

CSE 124: Networked Services

Prof. George Porter

January 6, 2015

Outline

- Course Mechanics
- Course Topics/Outline
 - Networking/Distributed Systems problem statement
- Introduction to Computer Networks

Audience

- Who should take this course
 - Those interested in learning about large-scale network services like Google, Youtube, Akamai, Cloud computing...
Learn how things work “soup to nuts”
 - Those interested in graduate school
 - Those interested in top industrial positions
Google, Microsoft, Cisco, Amazon, Yahoo, Akamai, IBM, Apple, etc.

Specifics

- Professor: George Porter, gmporter@cs.ucsd.edu
 - Group office hours: 3104 CSE (**typically Wed 4-5pm**, but for this week they will be *today* from 4-5pm)
- Teaching Staff
 - Yashar Asgarieh
 - Jake Maskiewicz
 - Office Hours: TBD
- Course Web Page
 - <http://www.cs.ucsd.edu/~gmporter/classes/wi15/cse124/>
- Discussion board & mailing list
 - <https://piazza.com/ucsd/winter2015/cse124/home>
 - Sign up today!

General Goals

- Gain background in networking and distributed systems
 - Homework, textbook, lectures
- Understanding of some emerging (research) issues
 - Study of relevant research papers
- Building large-scale distributed systems
 - Lectures, programming projects
- Overall: emphasize “*why*” and “*how*” over “*what*”

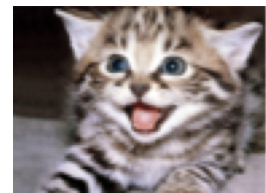
Non-Goals

- Teach the basics of programming, unix, files, ...
 - Assume familiarity with operating systems C, C++, Java
- 75 minute soliloquies
 - Lectures should be interactive!
 - Occasional in-class demos
 - Will be incorporating the iClicker this term
- Insulate the professor from the students

Grading

- 5% Class Participation
 - Success of class depends on student participation
 - Piazza, in-class, discussion section, ...
- 10% Homework
 - Paper evaluations
- 40% Exams
 - 15% midterm, 25% final exam
- 45% Programming projects (3)
 - Assignment 1: HTTP Server
 - Assignment 2: MapReduce using Hadoop
 - Assignment 3: CatContest photo ranking service

Google™



Paper Evaluations

- Will read 4-5 research papers on constructing high-performance distributed systems
- 1-2 page evaluation of reading for each paper
- Evaluations submitted *in advance* of class
- Describe:
 - Biggest contribution of the paper
 - Most glaring problem with the work
 - What the work implies for building robust, scalable distributed systems and networks

Course Projects

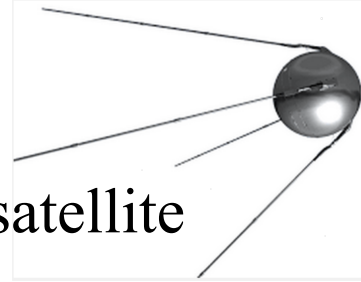
- 3 projects spanning the term
 - Hands on construction of interesting distributed services
- Assignment 1:
 - Build an HTTP Server in C/C++
 - Work in teams of 2
 - Support subset of HTTP/1.0 and HTTP/1.1 functionality
 - Details on web page
 - Due date: Feb 3, 2015

Outline

- Course Mechanics
- Course Topics/Outline
 - Networking/Distributed Systems problem statement
- Introduction to Computer Networks

Internet History

Background

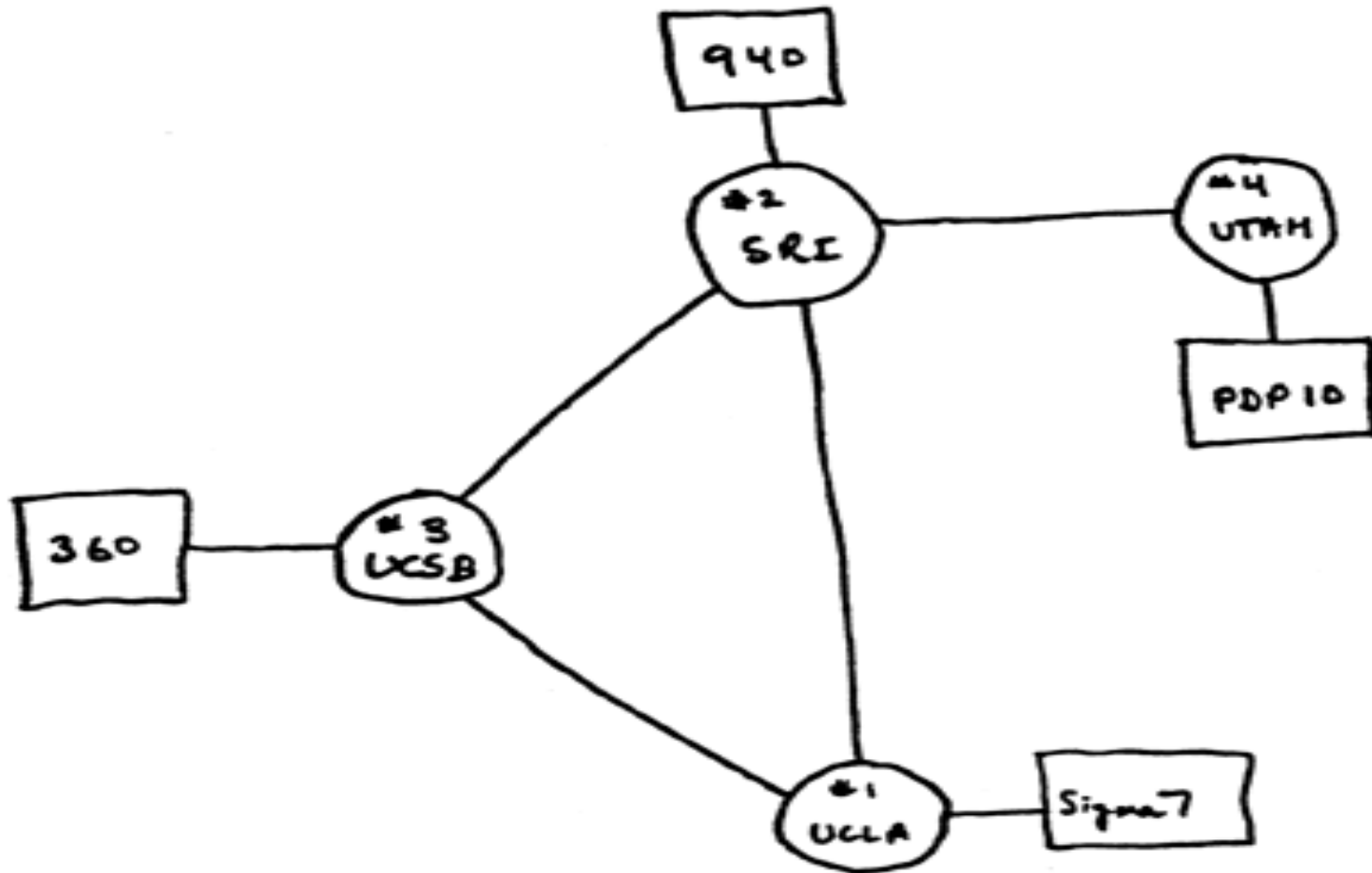


- 1957: USSR launches Sputnik, first artificial earth satellite
 - U.S. responds by forming ARPA
- 1962: Licklider's *Galactic Network*
- 1966: Roberts (MIT) *Towards a Cooperative Network of Time-Shared Computers*
- 1966: "Star Trek" debuts



- 1967: ACM SOSP *Multiple Computer Networks and Intercomputer Communication*

1969 Internet Map



1969 Internet Map

940

Q: What was the first message that was sent?

- [a] “Come quick dear Watson!”
- [b] “LOGIN”
- [c] A classified military message
- [d] “LO” then the network crashed...

UCLA

Signa7

10:30pm, October 29, 1969, Los Angeles, CA

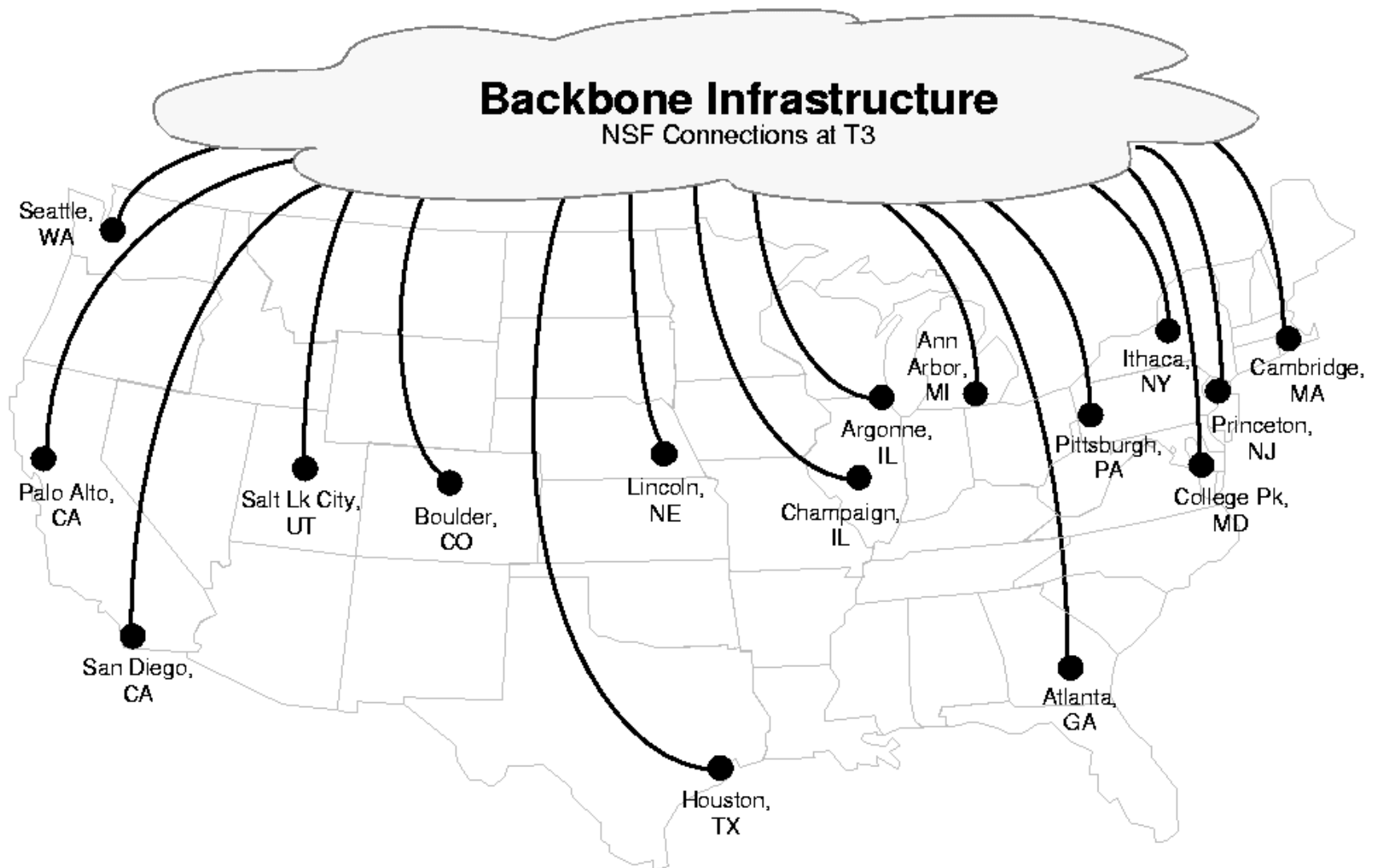


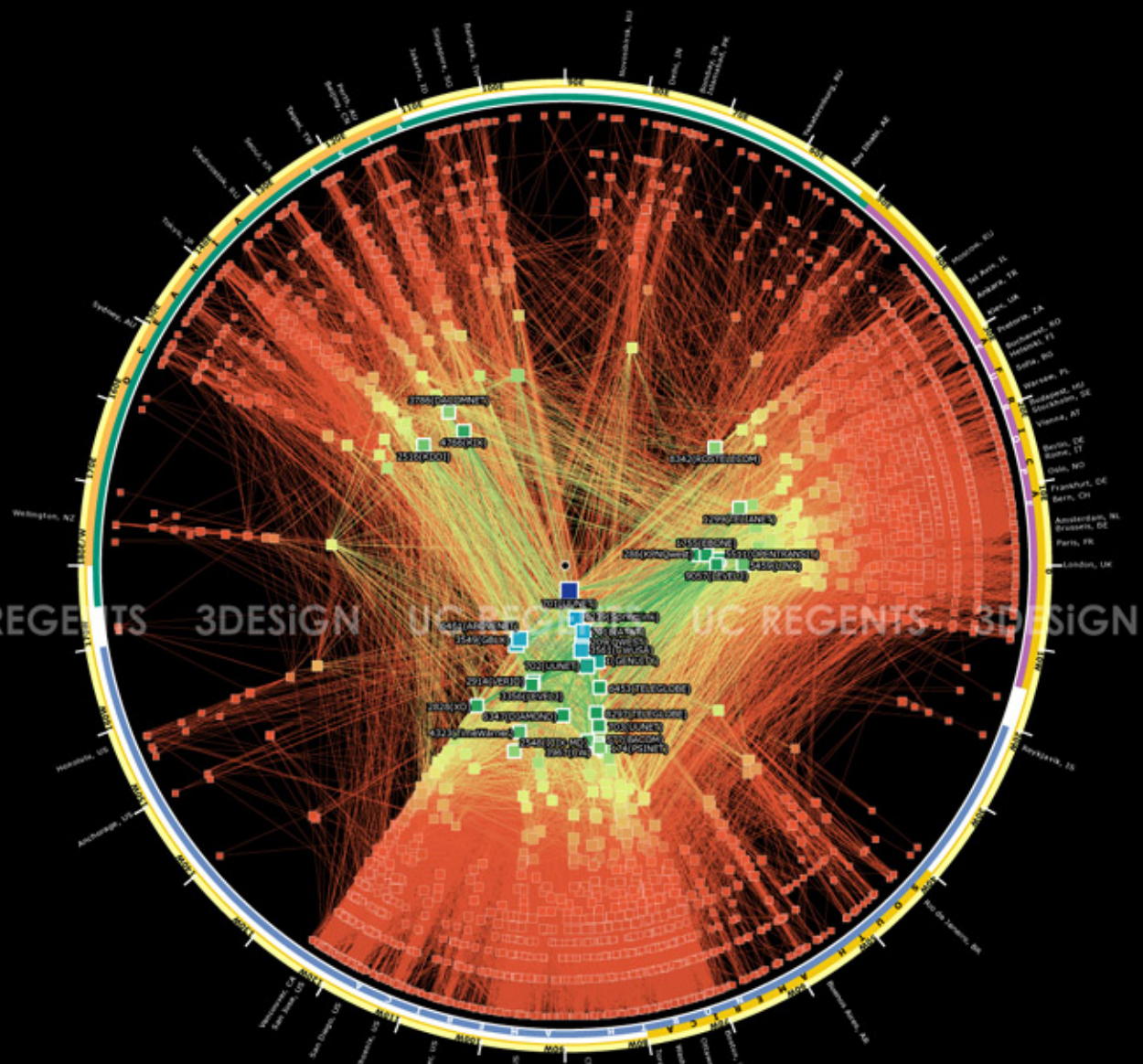
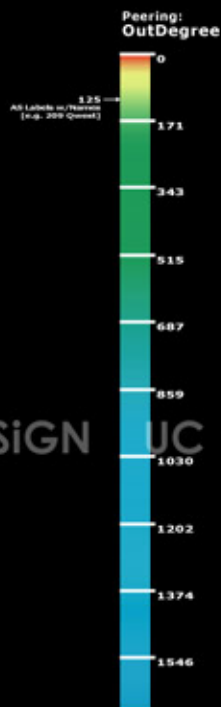
- UCLA student Charley Kline
- Keinrock's lab
- “LOGIN” to SRI on an SDS Sigma 7 computer
- Only ‘LO’ gets sent the 400 miles to Menlo Park
- 1 hour to recover from crash

NSFNET 1991

Backbone Infrastructure

NSF Connections at T3





skitter
AS network graph

This visualization represents a macroscopic snapshot of the Internet for two weeks.
01 April 2002 - 16 April 2002

The graph reflects 1,234,733 IP addresses and 2,003,194 IP links (immediately adjacent addresses in a hierarchical-like path) of topology data gathered from 21 monitors peering approximately 932,000 destinations spread across 75,000 (70% of the total) globally routable network prefixes.

We then aggregate this view of the network into a topology of Autonomous Systems (ASes), each of which approximately maps to an Internet Service Provider (ISP). We show each IP address in the AS responsible for routing it, i.e. the origin (end of path) AS for the next hop IP prefix of the address in Border Gateway Protocol (BGP) routing tables.

We use the largest publicly available set of over 100K routing tables, which are collected by the University of Oregon's RouteView project, and specifically use prefixes present in the majority of these 141 tables. For prefixes with multiple origin ASes, we used the most frequently occurring origin AS. The abstracted graph consists of 11,122 Autonomous Systems (ASes) nodes and 20,701 peering relations between pairs of these nodes.

For 123 ASes we could not determine a geographical location. The resulting graph contains 10,999 AS (94% of all ASes present in RouteView's BGP tables on April 10, 2002) and 34,209 peering relations.

The position of each AS node is plotted in polar coordinates, (azimuth, angle) (azimuth) is where

radius = $\sqrt{\text{OutDegree}}$
angle = θ
(longitude of the AS headquarters in which records)

The outdegree of an AS node is the number of east leg ASes that we observed accepting traffic from this AS.

Graphing dimensions of peering richness and geographic information reveals the highly "concentric" nature of certain ASes based in North America. However, since the creation of our last graph in August 2001, European ISPs have increased their presence among the most highly connected ASes. While ISPs in Europe and Asia have many peering relationships with ISPs in the U.S. there are few links directly between ISPs in Asia and Europe. Both technical (routing and router placement) and non-technical, as well as policy (insurance and road models), geo-political considerations factors contribute to peering arrangements represented in this graph.

One of CAIDA's skitter project goals is to develop techniques to illustrate relationships and depict critical components of the Internet infrastructure.

Note: Link directions are not labeled. Its enough to see the edges of the graph have links, their multiplicity is seen.

CAIDA - cooperative association for internet data analysis
www.caida.org

**topology mapping
ANALYSIS TEAM**

Bradley Huffaker
Andres Balle
Maura Fomenko
Young Hyeon
Dan Anderson
Ken Keys
David Moore
to staff

**skitter
DEVELOPER**

Daniel McMillan

**topology analysis
MONITOR HOSTS**

AFN: Univ. of Oregon - ANTC
CANET: Univ. of Bristol
IPN: Univ. of Bonn, UC
HCAR: Wijn Univ., Amsterdam
RISAC

**poster
DESIGN**

Oliver Jakobson



Back to the Future: Cloud Computing

Americas

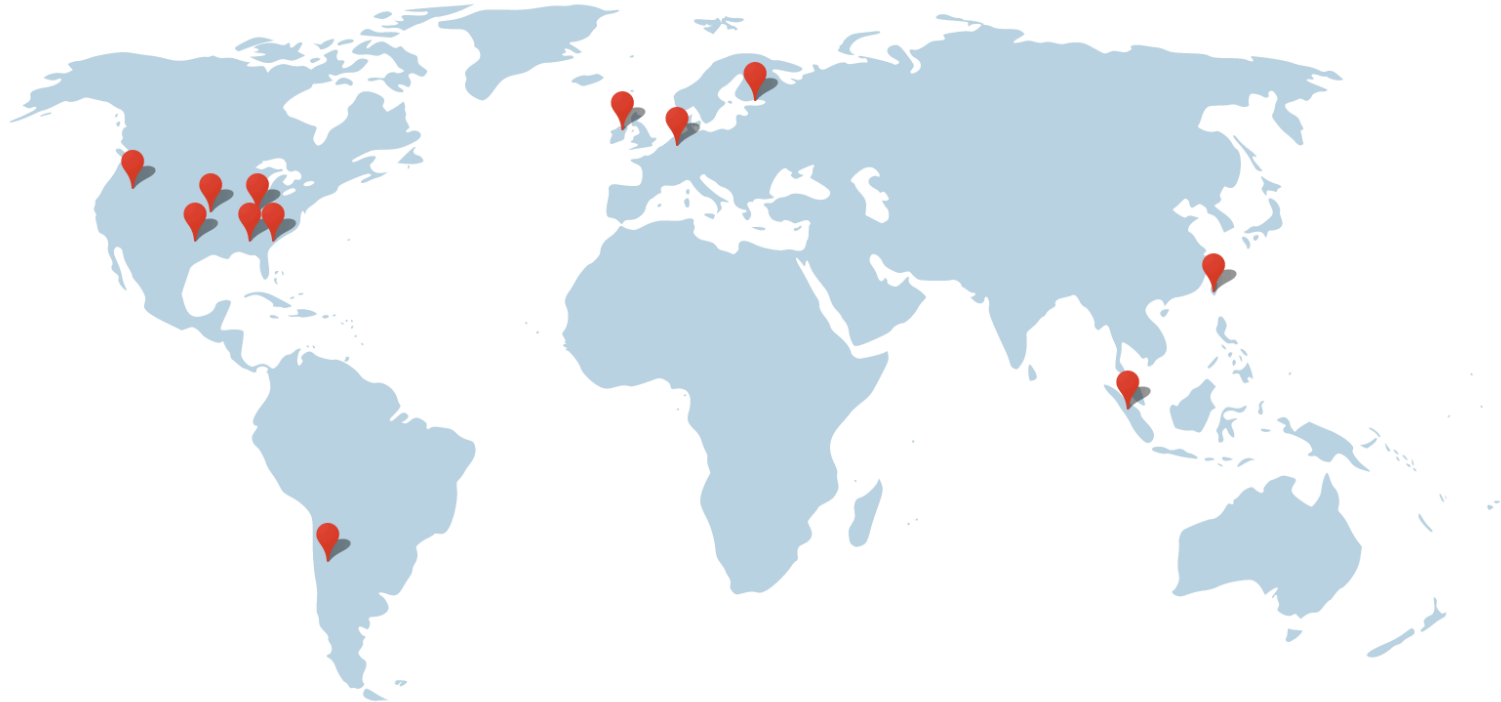
Berkeley County, South Carolina
Council Bluffs, Iowa
Douglas County, Georgia
Quilicura, Chile
Mayes County, Oklahoma
Lenoir, North Carolina
The Dalles, Oregon

Asia

Changhua County, Taiwan
Singapore

Europe

Hamina, Finland
St Ghislain, Belgium
Dublin, Ireland



Google's Worldwide Data Centers

<http://www.google.com/about/datacenters/inside/locations/>

Internet Timeline

- 1971: Tomlinson develops email program, big hit
- 1972: Telnet
- 1973: FTP
- 1974: TCP
- 1978: TCP split into TCP and IP
- 1979: USENET established using UUCP between Duke and UNC by Tom Truscott, Jim Ellis, and Steve Bellovin
- 1984: 1000 hosts connected to Internet, DNS introduced
- 1988: Internet worm brings down 10% of Internet
- 1991: WAIS, Gopher, WWW released

Internet Growth Trends

- 1977: 111 hosts on Internet
- 1981: 213 hosts
- 1983: 562 hosts
- 1984: 1,000 hosts
- 1986: 5,000 hosts
- 1987: 10,000 hosts
- 1989: 100,000 hosts
- 1992: 1,000,000 hosts
- 2001: 150 – 175 million hosts
- 2002: over 200 million hosts
- 2014: ~1B hosts (~3B users)

Major Course Topics

- Internetworking
 - Not all computers are directly connected
 - The Internet Protocol (IP)
- End-to-End Protocols
 - E.g., provide the abstraction of a reliable byte-stream over error-prone, packet-switched network
 - Transmission Control Protocol (TCP), Congestion Control
- Modern network services
 - How is Google actually built
 - Parallel computing
 - GFS/Map Reduce

Major Course Topics (cont.)

- Naming
 - Given a name for an Internet resource how do you find it?
- Remote Procedure Call
 - Client/Server systems
- Distributed operating systems/file systems
 - How do you break down a centralized service to run across multiple machines?
Client/server, peer-to-peer

Major Course Topics (cont.)

- Replication/Fault Tolerance
 - Copy service contents for increased availability, performance
 - Data consistency?
- Security
 - How to ensure authenticity and integrity of data transmitted across machines

Outline

- Course Mechanics
- Course Topics/Outline
 - Networking/Distributed Systems problem statement
- Introduction to Computer Networks

Some Definitions

- Host – computer, iPad, ...
- Packet – unit of transmission across a network
- Link – transmit bits
 - Wire or wireless, broadcast or switched (or both)
- Switch – move bits between links
 - Packet switching: stateless, store & forward
 - Circuit switching: stateful, cut through

Types of Connections

- Point-to-point

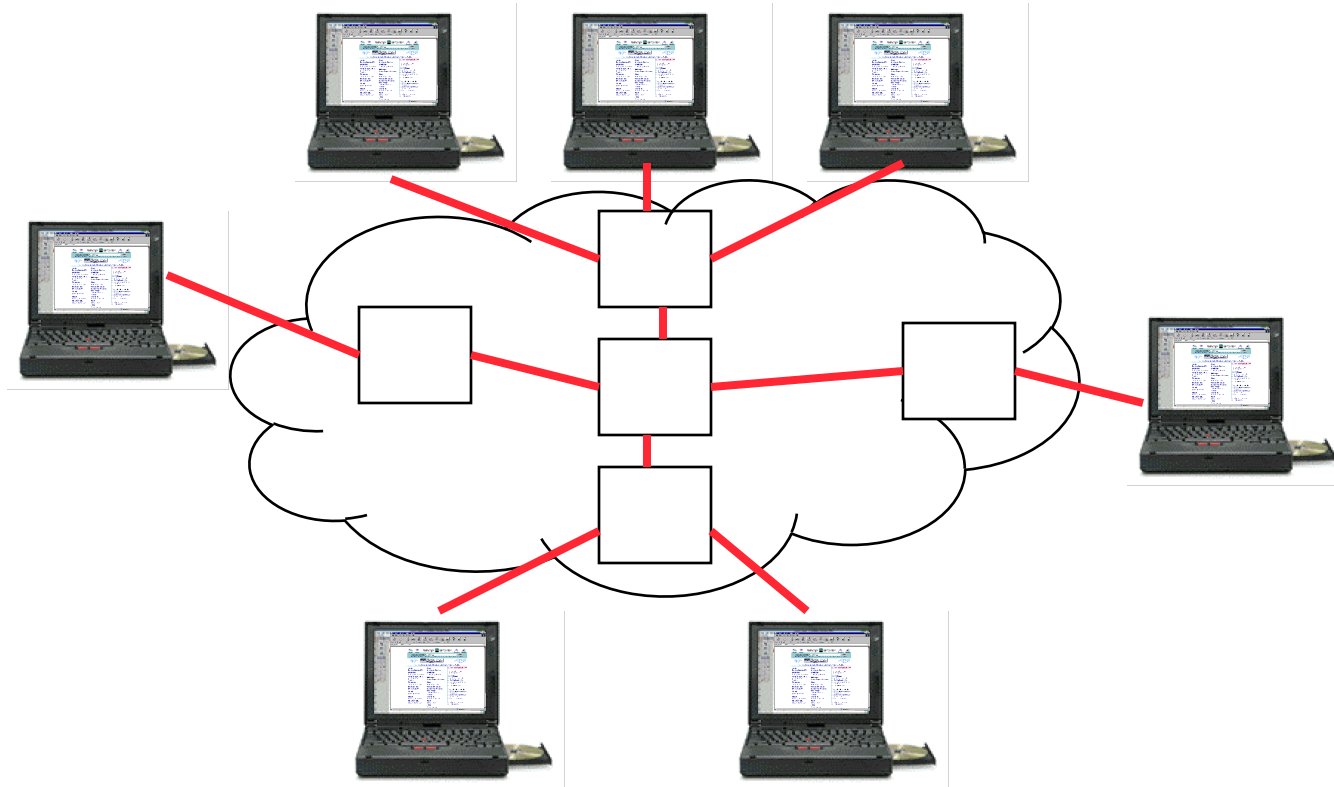


- Multiple access

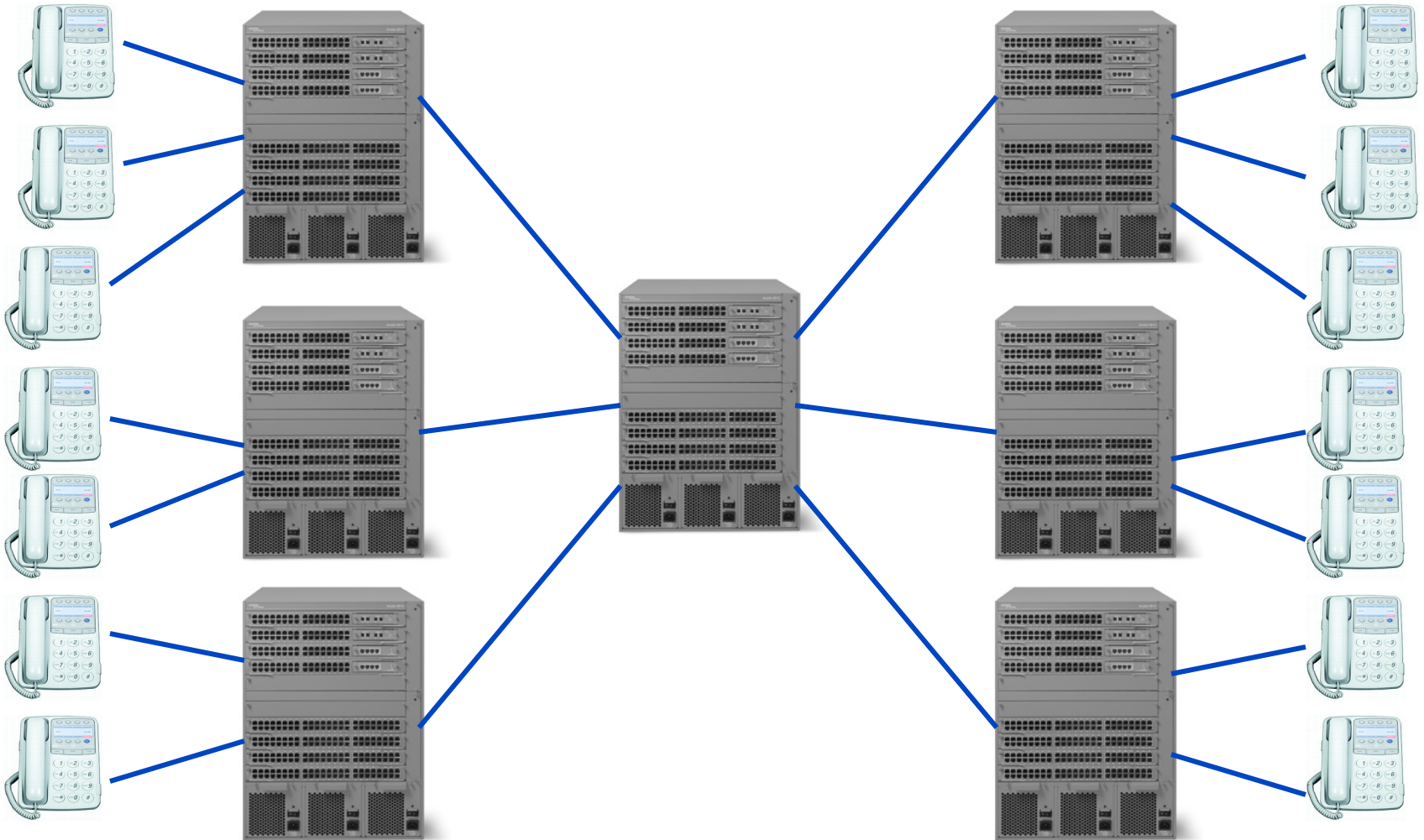


Types of Connections

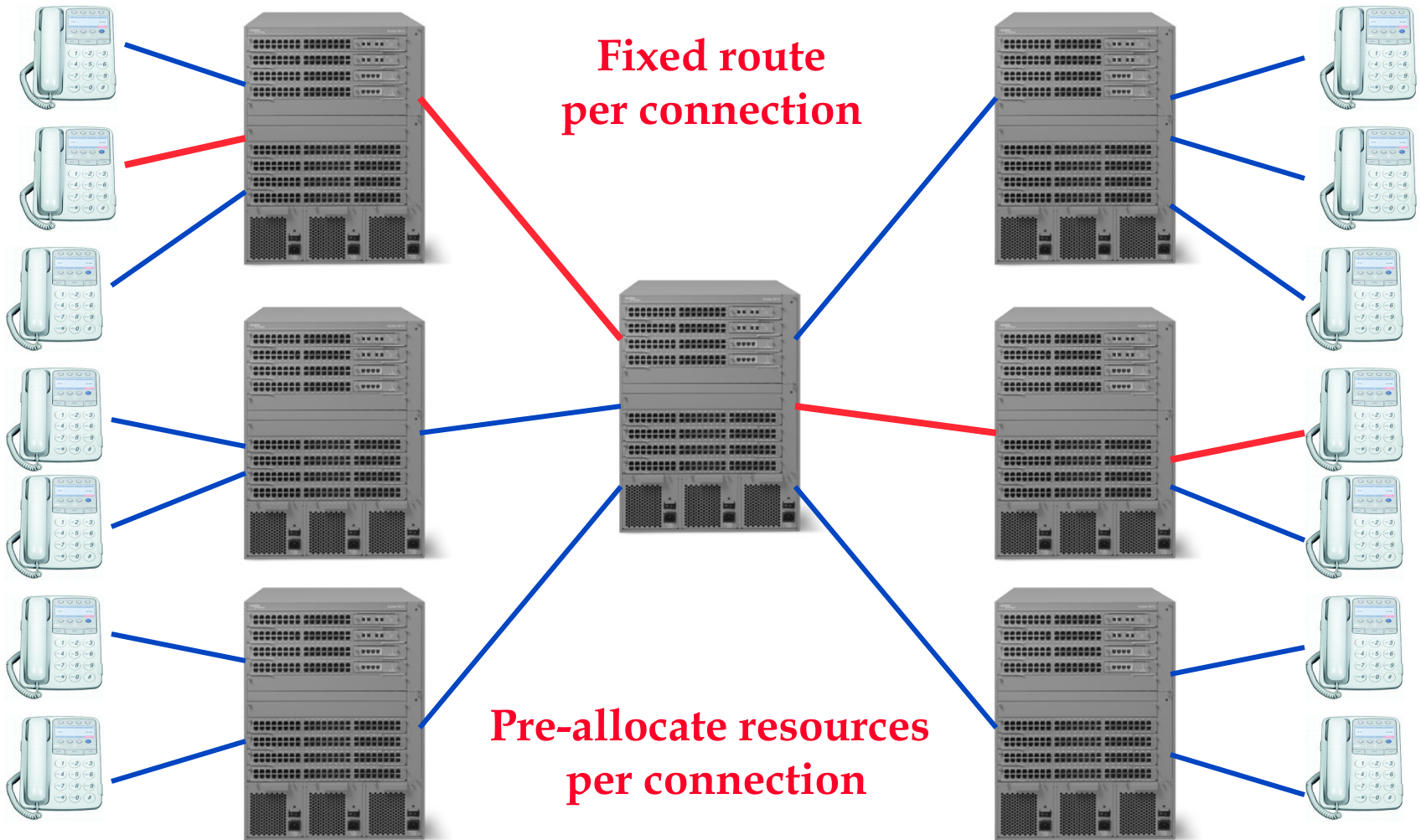
- Switched



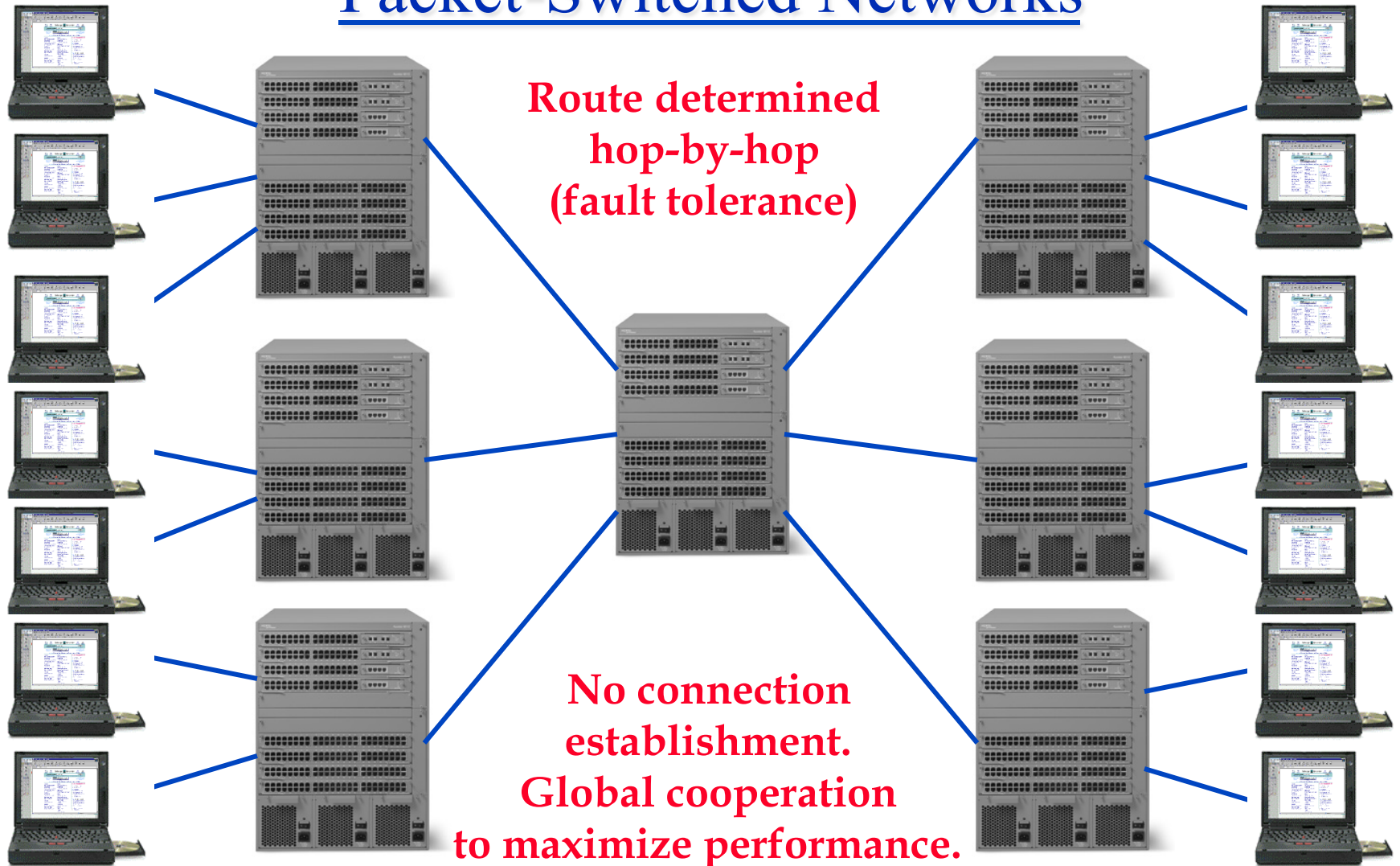
Circuit-Switched Networks



Circuit-Switched Networks



Packet-Switched Networks

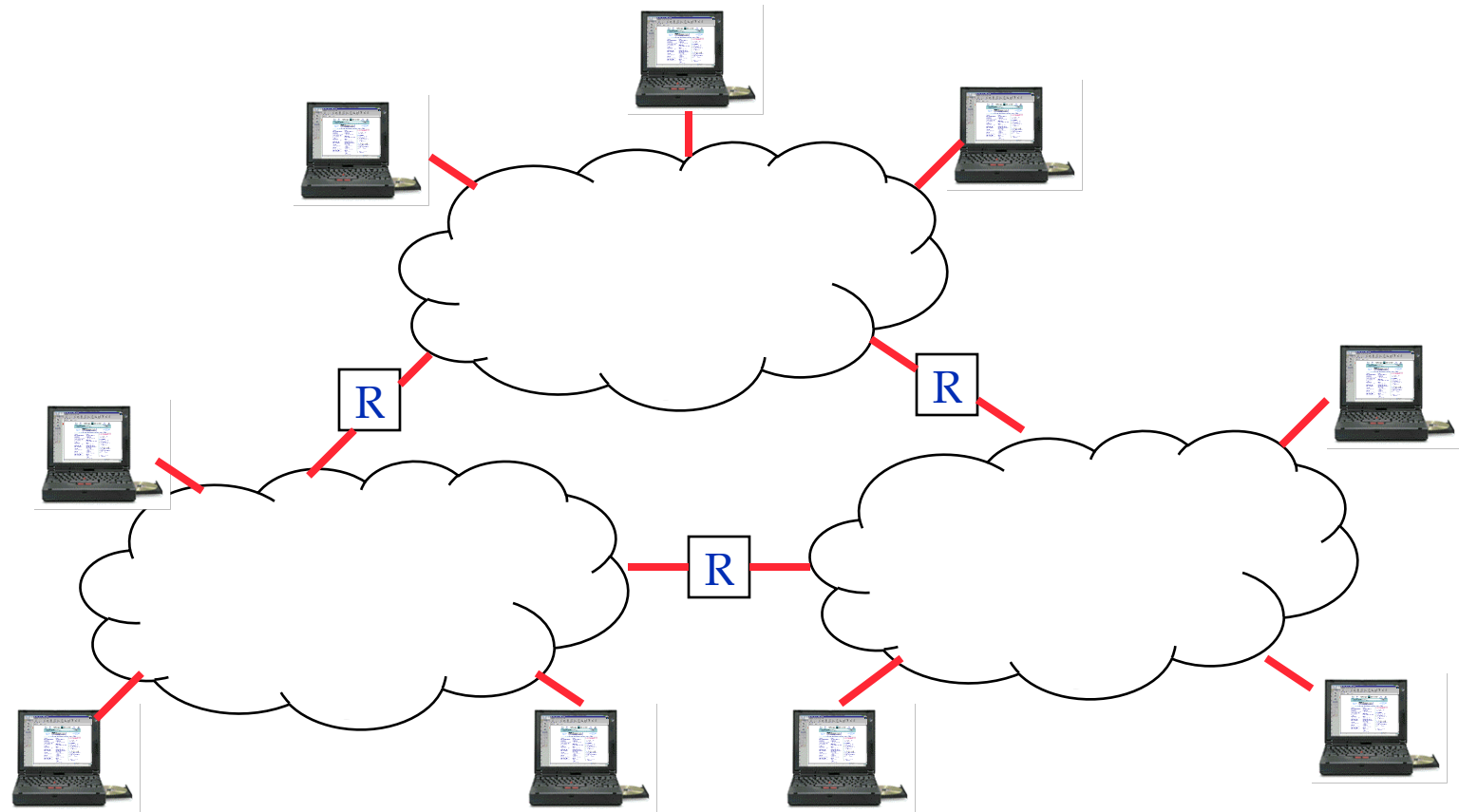


Circuit-Switched vs. Packet-Switched Networks

- Circuit-switched networks
 - Traditionally used for telephone networks (ATM)
 - Pre-allocate resources per connection
 - Pre-determine route on connection setup
 - Fixed bandwidth requirements per-connection
- Packet-switched networks
 - Internet
 - No pre-determined allocation of resources (coming though)
 - No connection setup, no pre-determined route
 - More fault tolerance
 - Efficient utilization in the face of bursty traffic patterns

Types of Connections

- Interconnection of networks



Q: Why packet switching?

Much of the Internet's success has been its basis in packet switching. In this context, what is the primary advantage of packet switching vs. circuit switching?

[a] Packet switching is more **reliable** than circuit switching

[b] Packet switching is more **efficient/cheaper** than circuit switching

[c] Packet switching is **faster** than circuit switching

Q: Why packet switching?

Much of the Internet's success has been its basis in packet switching. In this context, what is the primary advantage of packet switching vs. circuit switching?

[b] Packet switching is more efficient/cheaper than circuit switching

In the ARPAnet/Internet, wide-area links are scarce and expensive. Packet switching enables statistically multiplexing these expensive links over many users/connections, increasing efficiency.

Updated by: [2549](#), [6214](#)

EXPERIMENTAL

Errata Exist

Network Working Group

D. Waitzman

Request for Comments: 1149

BBN STC

1 April 1990

A Standard for the Transmission of IP Datagrams on Avian Carriers

Status of this Memo

This memo describes an experimental method for the encapsulation of IP datagrams in avian carriers. This specification is primarily useful in Metropolitan Area Networks. This is an experimental, not recommended standard. Distribution of this memo is unlimited.

Overview and Rational

Avian carriers can provide high delay, low throughput, and low altitude service. The connection topology is limited to a single point-to-point path for each carrier, used with standard carriers, but many carriers can be used without significant interference with each other, outside of early spring. This is because of the 3D ether space available to the carriers, in contrast to the 1D ether used by IEEE802.3. The carriers have an intrinsic collision avoidance system, which increases availability. Unlike some network technologies, such as packet radio, communication is not limited to line-of-sight distance. Connection oriented service is available in some cities, usually based upon a central hub topology.

Frame Format

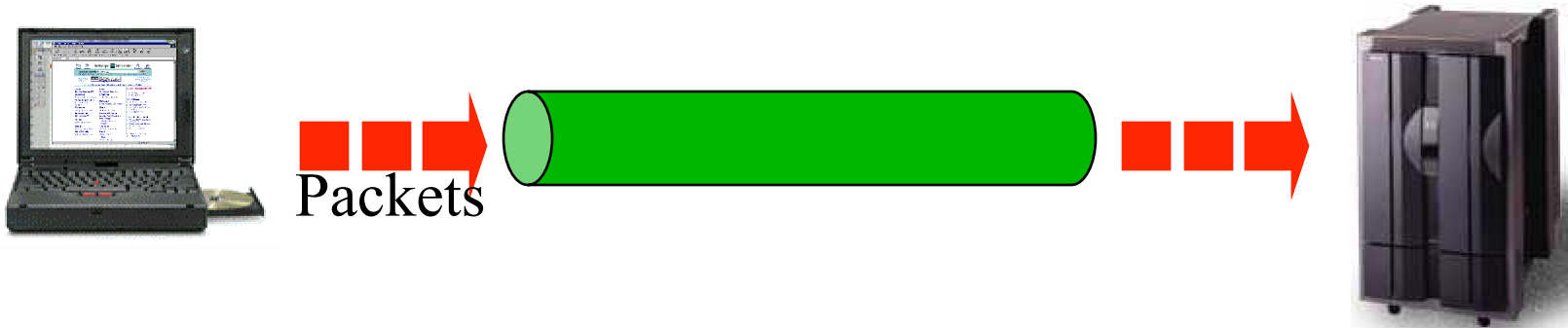
The IP datagram is printed, on a small scroll of paper, in hexadecimal, with each octet separated by whitestuff and blackstuff. The scroll of paper is wrapped around one leg of the avian carrier. A band of duct tape is used to secure the datagram's edges. The bandwidth is limited to the leg length. The MTU is variable, and paradoxically, generally increases with increased carrier age. A typical MTU is 256 milligrams. Some datagram padding may be needed.

Upon receipt, the duct tape is removed and the paper copy of the datagram is optically scanned into a electronically transmittable form.

How is the Network Characterized?

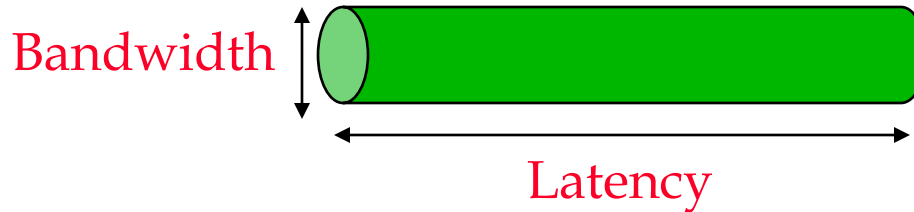
Simple Network Model

- Network is a pipe connecting two computers



- Basic Metrics
 - Bandwidth, latency, overhead, error rate, and message size

Performance Metrics



- Bandwidth: number of bits transmitted per unit of time
- Latency = Propagation + Transmit + Queue
 - Propagation = Distance/SpeedOfLight(*)
 - Transmit = 1 bit/Bandwidth
- Overhead
 - # secs for CPU to put message on wire
- Error rate
 - Probability P that message will not arrive intact

Bandwidth vs. Latency

1 Byte Object

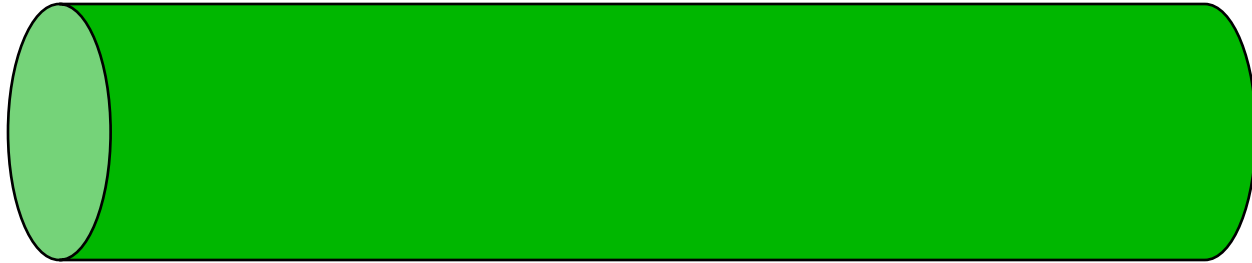
	Latency: 1 ms	Latency: 100 ms
Bandwidth: 1 Mbps	1,008 μ s	100,008 μ s
Bandwidth: 100 Mbps	1,000 μ s	100,000 μ s

10 MB Object

	Latency: 1 ms	Latency: 100 ms
Bandwidth: 1 Mbps	80.001 s	80.1 s
Bandwidth: 100 Mbps	.801 s	.9 s

⇒ Desirable network qualities based on application semantics

What Type of Pipe Do You Want?

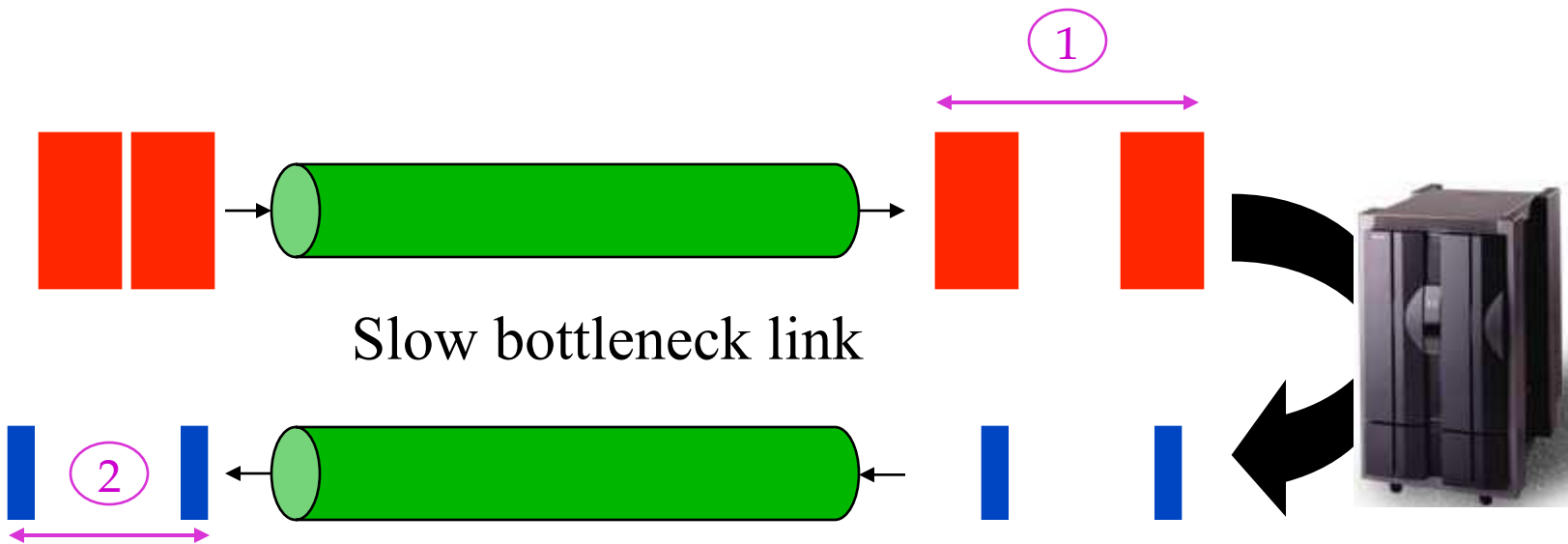


Performance Metrics Confusions

- Mega versus Mega, Kilo versus Kilo
 - Computer architecture: Mega $\rightarrow 2^{20}$, Kilo $\rightarrow 2^{10}$
 - Computer networks: Mega $\rightarrow 10^6$, Kilo $\rightarrow 10^3$
- Mbps versus MBps
 - Networks: typically megabits per second
 - Architecture: typically megabytes per second
- Bandwidth versus throughput
 - Bandwidth: available over link
 - Throughput: available to application

How to Measure Bandwidth?

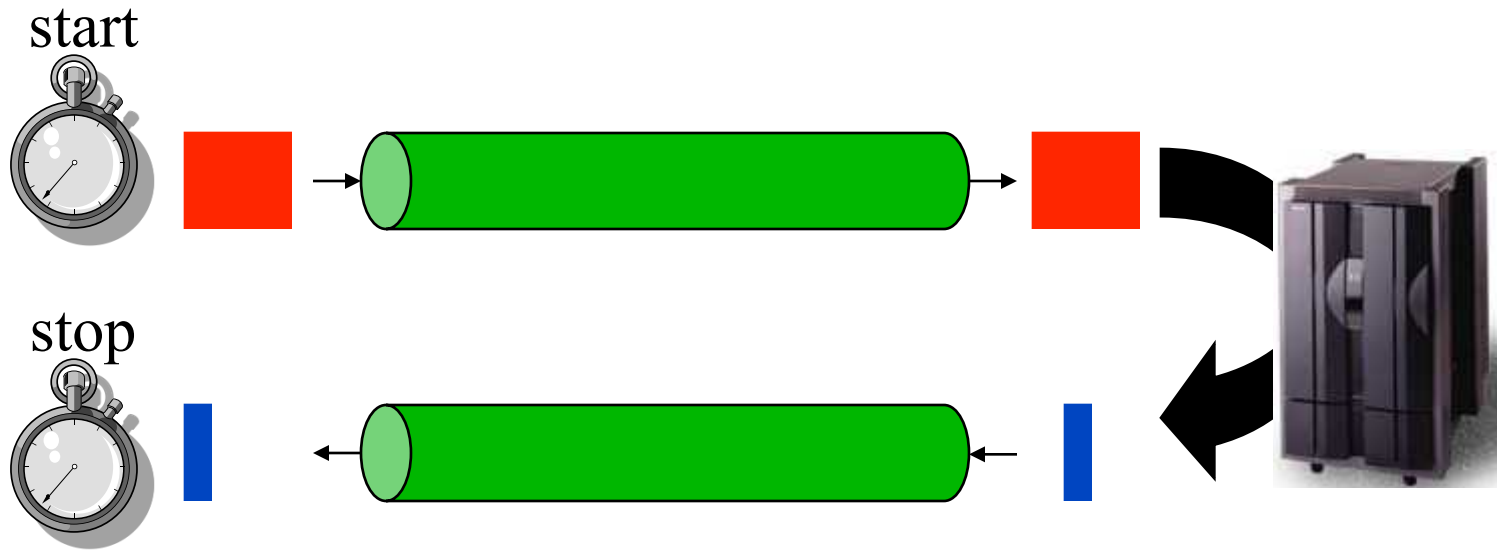
1) Measure total time to receive all packets at receiver



2) Measure how slow link increases gap between ACK' s

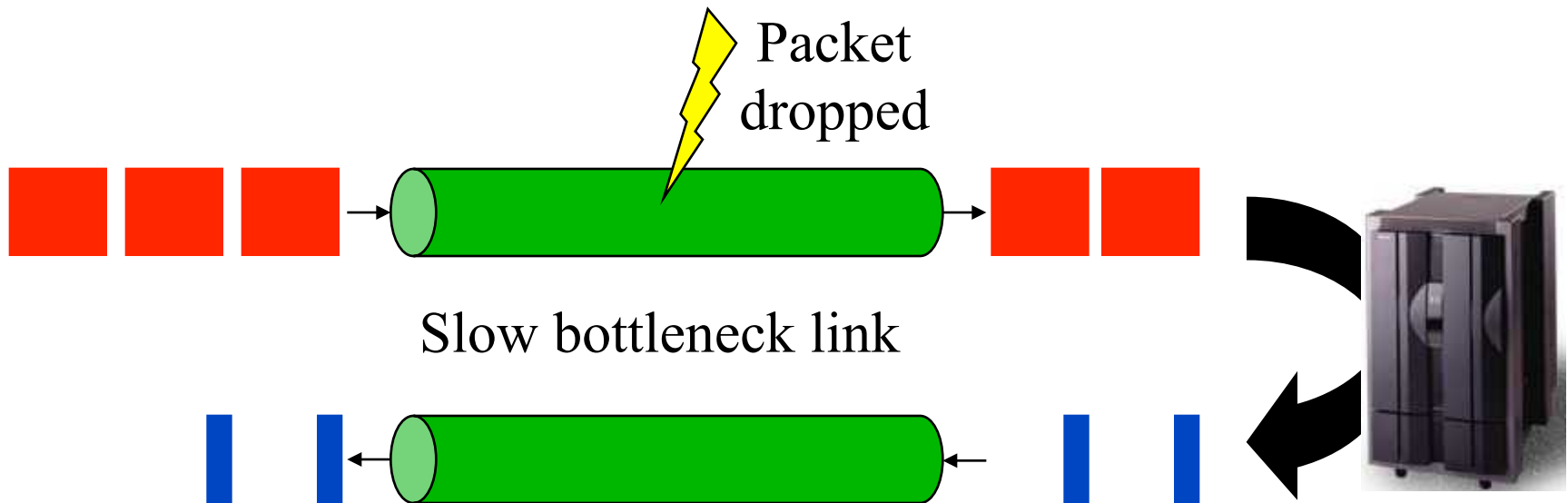
How to Measure Latency?

Measure round-trip time (assumes symmetric links)



How to Measure Error Rate?

Measure number of packets acknowledged



Google Cloud Demo

A large indoor recreational facility with a yellow curved ceiling, blue and red structural elements, and various games including ping pong, pool, and foosball. The space is brightly lit with modern lighting fixtures. In the foreground, there are two blue ping pong tables. To the right, a pool table is visible. In the background, there are foosball tables and a large mural on the wall. The ceiling features a prominent yellow curved structure with integrated lighting. The overall design is modern and playful, suitable for a family entertainment center or a community hub.

Europe Data Centers



Dublin



Hamina, Finland



St. Ghislain,
Belgium