capacity, bid price < spot price, process is preempted
- **Reserved Pricing**
  - 1 yr to 3 yr commitment; upfront payment
  - Large instance $549 plus 0.063 per hour
Cost of cloud

- Many cost models for compute resources
- Storage (Amazon S3)
  - Lease: 1 TB/Month 3$ regular; 1.25$ infrequent; Glacier (archive) 0.70$
  - Buy: 1 TB 60$
- Cost models for Database, management, storage, network, computation
- Data mining/analytics (rapidminer)
  - Price proportional to number of tuples used
New Metrics

- **Cost $$**
- **Cost of Computing**
  - Trying to figure how much it is going to cost
  - Bidding for cloud services
  - Computation (AWS, AZURE), B/W, Storage (box, dropbox)
  - How to bid the cloud [Liang, Zheng et.al. SIGCOMM 2014]
  - Need protocols that reduces burden on end-user

Sliding Scale

- Buy
- Rent

Static, Dynamic Auction Pricing
Bid strategies

- One time bids
  - Present once, exit if it falls below current market price
  - Risk: job may not complete,

- Persistent bids
  - Resubmit each time period until job finishes

- Hybrid
  - Important nodes one time bid and slaves persistent bids (e.g., map-reduce)

- Requires transparent provider pricing
Cost continuum tools

- Can we build tools that can answer questions like
  - What is the cost for doing X?
  - What should I own, rent, or bid?
  - How should I distribute my computation for a given cost model?
  - What is the performance impact?
    - Cost vs accuracy, cost vs latency, cost vs availability
  - Replace CPU with any other resource: storage, DB, B/W, management etc
  - Edge vs Cloud -- “cache” to save dollars
Current Network

Routing, management, mobility management, access control, VPNs, ...

- Million of lines of source code
- 5400 RFCs: Barrier to entry
- Billions of gates
- Bloated: Power Hungry

Many complex functions baked into the infrastructure:

- OSPF, BGP, multicast, differentiated services,
- Traffic Engineering, NAT, firewalls, MPLS, redundant layers, ...

An industry with a “mainframe-mentality”, reluctant to change

Slides from Stanford Site: Nick Mckeown, Martin Casado, Scott Shenker et al,
The SDN Approach

Separate control from the datapath
  ● i.e. separate policy from mechanism

Datapath: Define minimal network instruction set
  ● A set of “plumbing primitives”
  ● A vendor-agnostic interface: e.g., OpenFlow

Control: Define a network-wide OS
  ● An API that others can develop on
SDN approach

Conventional

- Application
- Network OS
- ASIC

- Application
- Network OS
- ASIC

- Application
- Network OS
- ASIC

SDN

- Application
  - North bound API
  - South bound API
  - Openflow, POF, OpenSwitch

- Controller
  - Switch OS
  - Switch H/W
At each table, action set is constructed; at the end action set is executed
Flow Table Entry
“Type 0” OpenFlow Switch

Rule | Action | Stats
--- | --- | ---

Packet + byte counters

1. Forward packet to port(s)
2. Encapsulate and forward to controller
3. Drop packet
4. Send to normal processing pipeline

Switch Port | MAC src | MAC dst | Eth type | VLAN ID | IP Src | IP Dst | IP Prot | TCP sport | TCP dport
--- | --- | --- | --- | --- | --- | --- | --- | --- | ---

+ mask

Source: Open flow switch specification-Open Network Foundation
# Examples

## Switching

<table>
<thead>
<tr>
<th>Switch Port</th>
<th>MAC src</th>
<th>MAC dst</th>
<th>Eth type</th>
<th>VLAN ID</th>
<th>IP Src</th>
<th>IP Dst</th>
<th>IP Prot</th>
<th>TCP sport</th>
<th>TCP dport</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>*</td>
<td>*</td>
<td>00:1f:..</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>port6</td>
</tr>
</tbody>
</table>

## Flow Switching

<table>
<thead>
<tr>
<th>Switch Port</th>
<th>MAC src</th>
<th>MAC dst</th>
<th>Eth type</th>
<th>VLAN ID</th>
<th>IP Src</th>
<th>IP Dst</th>
<th>IP Prot</th>
<th>TCP sport</th>
<th>TCP dport</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>port3</td>
<td>00:20..</td>
<td>00:1f..</td>
<td>0800</td>
<td>vlan1</td>
<td>1.2.3.4</td>
<td>5.6.7.8</td>
<td>4</td>
<td>17264</td>
<td>80</td>
<td>port6</td>
</tr>
</tbody>
</table>

## Firewall

<table>
<thead>
<tr>
<th>Switch Port</th>
<th>MAC src</th>
<th>MAC dst</th>
<th>Eth type</th>
<th>VLAN ID</th>
<th>IP Src</th>
<th>IP Dst</th>
<th>IP Prot</th>
<th>TCP sport</th>
<th>TCP dport</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>22</td>
</tr>
</tbody>
</table>
### Examples

#### Routing

<table>
<thead>
<tr>
<th>Switch Port</th>
<th>MAC src</th>
<th>MAC dst</th>
<th>Eth type</th>
<th>VLAN ID</th>
<th>IP Src</th>
<th>IP Dst</th>
<th>IP Prot</th>
<th>TCP sport</th>
<th>TCP dport</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>5.6.7.8</td>
<td>*</td>
<td>port6</td>
</tr>
</tbody>
</table>

#### VLAN Switching

<table>
<thead>
<tr>
<th>Switch Port</th>
<th>MAC src</th>
<th>MAC dst</th>
<th>Eth type</th>
<th>VLAN ID</th>
<th>IP Src</th>
<th>IP Dst</th>
<th>IP Prot</th>
<th>TCP sport</th>
<th>TCP dport</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>*</td>
<td>*</td>
<td>00:1f..</td>
<td>*</td>
<td>vlan1</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>port6, port7, port9</td>
</tr>
</tbody>
</table>
SDN use cases

- Global Traffic Engineering
  - B4, SWAN

- Network function Virtualization
  - Allow multiple functionalities implemented on Hardware
  - Made programmable by SDN
Boxes, boxes everywhere

Load balancer

Firewall

Wireless Router

Layer 2 Switch

NAT box

Controller

Composed flow tables
SDN issues

- Correctness, Consistency, Configuration
- Policy to flow rules
- Controller robustness
  - Security, availability
- Management Plane
Research

- Learning in networks
  - Network functions
  - Edge learning
  - Smart control plane
  - Deep learning
- Querying vs lookup
  - Bots for Network management
- Network defined Economy
  - Shared resources
- Smart cities