6.829
Computer Networks

Lecture 1:
Introduction & Course Overview

Mohammad Alizadeh

Fall 2018

The Internet: An Exciting Time

One of the most influential inventions

- A research experiment that escaped from the lab
- … to be the global communications infrastructure

Ever wider reach

- Today: 3 billion users
- Tomorrow: more users, computers, sensors, "things", … 40 billion devices by 2020

Constant innovation

- Apps: Web, P2P, social networks, virtual worlds …
- Links: Optics, WiFi, cellular, ...

◇ Slides 1-12 based on lectures by Nick McKeown (Stanford), Jennifer Rexford (Princeton)

Transforming Everything

The ways we do business

- E-commerce, advertising, cloud computing, ...

The way we have relationships

- E-mail, IM, Facebook, Instagram, virtual worlds

How we think about law

- Interstate commerce? National boundaries? Wikileaks?

The way we govern

- E-voting
- Censorship and wiretapping

The way we fight

- Cyber-attacks, nation-state attacks, fake news

But what *is* networking?

A Plethora of Protocol Acronyms?

SNMP WAP SIP PPP IPX MAC

LLDP FTP UDP

OSPF RTP ICMP IMAP IGMP HIP

PIM BGP IP RED ARP ECN

RIP HTTP MPLS TCP RTCP

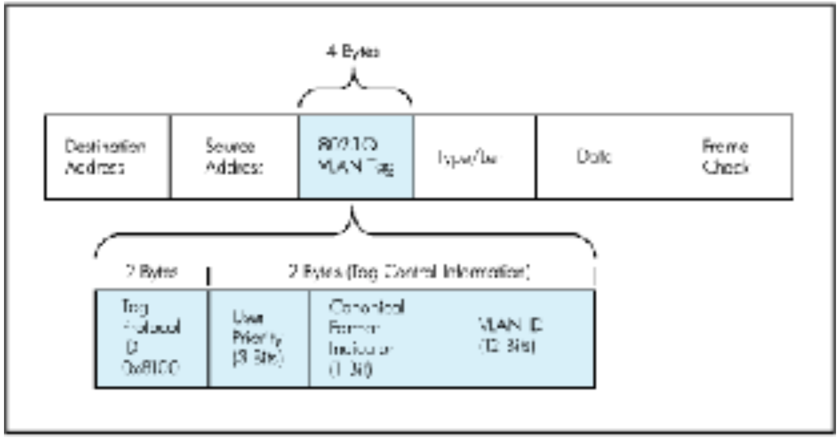
SMTP RTSP BFD CIDR

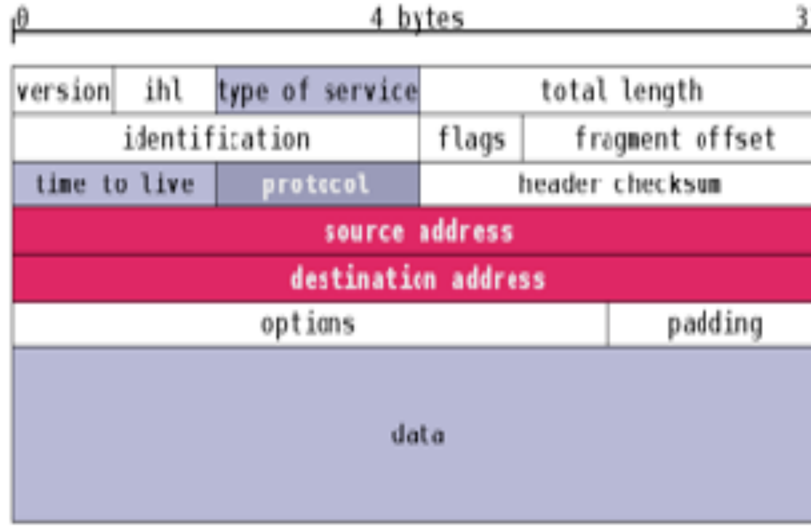
NNTP SACK TLS NAT STUN

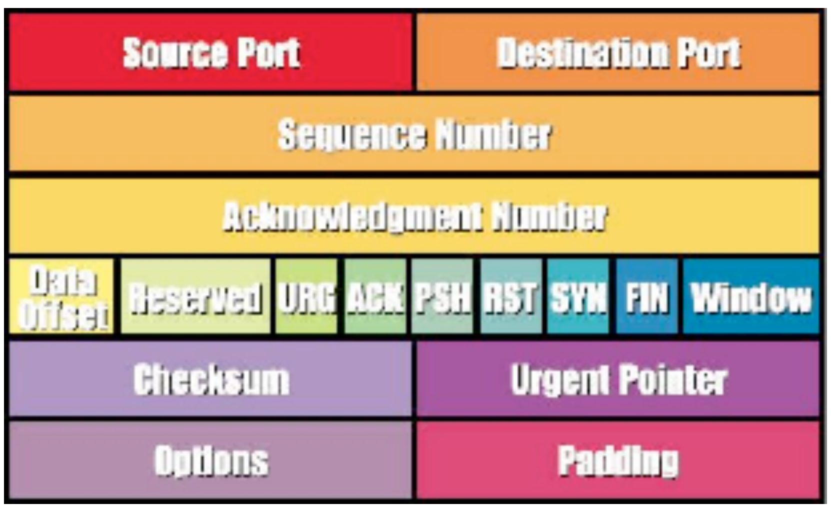
DNS SSH DHCP

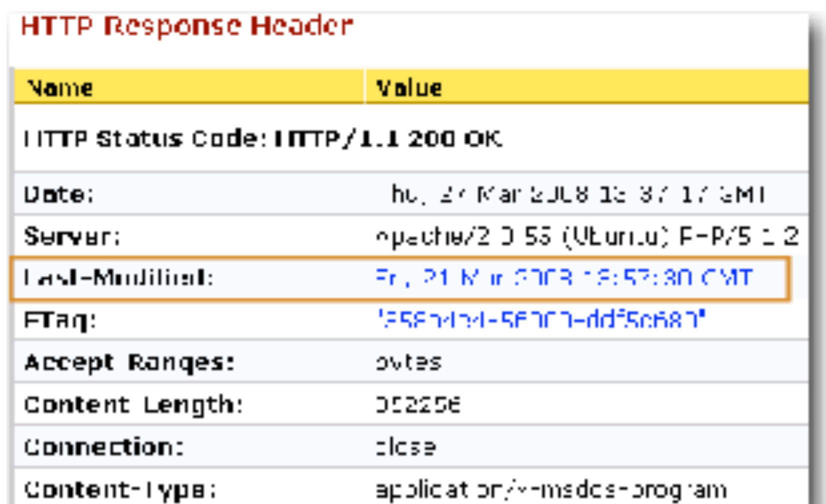
POP VLAN LISP VTP TFTP LDP

A Heap of Header Formats?

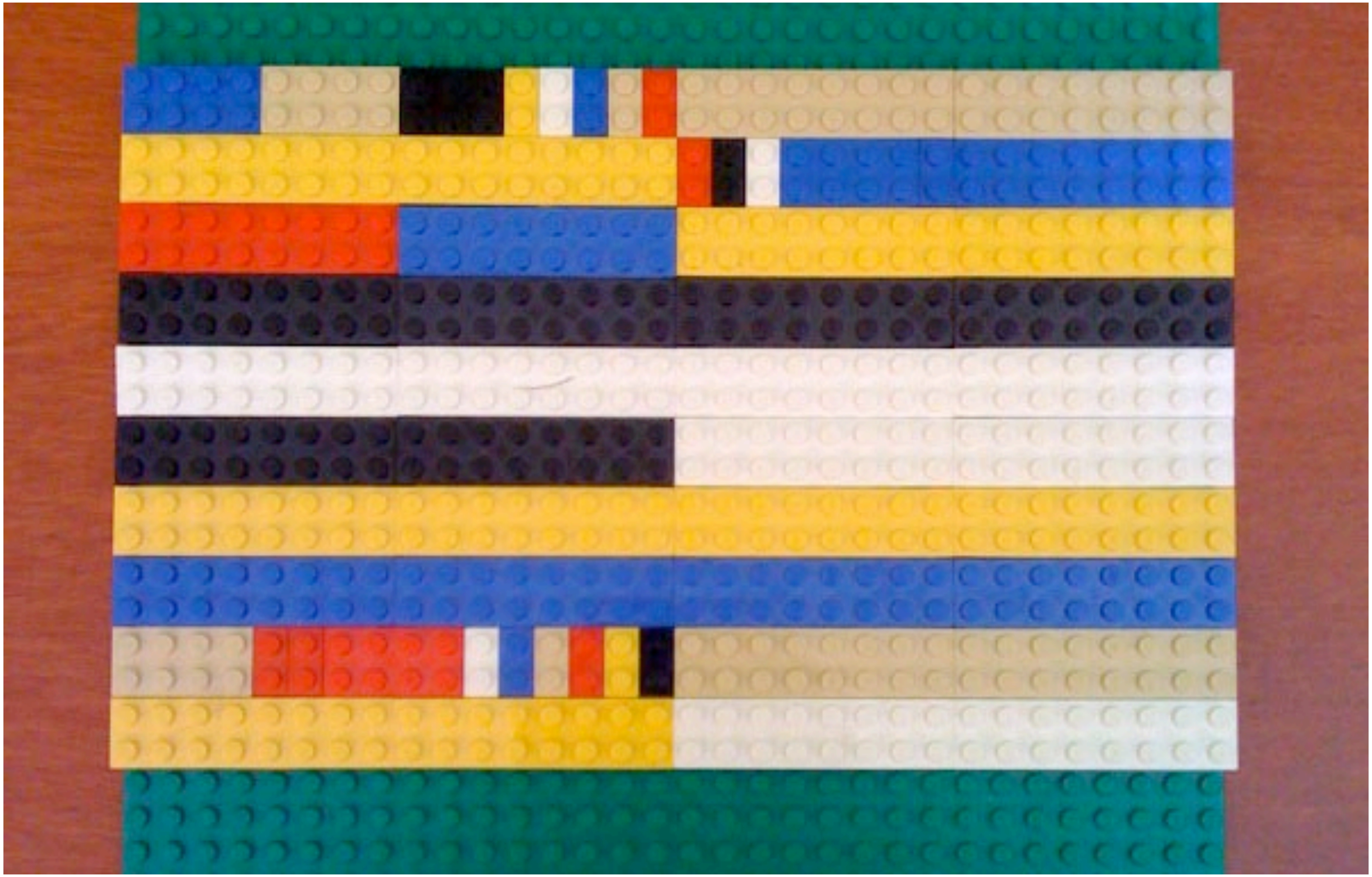








TCP/IP Header Formats in Lego



A Big Bunch of Boxes?

Router

Label Switched Router

Load balancer

Switch

Scrubber

Repeater

Gateway

Intrusion Detection System

Bridge

Route Reflector

Deep Packet Inspection

DHCP server

Packet shaper

Firewall

NAT

Hub

Packet sniffer

WAN accelerator

DNS server

Base station

Proxy

An Application Domain?

A place to apply theory?

- Algorithms and data structures
- Graph theory
- Control theory
- Queuing theory
- Optimization theory
- Game theory and mechanism design
- Machine learning & AI
- Cryptography
- Programming languages
- Formal methods

A place to build systems?

- Distributed systems
- Operating systems
- Computer architecture
- Software engineering

…

What Peers in Other Fields Say

“What are the top ten classic problems in networking? I would like to solve one of them and submit a paper to SIGCOMM.”

After hearing that we don't have such a list: "Then how do you consider networking a discipline?”

“So, these networking research people today aren't doing theory, and yet they aren't the people who brought us the Internet. What exactly are they doing?”

“Networking papers are strange. They have a lot of text.”

Is networking a problem domain or a scholarly discipline?

Before you all leave …

So, Why is Networking Cool?

Relevant

- Can impact the real world
- Can measure/build things

Interdisciplinary

- Well-motivated problems + rigorous solution techniques
- Interplay with policy and economics

Widely-read papers

- Many of the most cited papers in CS are in networking
- Congestion control, distributed hash tables, resource reservation, self-similar traffic, multimedia protocols

So, Why is Networking Cool?

Young, relatively immature field

- Tremendous intellectual progress is still needed
- *You* can help decide what networking really is

Defining the problem is a big part of the challenge

- Recognizing a need, formulating well-defined problem
- … is at least as important as solving the problem.

Lots of platforms for building your ideas

- Testbeds: Emulab, PlanetLab, Orbit, GENI
- Programmability: Click, Mininet, NetFPGA
- Measurements: traceroute, RouteViews, Internet2

About this class

Goals

1. To understand a good slice of the state-of-the-art in network architecture, protocols, and systems.

2. To learn how to conduct networking research and develop innovative ideas.

Course Structure

Lectures & Readings

Three Problem Sets

One Quiz (Nov. 15, in class)

Final Project

Lecture & Readings

Lecture

- Each class we will discuss 1-2 papers
- You <u>must</u> read the papers before class
- Most of lecture will be spent discussing/debating the papers
- Come prepared to discuss the main ideas!

Questions about readings

- Questions will be posted on class webpage per lecture
- Submit answers online **by midnight** the night before lecture (see class webpage)
- To pass the class, you must submit answers **for at least 18 lectures**

Syllabus Summary

Internet Architecture & Protocols (8 classes)

- Key principles & protocols: congestion control, routing, mobility, measurement
- Classic research papers

“Underlay” Networks (8 classes)

- The infrastructure underlying the modern Internets
- Wireless & datacenter networks, optical networks, software-defined networks, programmable routers

“Overlay” Networks (5 classes)

- Applications that run on top of the Internet
- Video, peer-to-peer, content distribution, blockchain

Miscellaneous (2-3 classes)

- Guest lectures, project presentations, …

Logistics

Who will lead the discussions

- Hari Balakrishnan (hari@csail.mit.edu)
- Mohammad Alizadeh (alizadeh@csail.mit.edu)
- Guest experts (from Microsoft, Google, Fastly, Akamai, …)

TA

- Akshay Narayan (akshayn@mit.edu)

Class webpage

- http://web.mit.edu/6.829/www/currentsemester/

Piazza

- Sign up here: https://piazza.com/mit/fall2018/6829/home
- Quickest response; someone else probably has same question
- If private, please post a private Piazza post

Rest of Today

A sampler of class topics:

1. Internet Architecture & Protocols: Congestion control; Routing

2. Underlay networks: Cellular, Datacenters, and Programmable networks

3. Overlay networks: Streaming Video

Internet Architecture & Protocols

Internet Architecture

Architectural questions tend to dominate networking research

Definition and placement of function

- What to do, and where to do it

The "division of labor"

- Across components (hosts, routers, and management systems)
- Across multiple concurrent protocols and mechanisms

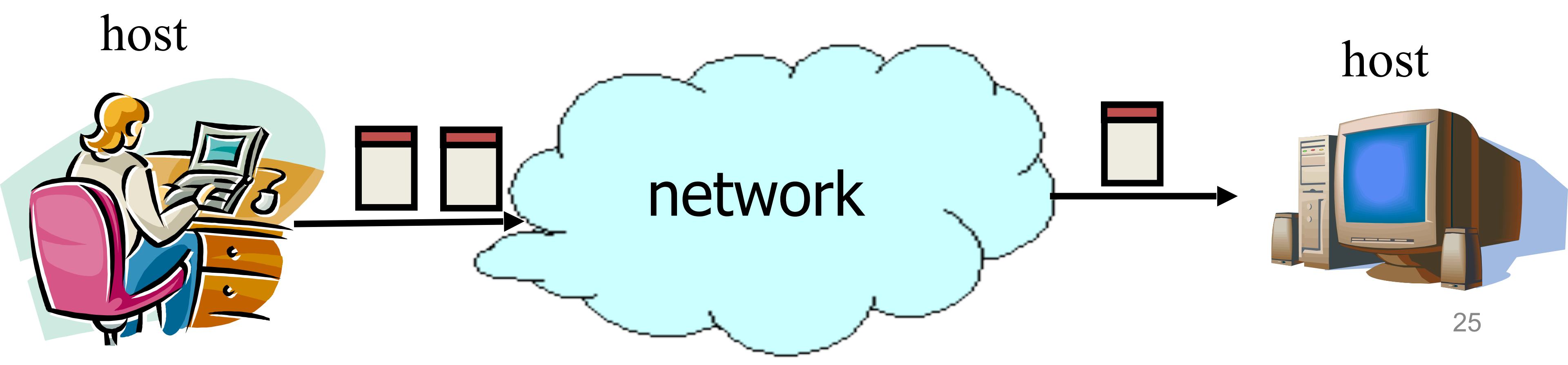
Example: Reliable communication in the Internet

Network

- **Best-effort** delivery between two end-point addresses
- Packets may be lost, corrupted, or come out-of-order

Host

- Everything else
- Retransmit lost/corrupt packets, put packets back in order, …



Why Best-Effort?

Never having to say you're sorry…

- Don't reserve bandwidth and memory
- Don't do error detection & correction
- Don't remember from one packet to next

Easier to survive failures

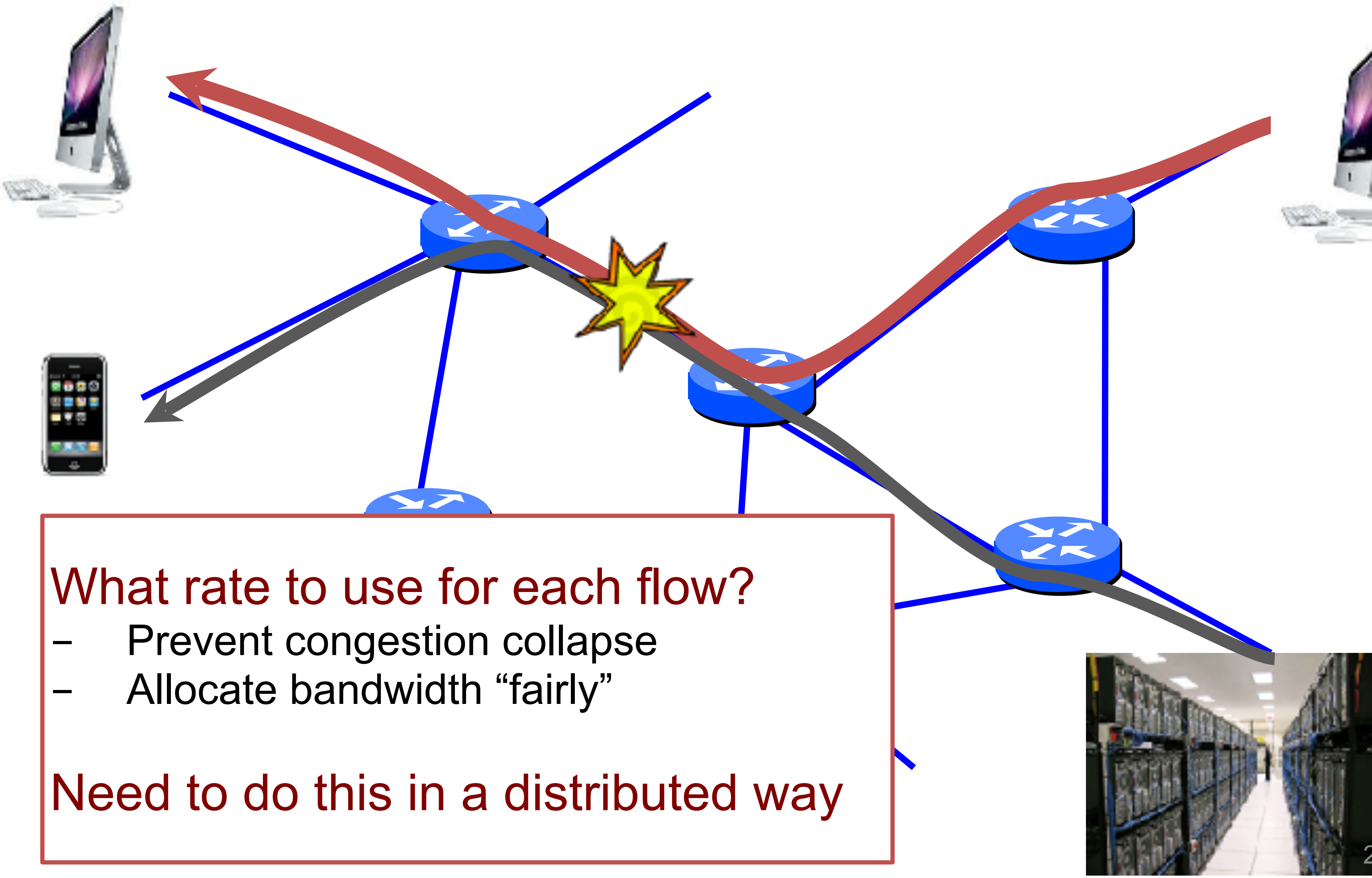
- Transient disruptions are okay during failover

End-to-end argument

- Reliability can only be guaranteed completely end-to-end

What are disadvantages of best-effort delivery?

Congestion Control



What rate to use for each flow?

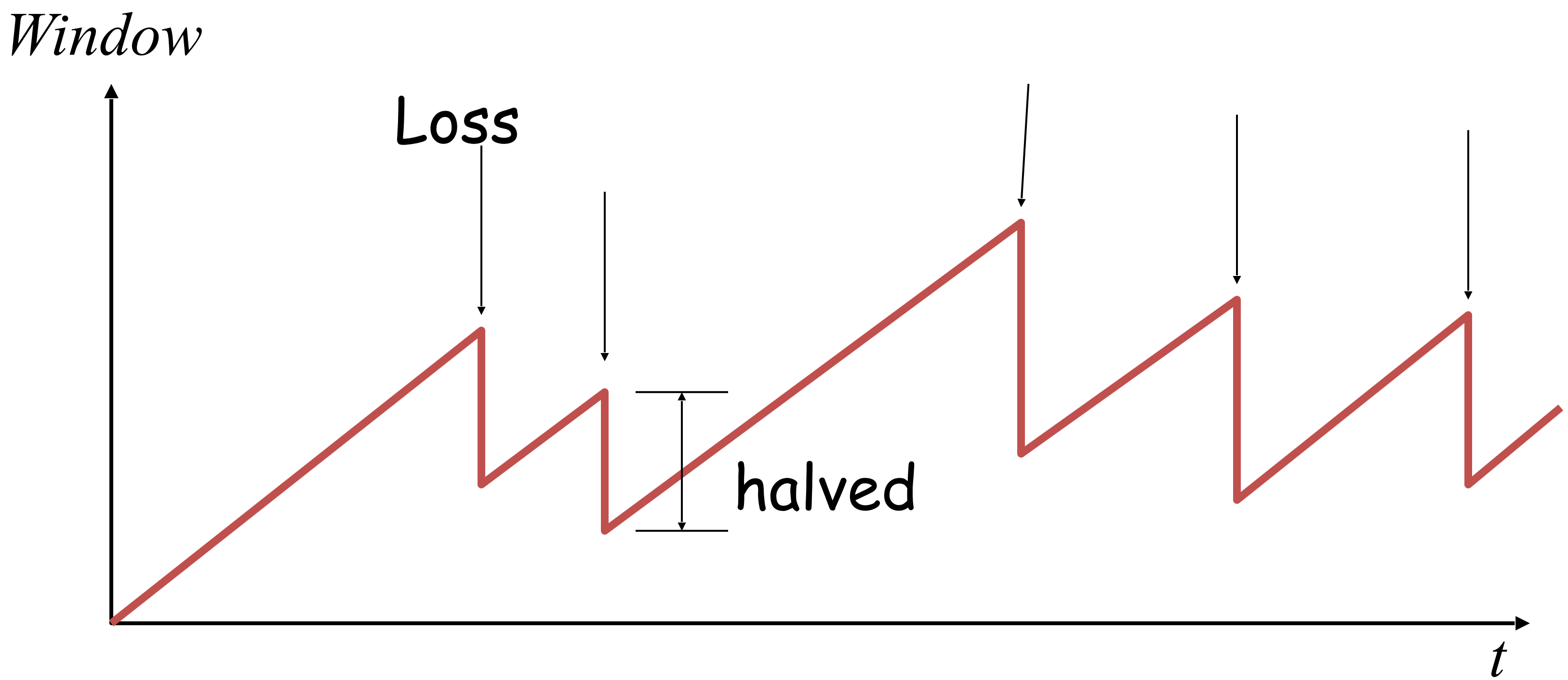
- Prevent congestion collapse
- Allocate bandwidth “fairly”

Need to do this in a distributed way

TCP Congestion Control

Additive increase, multiplicative decrease

- Each RTT, increase window by one packet
- On packet loss, divide congestion window in half



Some questions we'll study…

What's the "right" congestion signal?

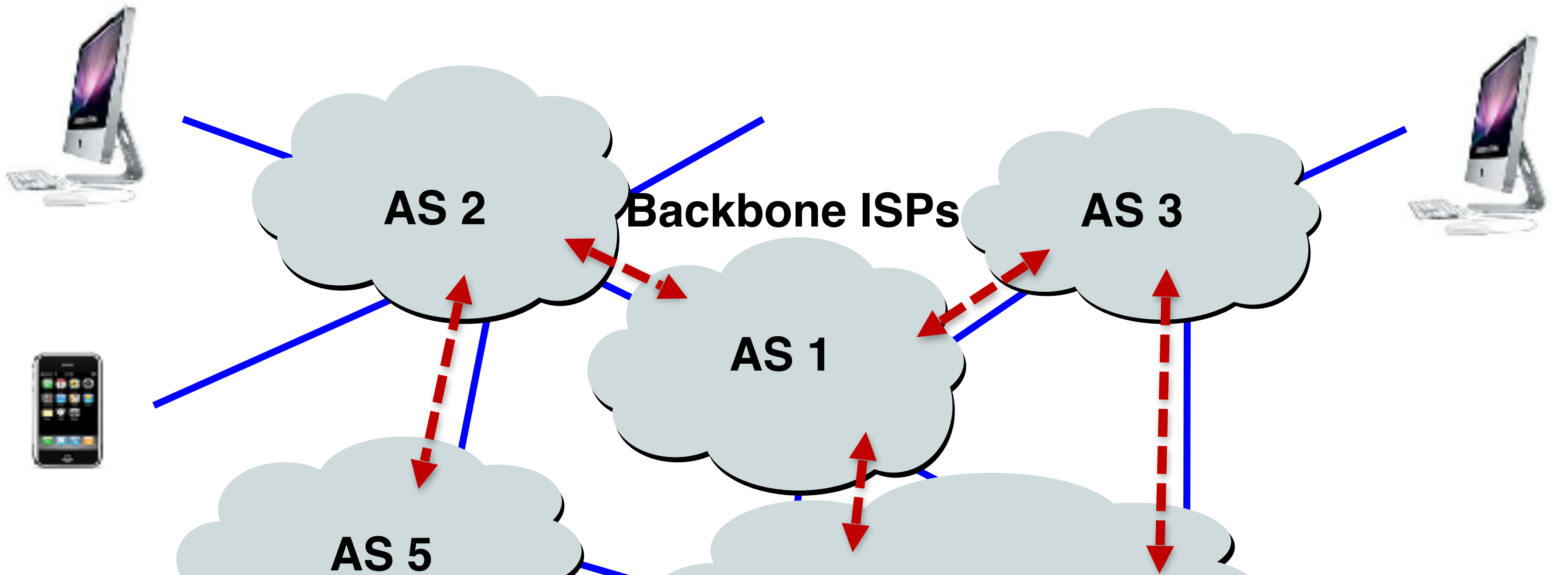
How can routers help with congestion control?

What does it mean to be "fair"?

What if some hosts don't follow the rules?

Inter-Domain Routing

How do routers find an end-to-end path?



Border Gateway Protocol (BGP)

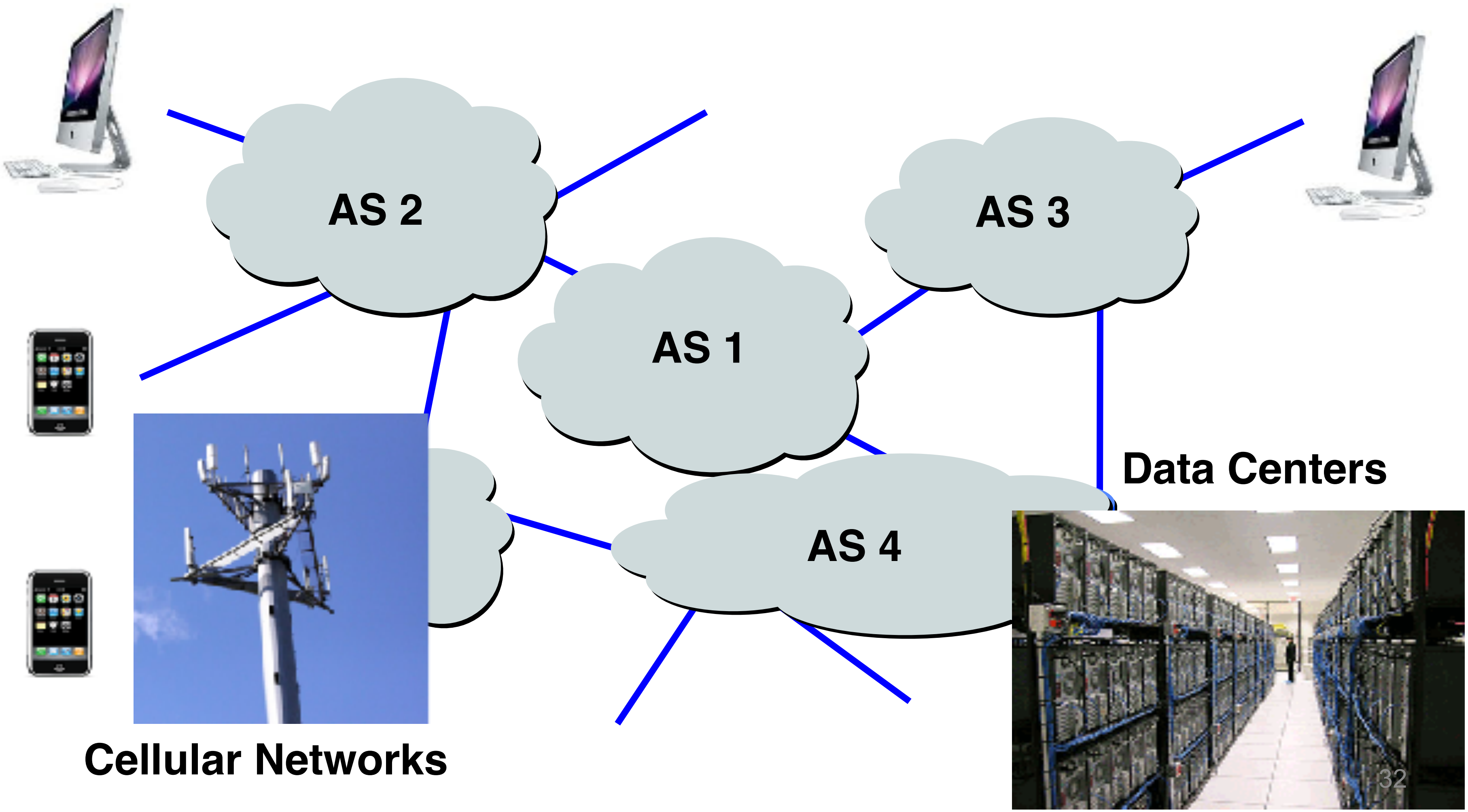
- How do ASes exchange information
- Policy-based routing
  (e.g., path preferences based on business relationships)

Measurement

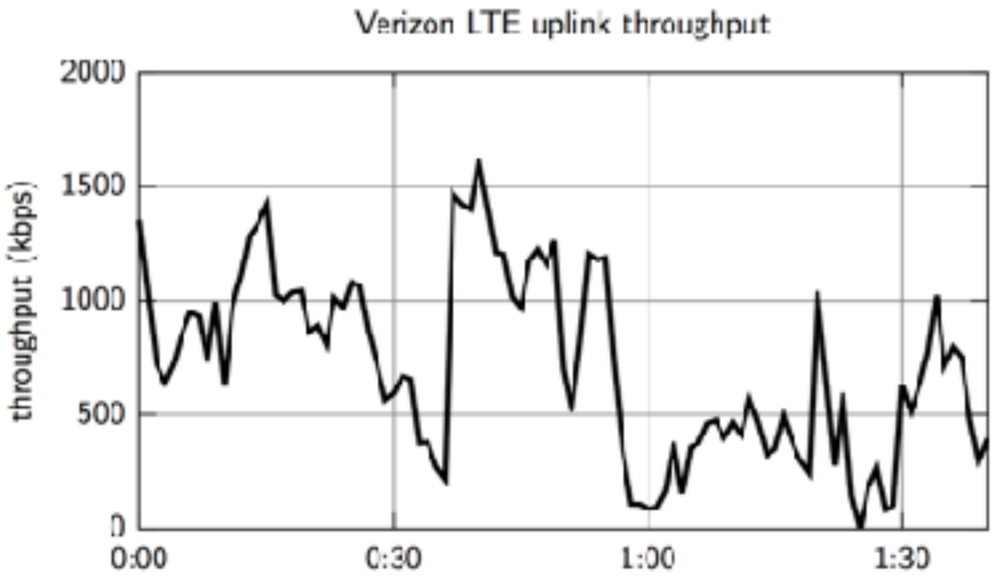
- How do we measure congestion?
- Where does it occur?

“Underlay” Networks

Most of the innovation is at the Internet "Edge"



Cellular Networks



Key Challenges

- Limited wireless spectrum
- Interference
- Highly variable throughput

Data Centers

These things are *really* big

10-100K servers

100s of Petabytes of storage

100s of Terabits/s of Bw
(more than core of Internet)

10-100MW of power
(1-2 % of global energy consumption)

100s of millions of dollars


Microsoft

Networking *Inside* Datacenters

1 user request
→ 100s of Intra DC messages

Tail latency matters
- 99.9th percentile

Datacenter Networking Challenges

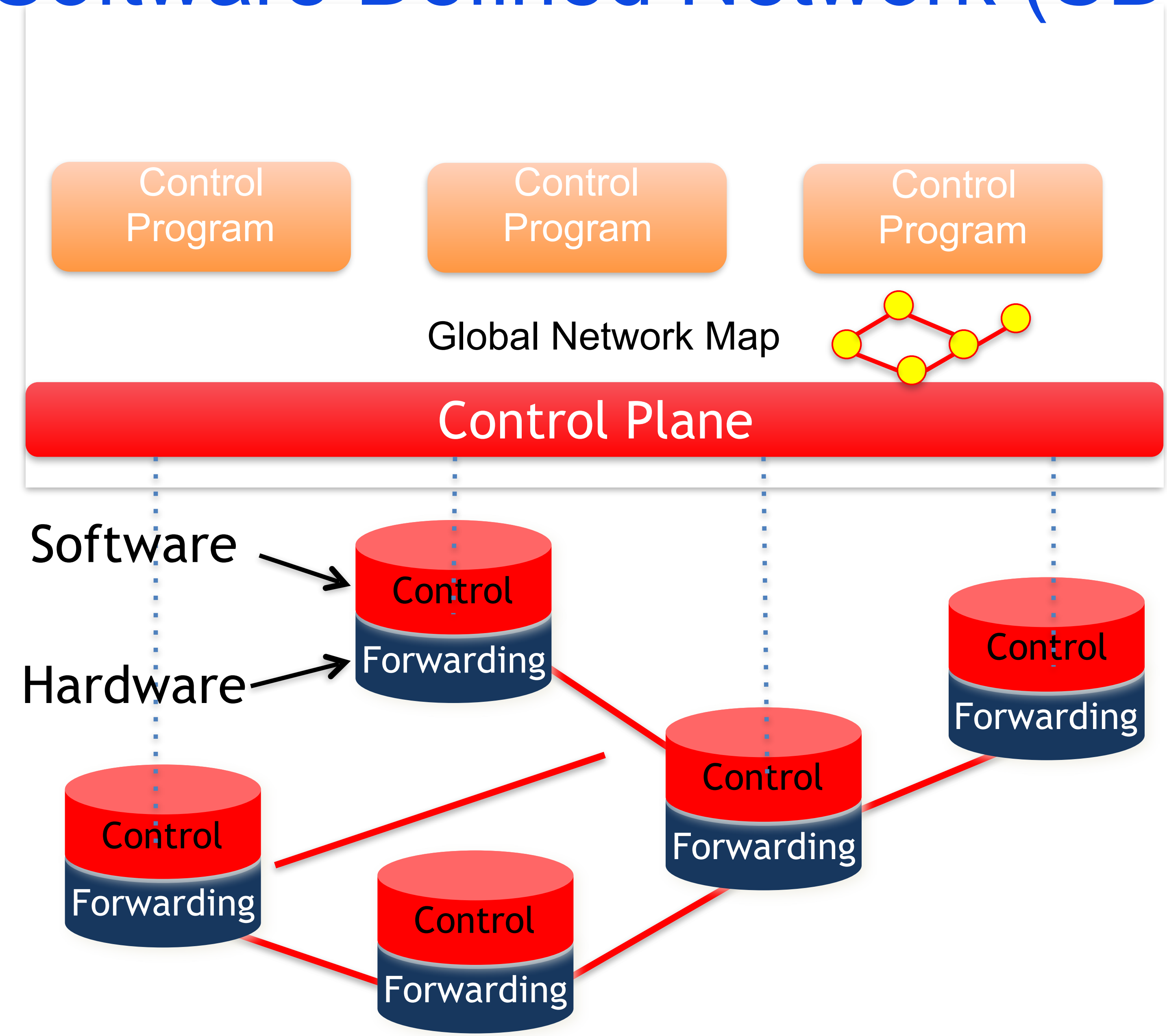
- Very high speed links (10-100Gbps)
- Tiny round trip times (microseconds)
- Dense, multi-path topologies
- Cheap switches with small buffers
- Message latency is King

On the other hand…

- Single administrative domain
- No need to be compatible with outside world

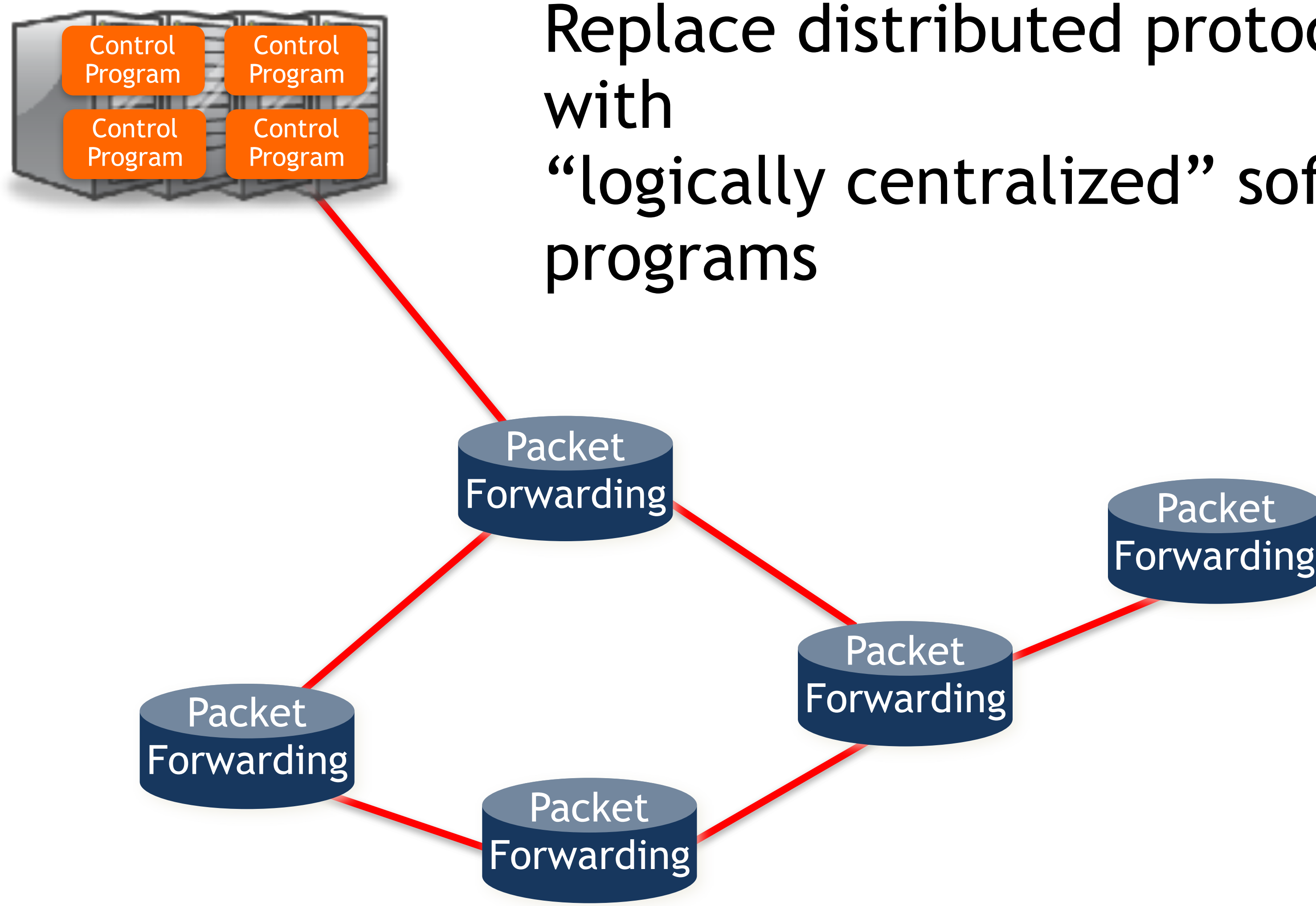
Network Programmability

Software Defined Network (SDN)



Software Defined Network (SDN)

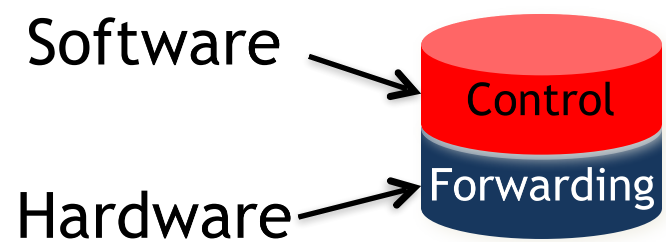
Replace distributed protocols with
"logically centralized" software programs



Programmable Data Plane

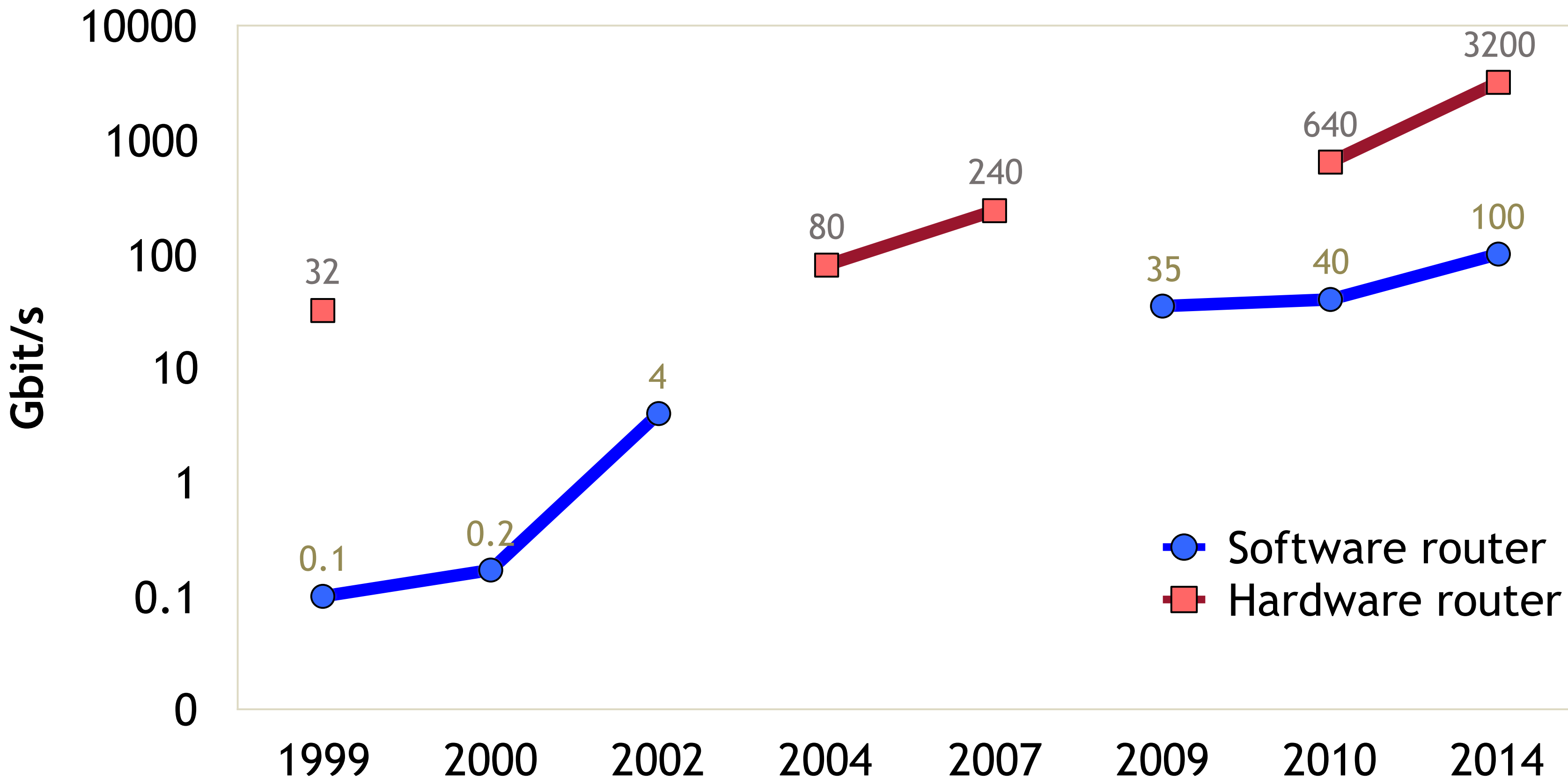
Can we program new data-plane algorithms?

- Congestion control
- Measurement
- Load balancing
- Packet scheduling



Today, these algorithms are hard-coded into router hardware

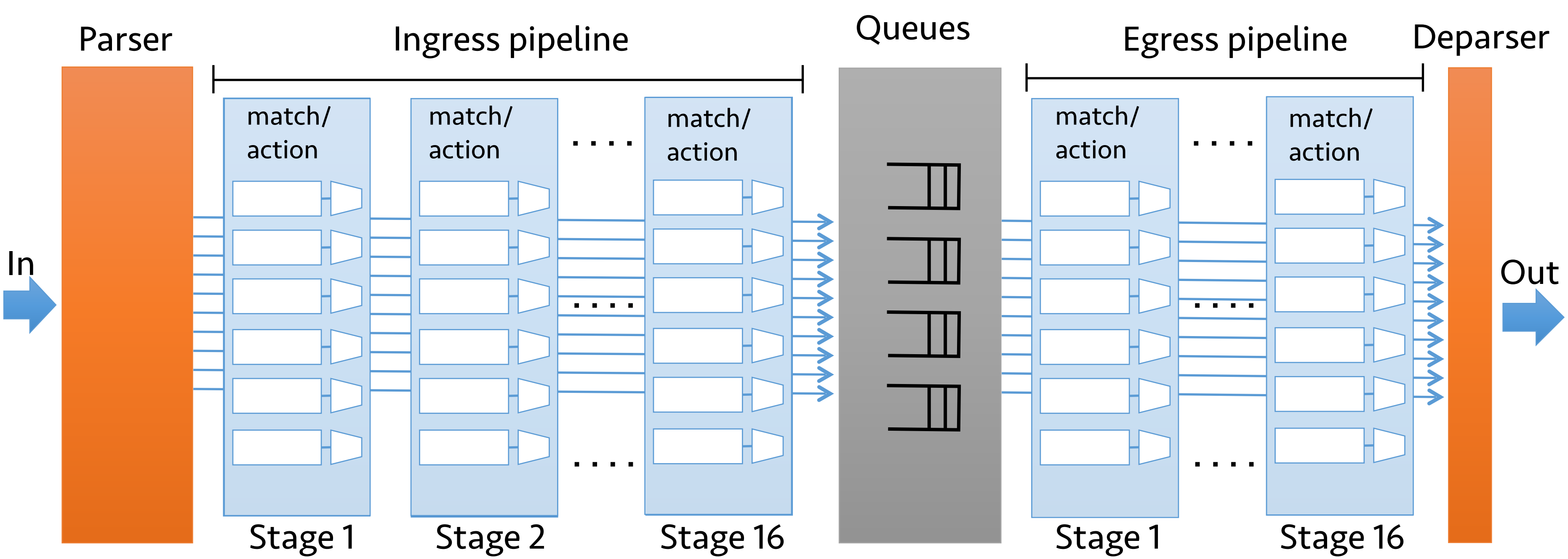
Software vs. Hardware routers



10—100X gap between hardware and software routers

Programmable router hardware

Same performance as fixed-function chips,
Some programmability



“Overlay” Networks

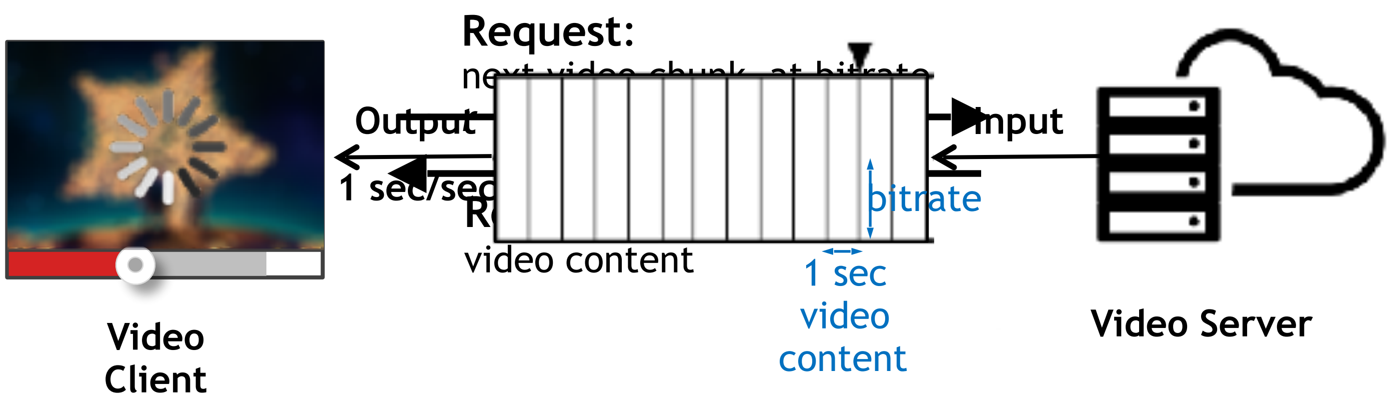
Video is the BIG thing on the Internet

Video is more than 50% of peak traffic in the US

Trend: Streaming over HTTP

- Content Distribution Networks (CDNs)
  - Well-provisioned HTTP servers at the edge of the Internet
- Firewall friendliness

Dynamic Streaming over HTTP



How to pick video resolution (bitrate) based on network conditions?

How does the transport (TCP) impact video quality?

How should the video codec and transport interact?

Animation from Te-Yuan Huang (SIGCOMM '14) http://conferences.sigcomm.org/sigcomm/2014/doc/slides/38.pdf

Our first two papers…

THE DESIGN PHILOSOPHY OF THE DARPA INTERNET PROTOCOLS

David D. Clark
Massachusetts Institute of Technology
Laboratory for Computer Science (now CSAIL)
Cambridge, Ma. 02139

This paper was originally published in 1988 in ACM SigComm. Revised, with extensive commentary, 2013[1]. The original text has been reformatted, but is otherwise unchanged from the original except for a few spelling corrections.

Version 1.2 of March 14, 2013

Limited distribution for comments.

Abstract

The Internet protocol suite, TCP/IP, was first proposed fifteen years ago. It was developed by the Defense Advanced Research Projects Agency (DARPA), and has been used widely in military and commercial systems. While there have been papers and specifications that describe how the protocols work, it is sometimes difficult to deduce from these why the protocol is as it is. For example, the Internet protocol is based on a connectionless or datagram mode of service. The motivation for this has been greatly misunderstood. This paper attempts to capture some of the early reasoning which shaped the Internet protocols.

Why the Internet only just works

M Handley

The core Internet protocols have not changed significantly in more than a decade, in spite of exponential growth in the number of Internet users and the speed of the fastest links. The requirements placed on the net are also changing, as digital convergence finally occurs. Will the Internet cope gracefully with all this change, or are the cracks already beginning to show? In this paper I examine how the Internet has coped with past challenges resulting in attempts to change the architecture and core protocols of the Internet. Unfortunately, the recent history of failed architectural changes does not bode well. With this history in mind, I explore some of the challenges currently facing the Internet.

1. Introduction

The Internet only just works. I want to make it clear though, right from the start, that this is not a forecast of imminent doom and disaster. My reasons for making this assertion are twofold. Firstly, I believe that this has historically been the natural state of the Internet and it is likely to remain so in future. Unless this is understood,

most obvious are remote log-in, e-mail, and file transfer, but there were also ARPAnet experiments with packet voice, which predate common usage of voice-over-IP by over twenty years.

The ARPAnet was very successful, but it was also clear that flexibility should be of prime importance in the