INFM 603: Information Technology and Organizational Context

Session 2: HTML and CSS

(And Computing Tradeoffs, Networking)

Jimmy Lin
The iSchool
University of Maryland

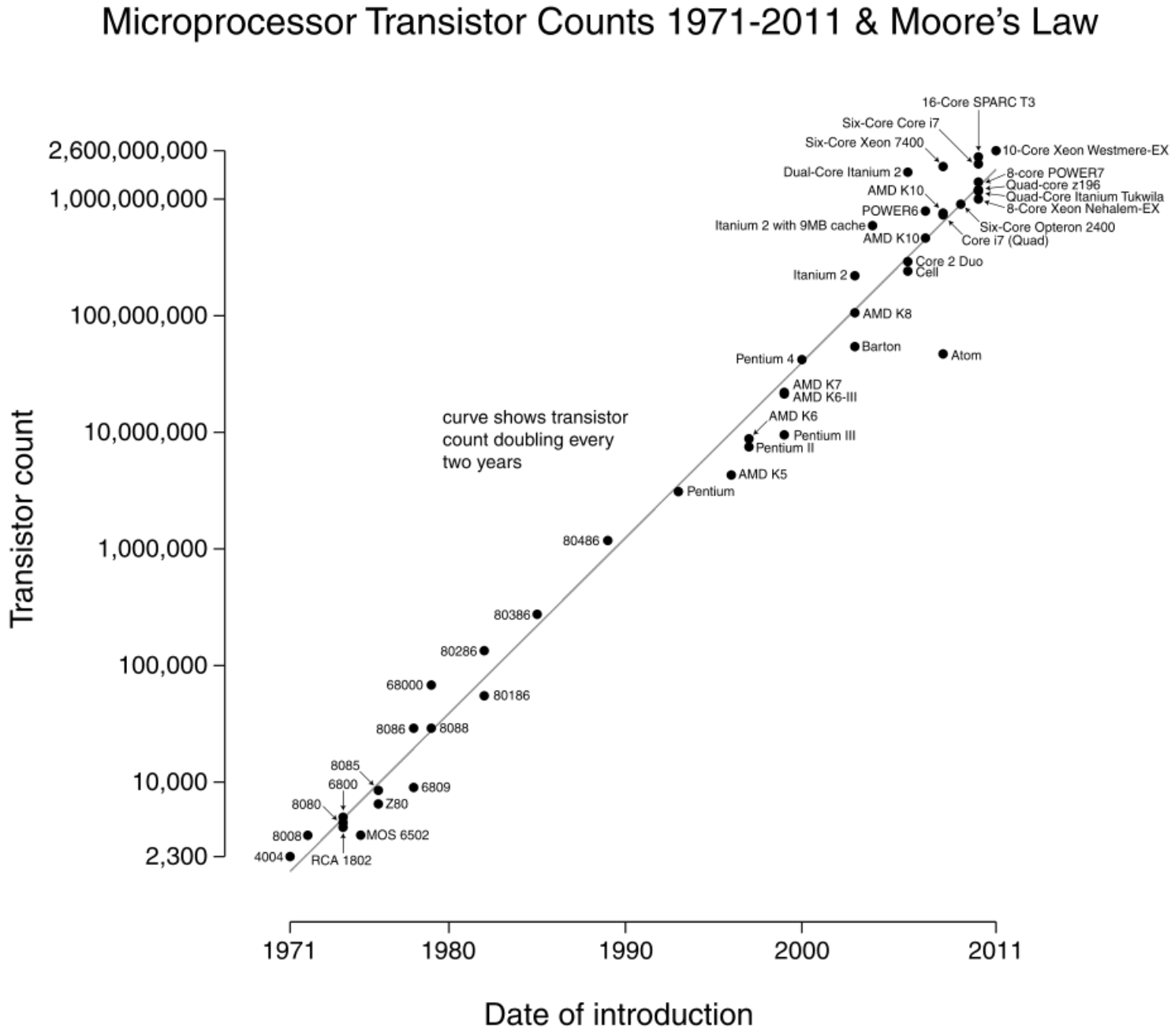
Thursday, September 12, 2013

Ways to characterize computing

- How big?
- How fast?
- How reliable?

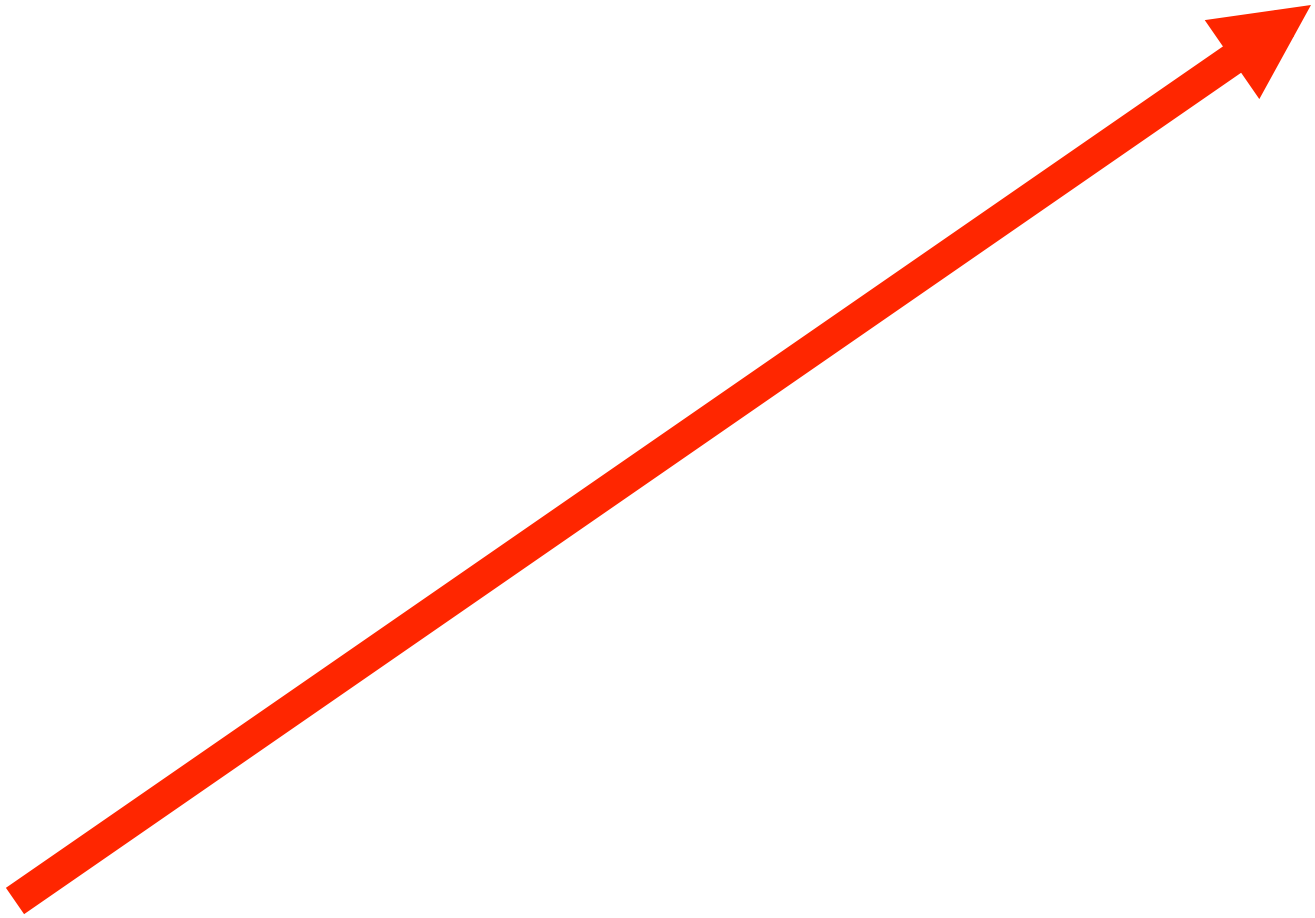
Computing is fundamentally about tradeoffs!

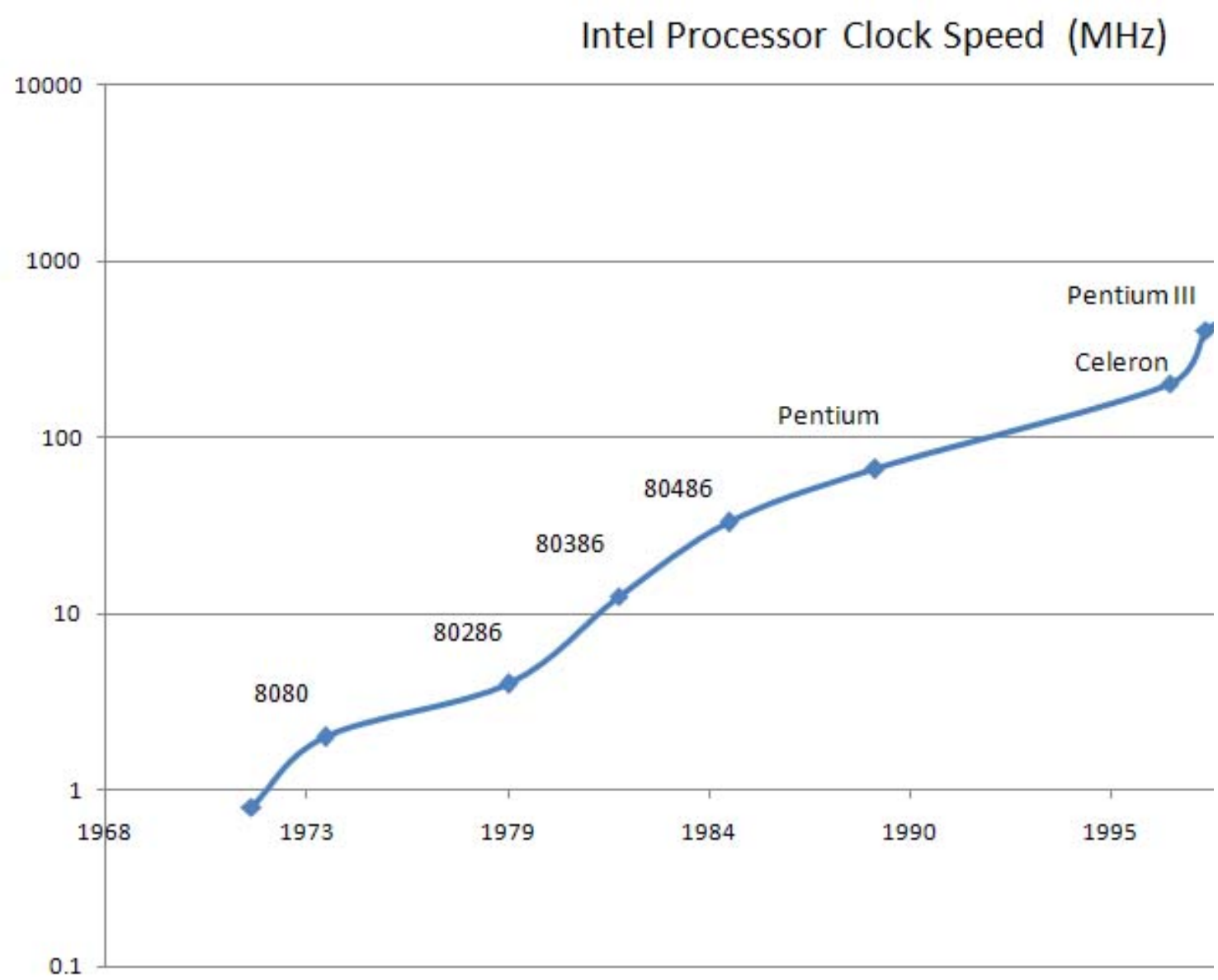
Example 1: Multi-Core



Source: Wikipedia (Moore's Law)

Trends in Computing: #1



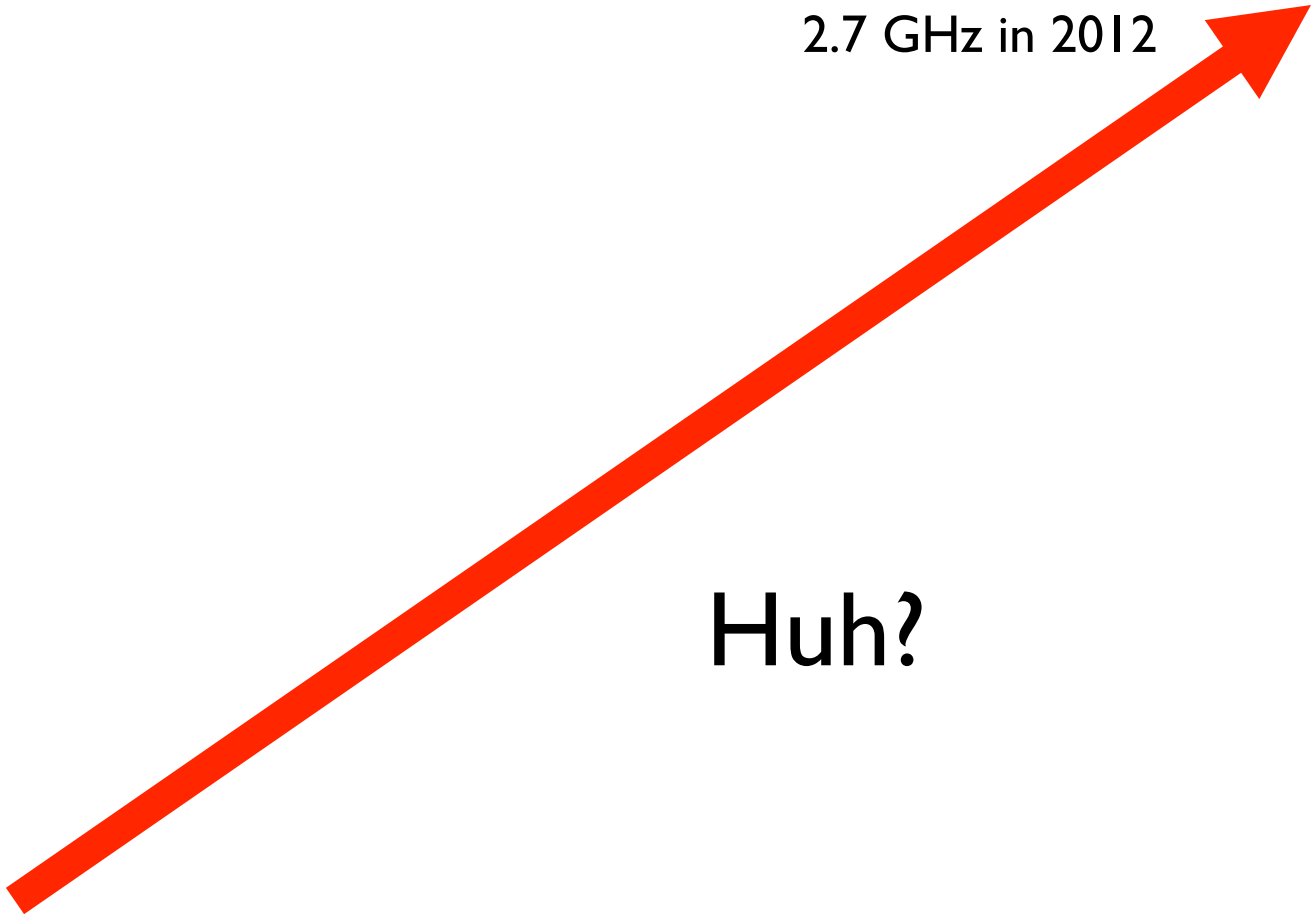


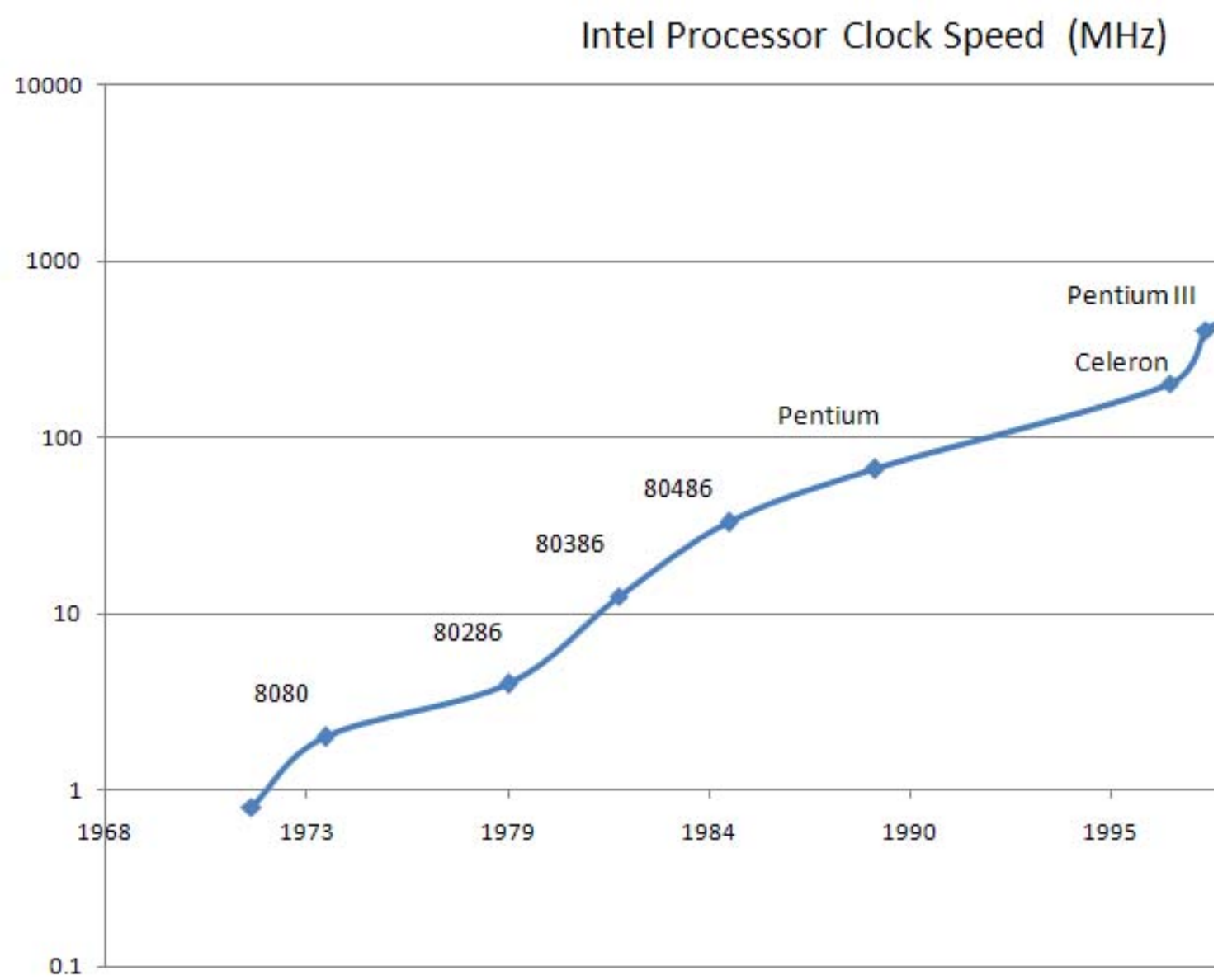
Source: smoothspan.wordpress.com


About This Mac
OS X
Version 10.8.1
Software Update...
Processor 2.7 GHz Intel Core i7
Memory 16 GB 1600 MHz DDR3
More Info...
TM and © 1983–2012 Apple Inc.
All Rights Reserved. License Agreement


INTEL® © '03
PENTIUM® 4
3.40GHZ/1M/800
SL7J8 COSTA RICA
3429A551
3.00GHZ/1M/800
SL8BM COSTA RICA
3429A551T

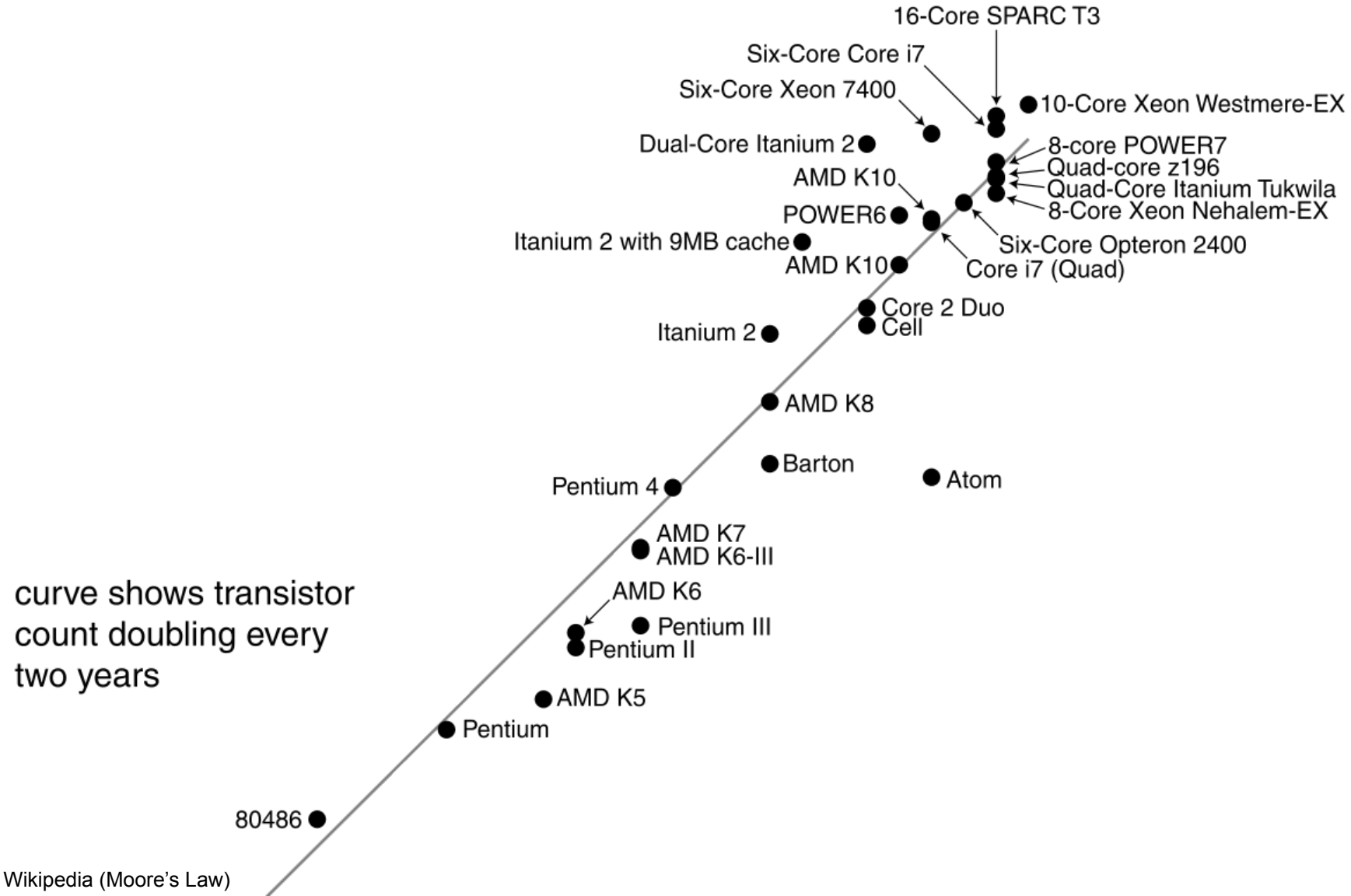
Trends in Computing: #1





Source: smoothspan.wordpress.com

ransistor Counts 1971-2011 & Moore's Law



Source: Wikipedia (Moore's Law)

What's big shift?

- From single to multiple cores:
  - Increasing speed of single processor reached point of diminishing returns
  - Solution: put more cores on a processor!
- Important issues:
  - Power
  - Cool
  - Parallelism

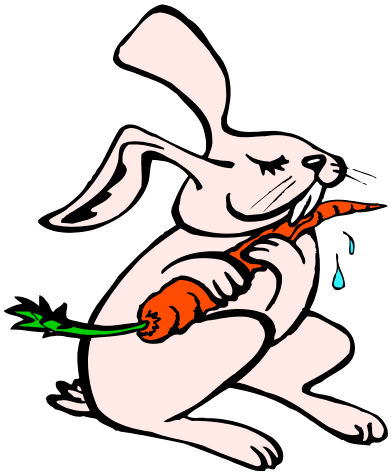
Example 2: Caching

Source: Wikipedia

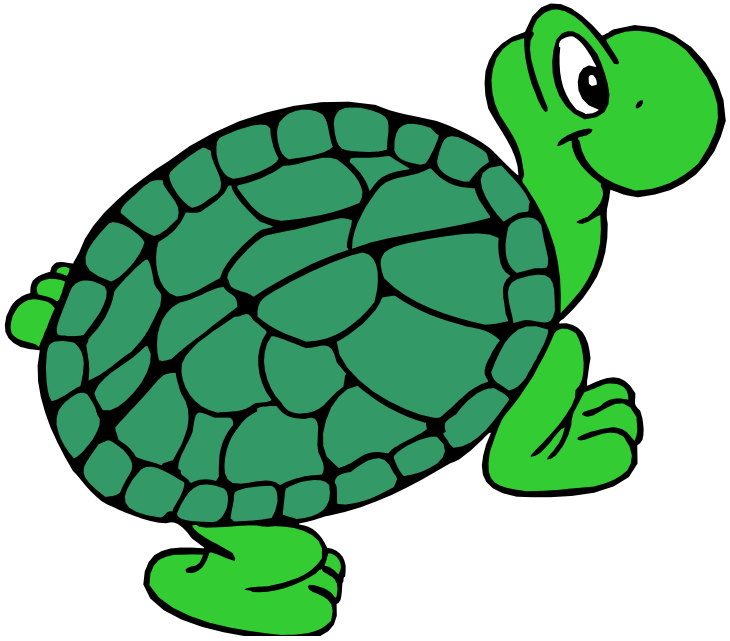
Source: Wikipedia

Pick two

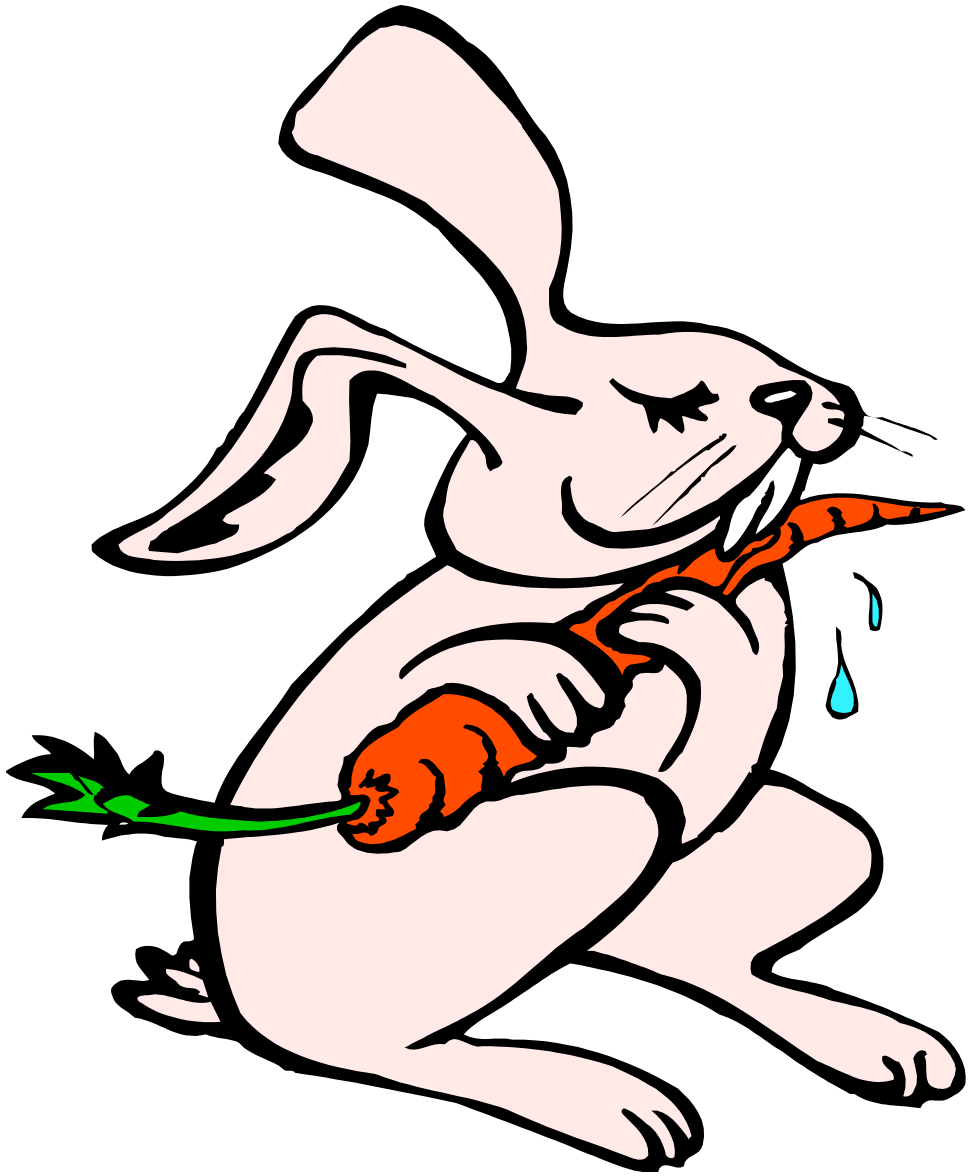
- Speed
- Capacity
- Cost



RAM: small, expensive, fast



Hard drives: big, cheap, slow



Best of both worlds? cheap, fast, and big

Caching

- **Idea:** move data you're going to use from slow memory into fast memory
  - Slow memory is cheap so you can buy lots of it
  - Caching gives you the illusion of having lots of fast memory
- Physical analogy?
- How do we know what data to cache?
  - Spatial locality: If the system fetched *x*, it is likely to fetch data located near *x* (Why?)
  - Temporal locality: If the system fetched *x*, it is likely to fetch *x* again (Why?)

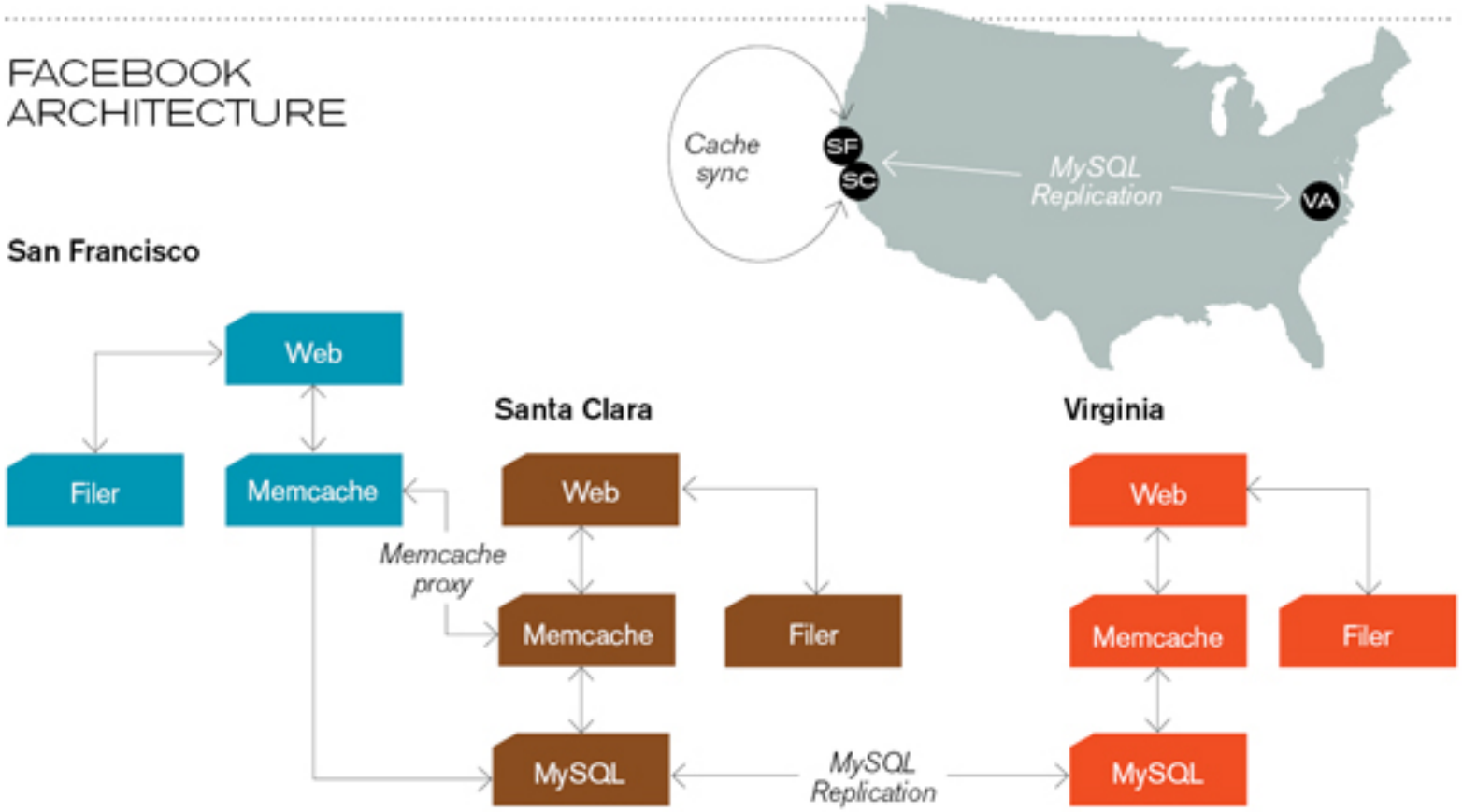
Example 3: Replication

Characterizing Reliability

"Nines"	Availability	Downtime (per year)
One nine	90%	36.5 d
Two nines	99%	3.65 d
Three nines	99.9%	8.76 h
Four nines	99.99%	52.56 m
Five nines	99.999%	5.256 m
Six nines	99.9999%	31.536 s

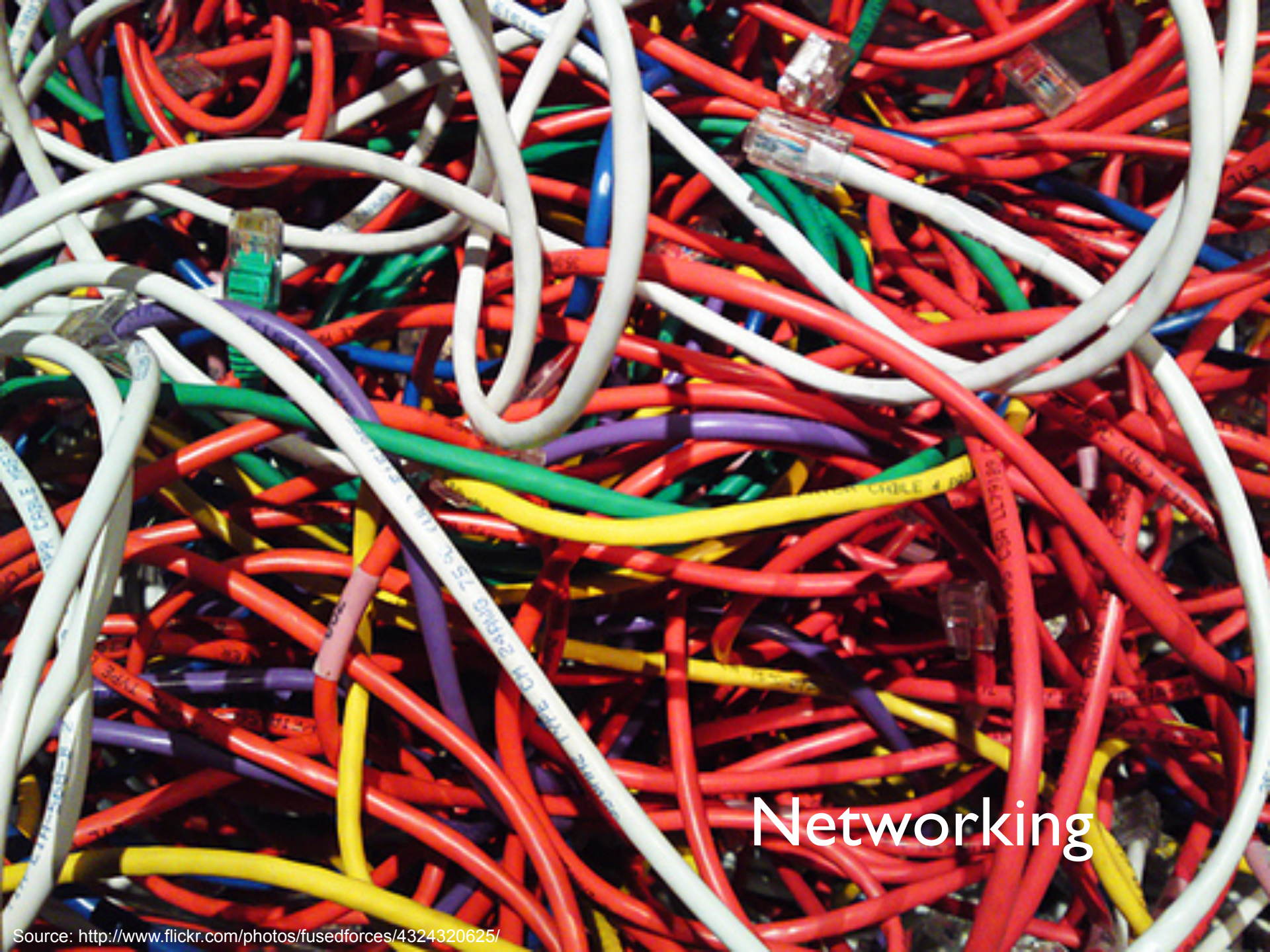
How do you ensure reliability?

- Keep multiple copies:
  - On different machines
  - On different machines far apart
- What are the challenges with this?
  - Synchronous vs. Asynchronous
  - Active-Active vs. Active-Passive
  - …



Facebook architecture
(circa 2008)

Source: Technology Review

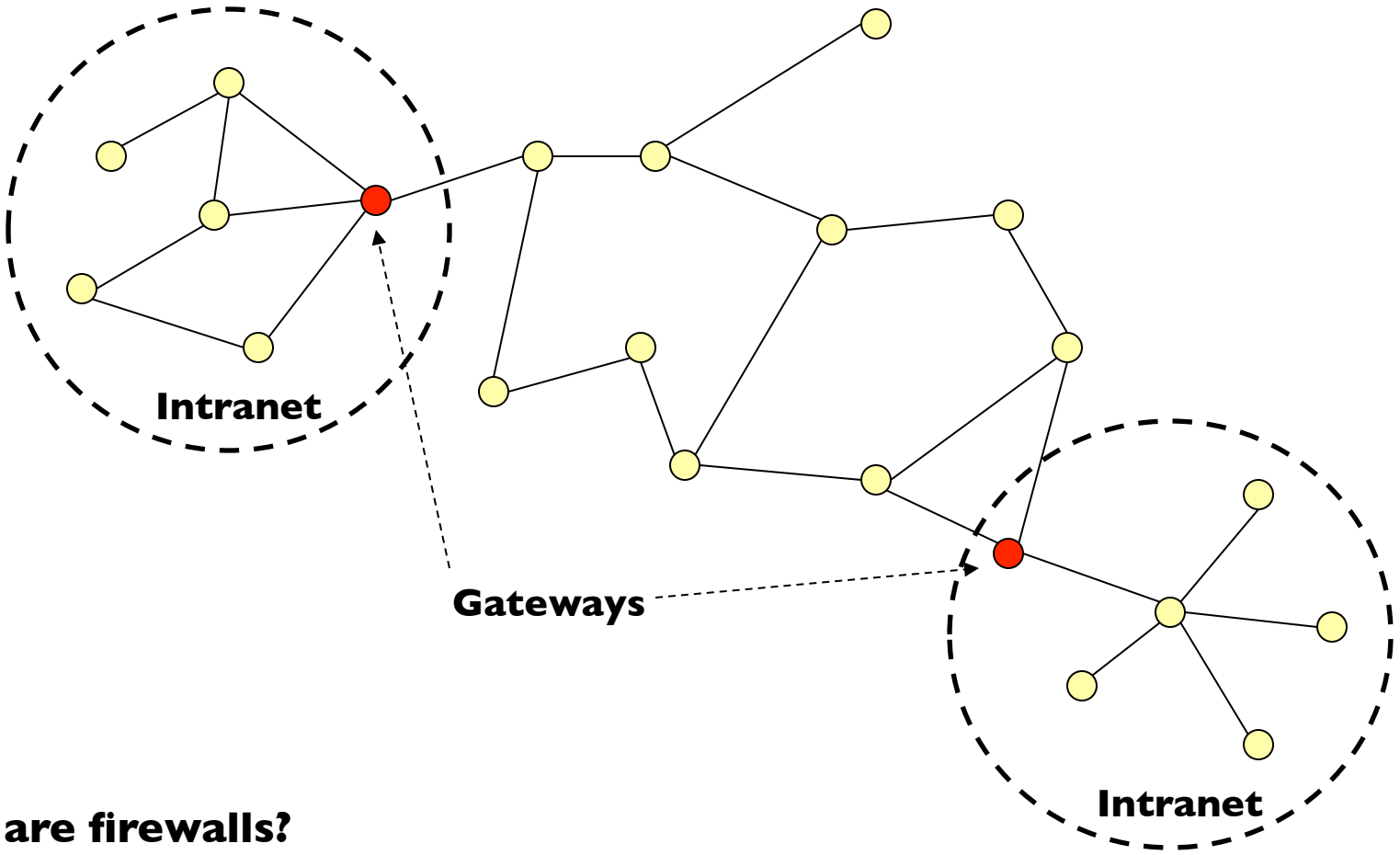


Source: http://www.flickr.com/photos/fusedforces/4324320625/

Internet ≠ Web

- Internet = collection of global networks
- Web = particular way of accessing information on the Internet
  - Uses the HTTP protocol
- Other ways of using the Internet
  - Usenet
  - FTP
  - email (SMTP, POP, IMAP, etc.)
  - Internet Relay Chat

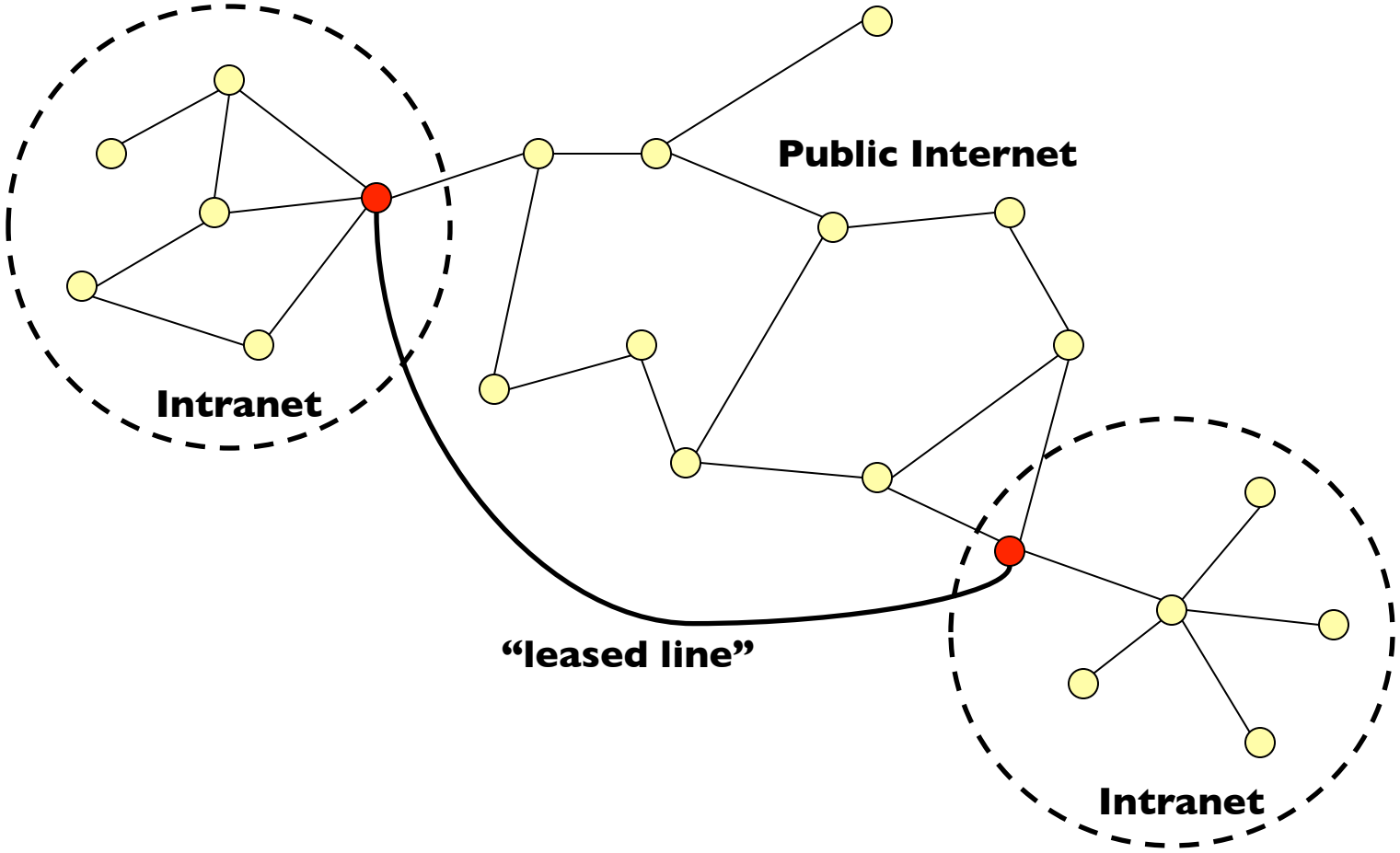
Intranets



What are firewalls?
Why can't you do certain things behind firewalls?

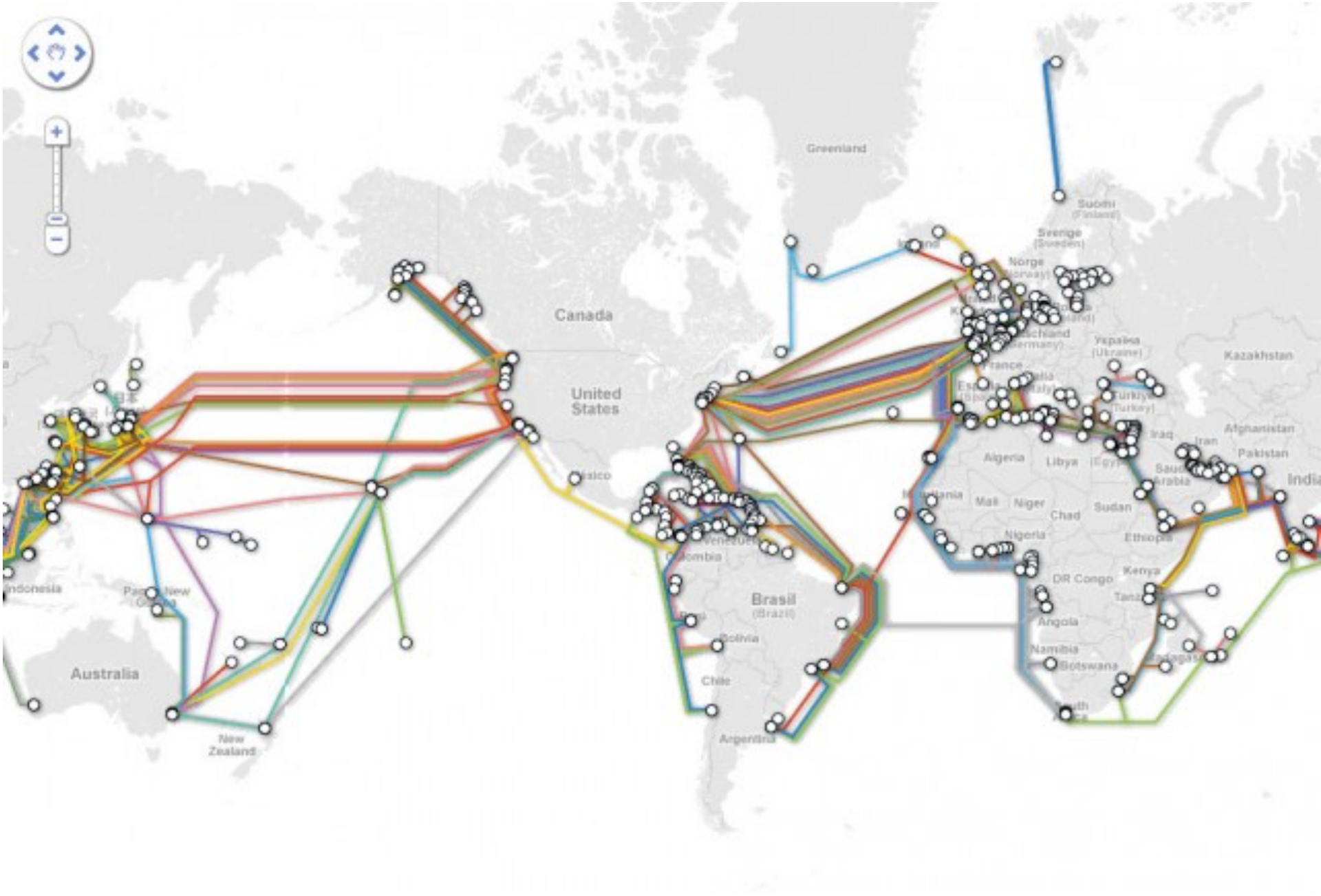
Intranets

Problem: How do you securely connect separate networks?



VPN = Virtual Private Network

a secure private network over the public Internet



Source: http://www.extremetech.com/computing/96827-the-secret-world-of-submarine-cables

Foundations

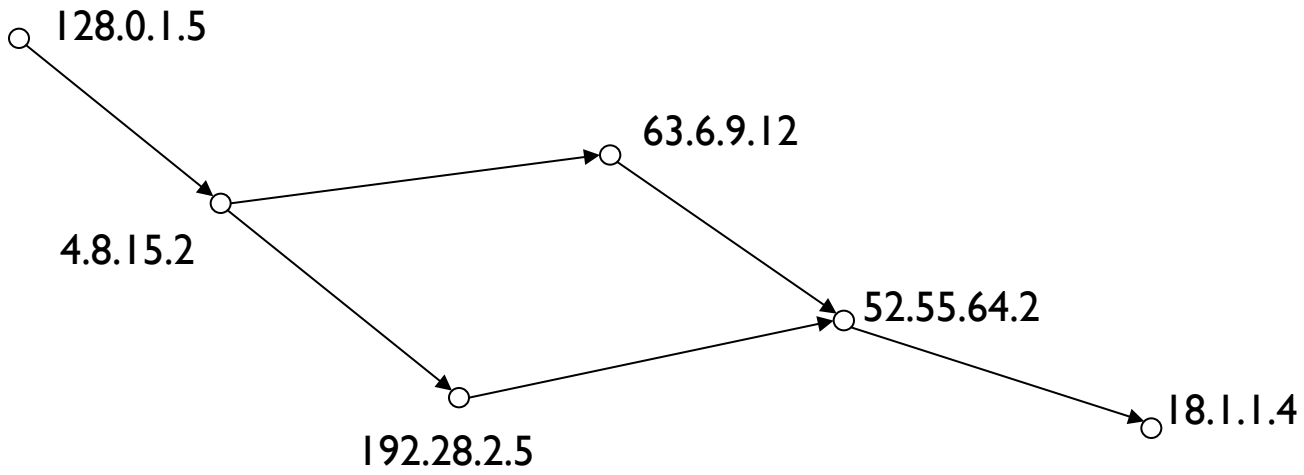
- Basic protocols for the Internet:
  - TCP/IP (Transmission Control Protocol/Internet Protocol):
    basis for communication
  - DNS (Domain Name Service):
    basis for naming computers on the network
- Protocol for the Web:
  - HTTP (HyperText Transfer Protocol):
    protocol for transferring Web pages

IP Address

- Every computer on the Internet is identified by a address
- IP address = 32 bit number, divided into four "octets"
  - Example: go in your browser and type "http://74.125.131.147/"

Are there enough IP addresses to go around?
What is the difference between static and dynamic IP?

Packet Routing (TCP/IP)



(Much simplified) Routing table for 4.8.15.2

Destination	Next Hop
52.55.*.*	63.6.9.12
18.1.*.*	192.28.2.5/63.6.9.12
4.*.*.*	225.2.55.1
…	

Domain Name Service (DNS)

- Domain names improve usability
  - Easier to remember than IP addresses
  - DNS provides a lookup service
- Each name server knows one level of names
  - "Top level" name server knows .edu, .com, .mil, …
  - .edu name server knows umd, mit, stanford, …
  - .umd.edu name server knows ischool, wam, …

Demo

- Play with various utilities at
  - http://network-tools.com/
  - http://www.yougetsignal.com/tools/visual-tracert/
  - http://en.dnstools.ch/visual-traceroute.html

HyperText Transfer Protocol

- Send request

```
GET /path/file.html HTTP/1.0
From: someuser@somedomain.com
User-Agent: HTTPTool/1.0
```

- Server response

```
HTTP/1.0 200 OK
Date: Fri, 31 Dec 1999 23:59:59 GMT
Content-Type: text/html
Content-Length: 1354
<html><body> <h1>Happy New Millennium!</h1> … </body> </html>
```

Tell me what happens...

- From the moment you click on “check messages” to the moment you start reading your email
- From the moment you click “send” to the moment the other party receives the email
- From the moment you type a URL and hit “enter” to the moment you see the Web page

Tables

Source: Wikipedia (Table)

Tables

<table>
<tr><td> eenie </td><td> mennie </td><td> miney </td></tr>
<tr><td> mo </td><td> catch </td><td> a tiger </td></tr>
<tr><td> by </td><td> the </td><td> toe </td></tr>
</table>

Source: Wikipedia (Waterfall)

What's a Document?

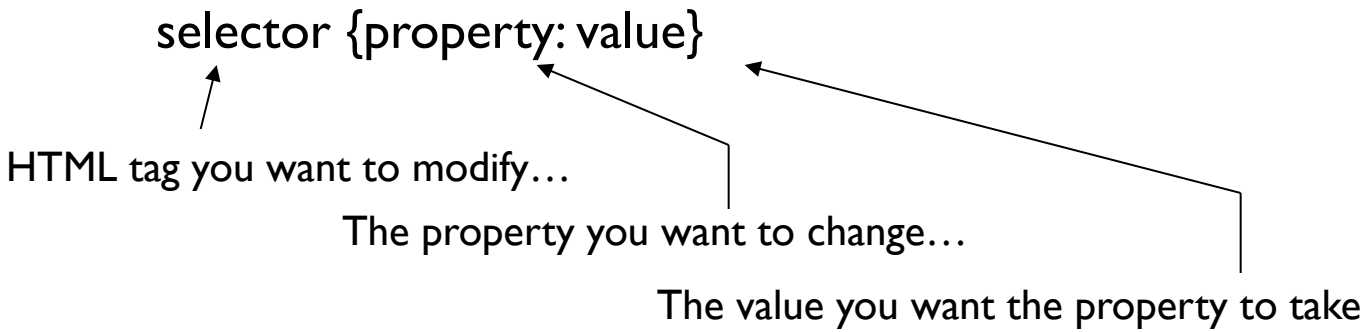
- Content
- Structure
- Appearance
- Behavior

CSS: Cascading Style Sheets

- Separating content and structure from appearance
- Rules for defining styles "cascade" from broad to narrow:
  - Browser default
  - External style sheet
  - Internal style sheet
  - Inline style

Basics of CSS

- Basic syntax:



- Example:

```
p { text-align: center;
    color: black;
    font-family: arial }
```

Causes

- Font to be center-aligned
- Font to be Arial and black

Different Ways for Using CSS

- Inline style:
  - Causes only the tag to have the desired properties

```
<p style="font-family:arial; color:blue">…</p>
```

- Internal stylesheet:
  - Causes all tags to have the desired properties

```
…
<head>…
<style type="text/css">
p { font-family:arial; color:blue}
</style>
</head>
<body>
<p>…</p>
…
```

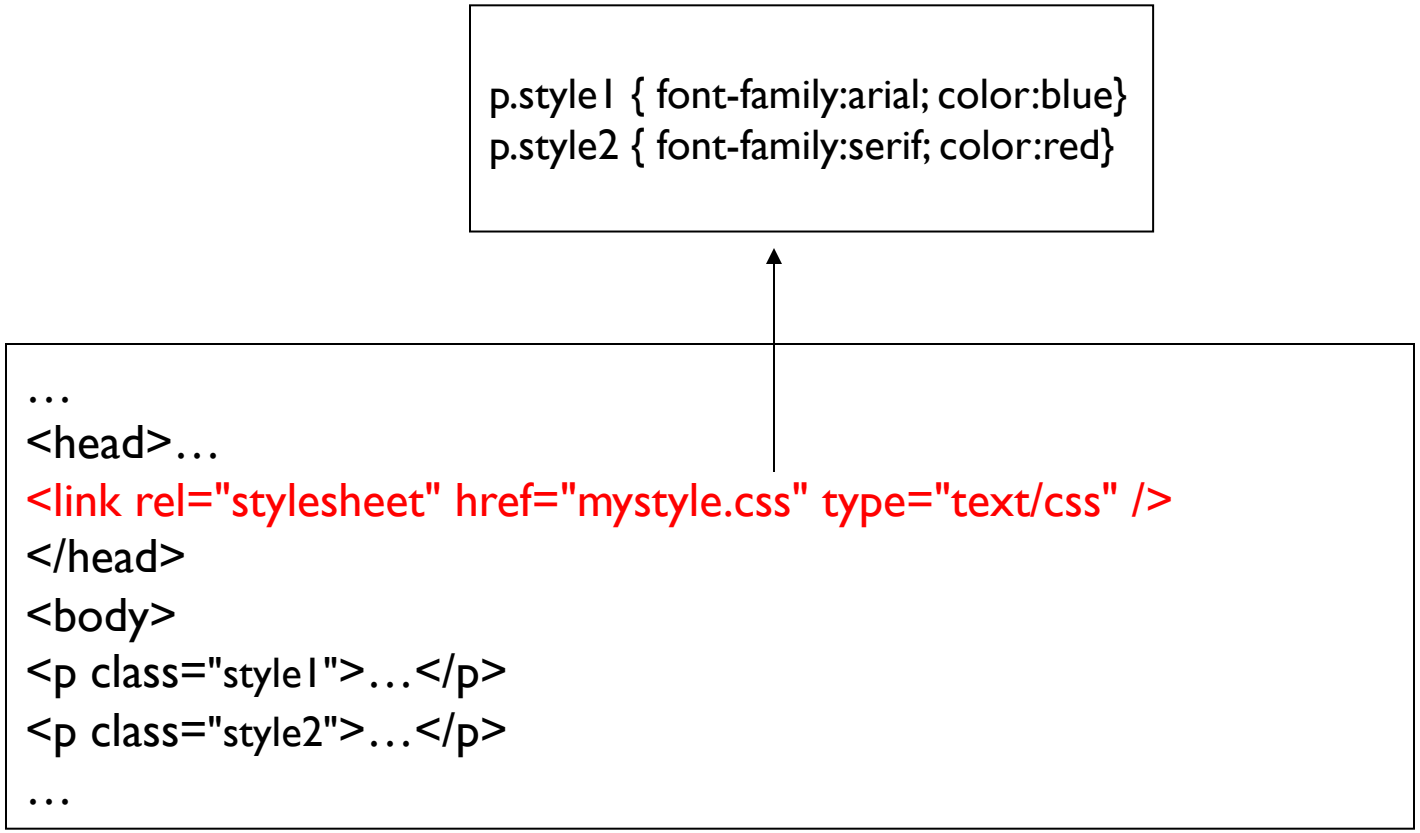
Customizing Classes

- Define customized styles for standard HTML tags:

```
…
<head>…
<style type="text/css">
p.style1 { font-family:arial; color:blue}
p.style2 { font-family:serif; color:red}
</style>
</head>
<body>
<p class="style1">…</p>
<p class="style2">…</p>
…
```

External Style Sheets

- Store formatting metadata in a separate file



Why Use CSS?

- What are the advantages of CSS?
- Why have three separate ways of using styles?