

Thinking about networks

Ankit Singla

ETH Zürich Spring 2017

This lecture ...

- What do we want from the network?
- How do we achieve these goals?
- Intro: “Data center traffic & applications”
 - Reading 1: Inside the Social Network’s (Datacenter) Network
 - Reading 2: Why the Internet only just works

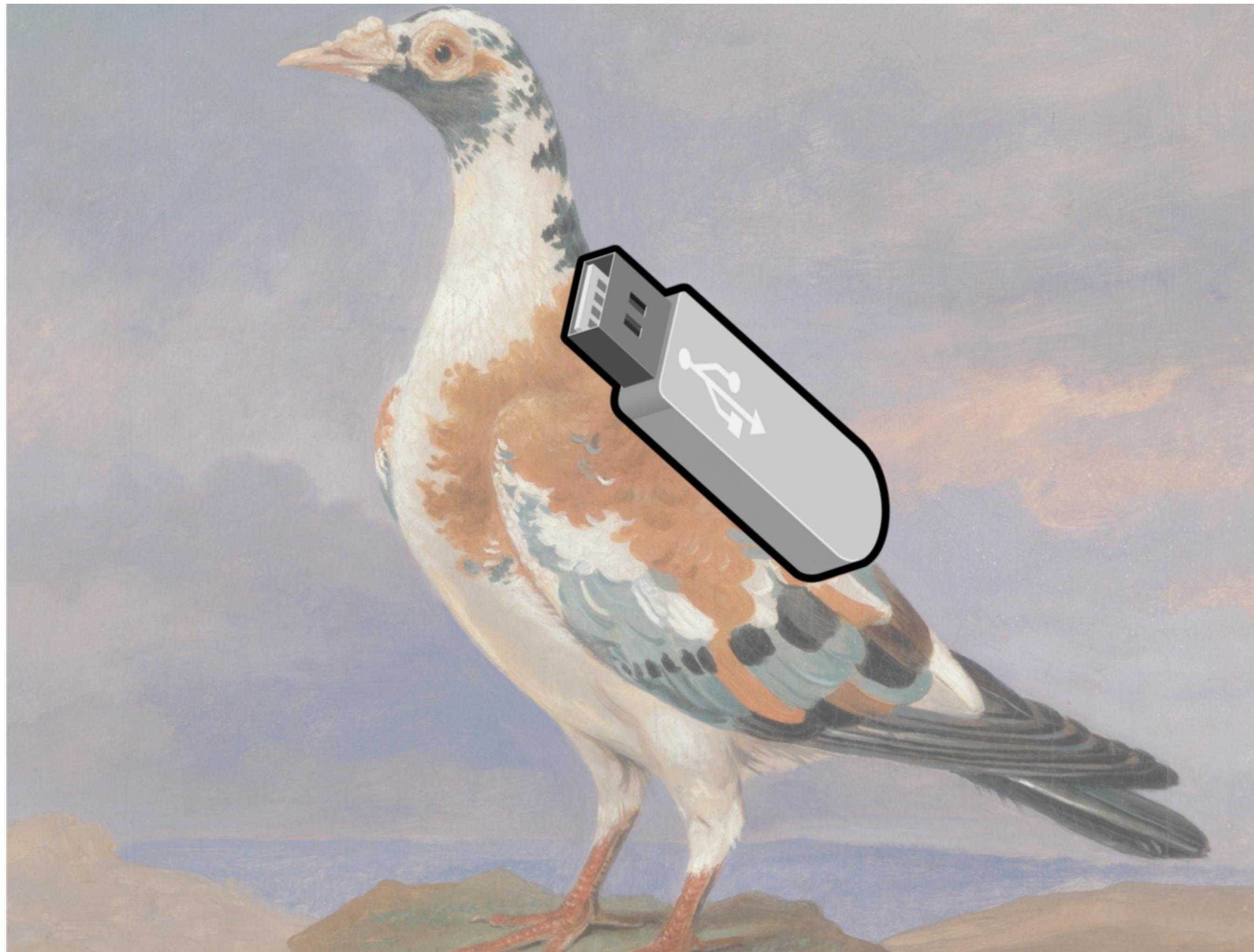
What do we want from the network?

- Performance: latency? bandwidth?
- Reliability / availability / security?
- Flexibility, manageability?
- ...

Network performance

Performance = bandwidth?

To get many megabits-per-second ...



[Pearson Scott Foresman]

To get many megabits-per-second ...



4400 km

80 Km / hour

1 TB USB stick

= 40 Mbps

... but 55 hours

Performance = latency?

By David A. Patterson

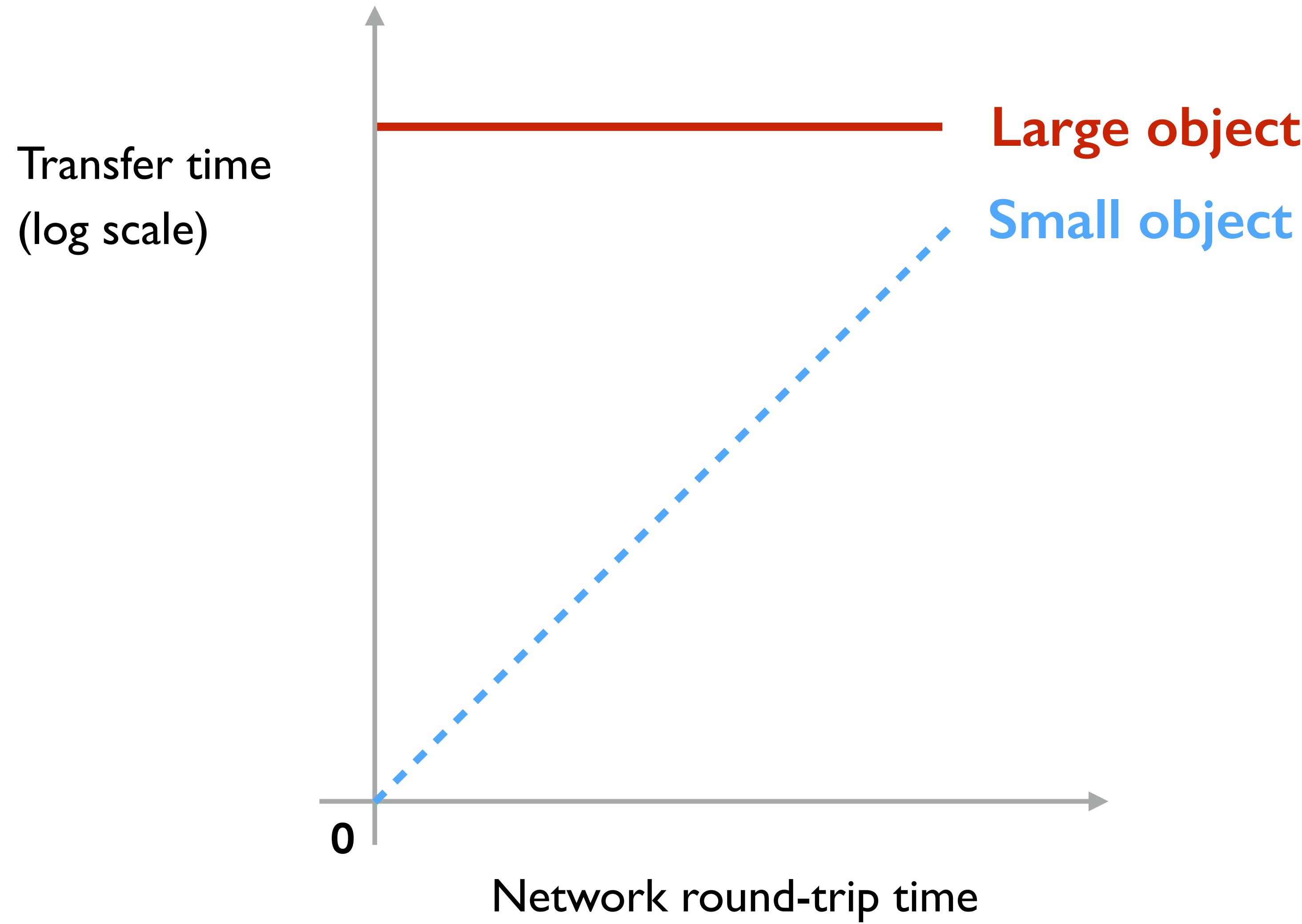
LATENCY LAGS BANDWIDTH

Recognizing the chronic imbalance between bandwidth and latency, and how to cope with it.

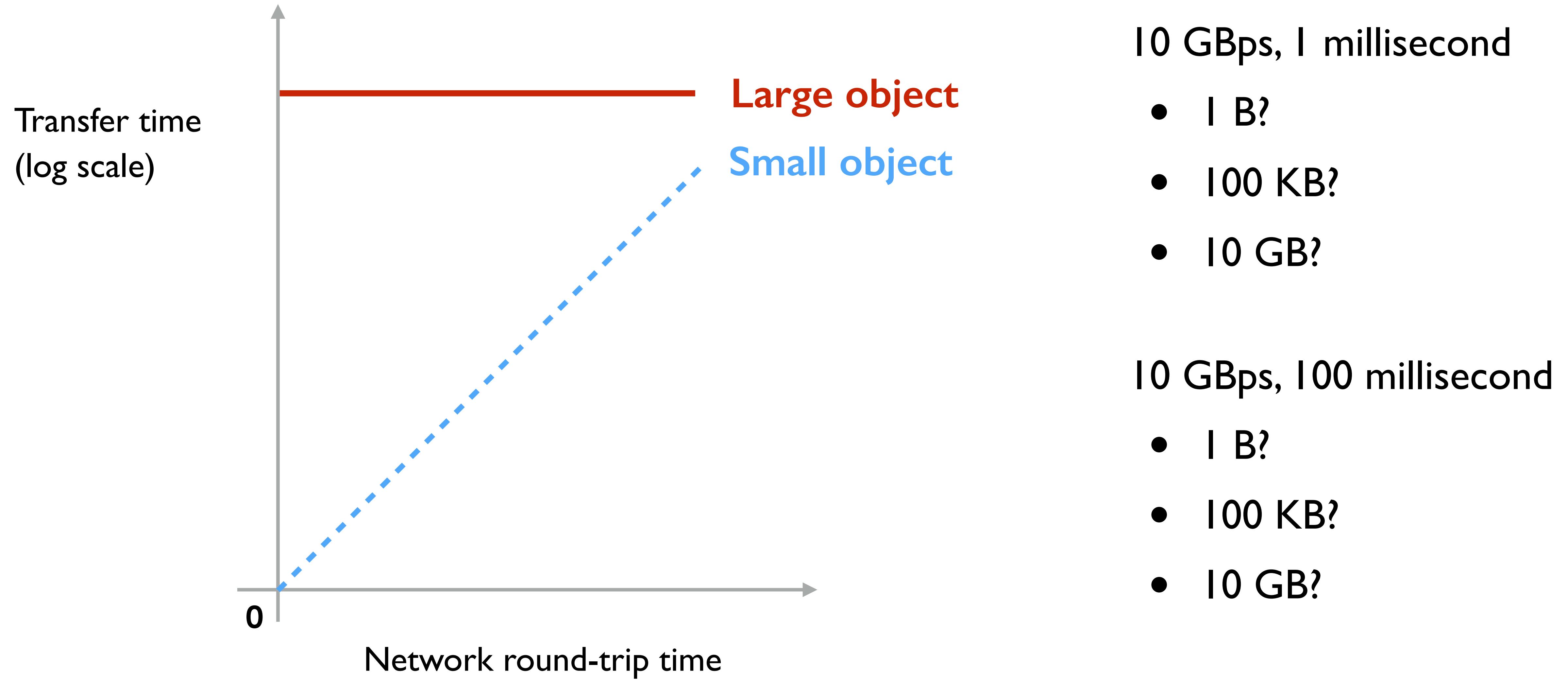


Enlightenment?!

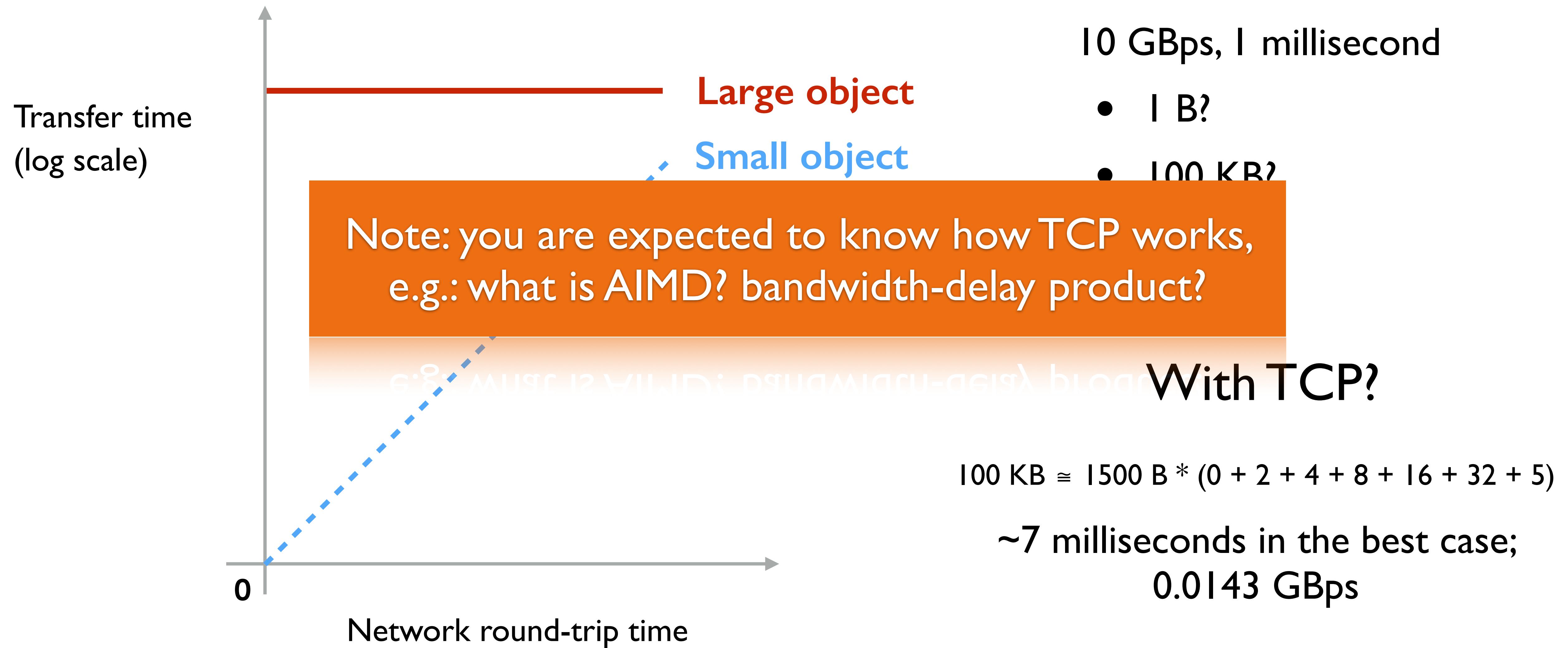
Bandwidth and latency



Bandwidth and latency



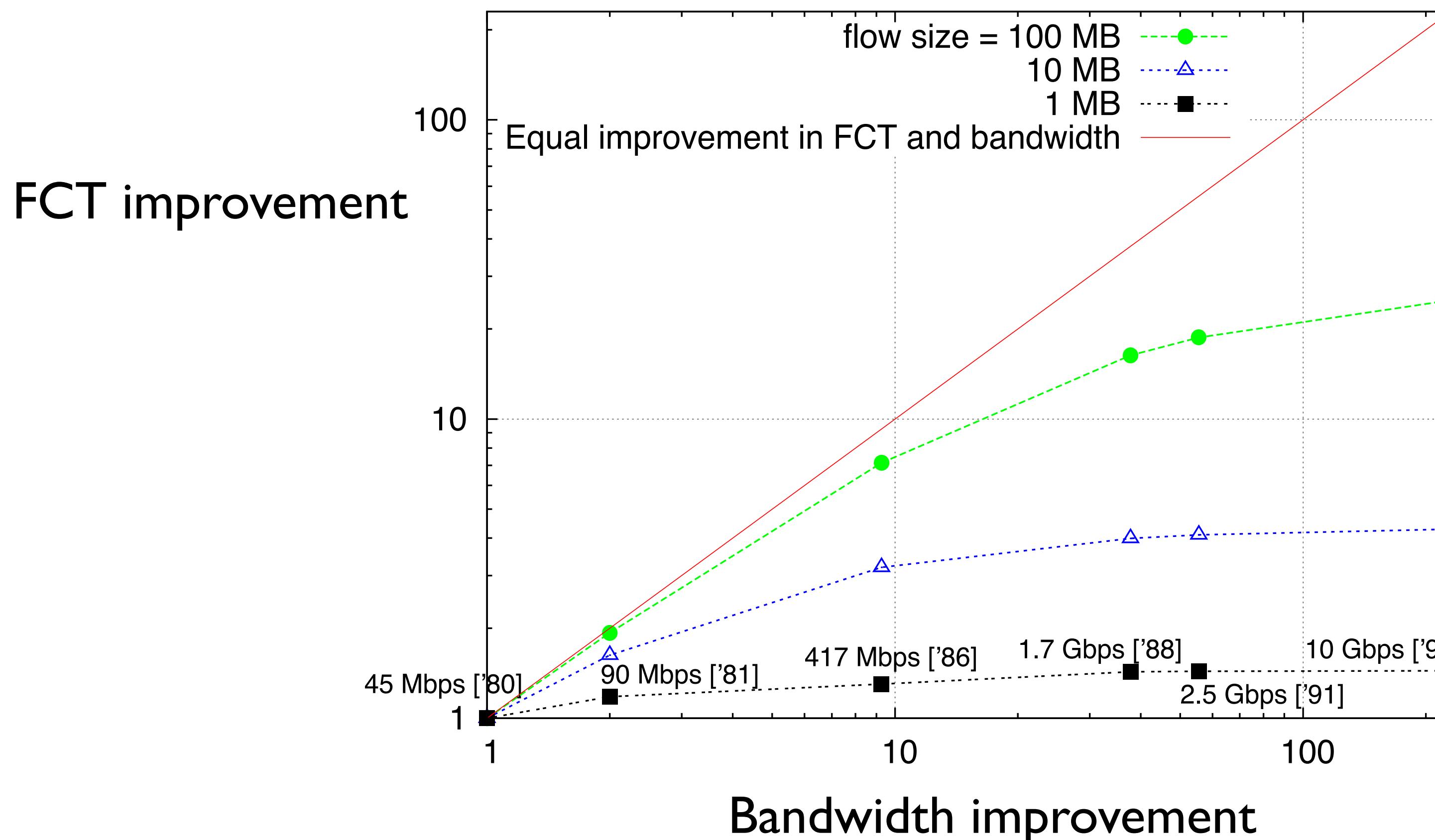
Bandwidth and latency



Why Flow-Completion Time is the Right Metric for Congestion Control

Nandita Dukkipati
Computer Systems Laboratory
Stanford University
Stanford, CA 94305-9025
nanditad@stanford.edu

Nick McKeown
Computer Systems Laboratory
Stanford University
Stanford, CA 94305-9025
nickm@stanford.edu



Beyond flow completion time?

- How long does <https://www.google.ch/?q=cool+stuff> take?
- What's the best video quality I can watch, without the “buffering”?
- How long does my Hadoop job take?
- ...

Also, we want *consistent, predictable* performance!

But what about fairness?!

Suppose a network is flow fair. How useful is that?

ACM CCR, 2007

Flow Rate Fairness: Dismantling a Religion

Bob Briscoe
BT Research & UCL
bob.briscoe@bt.com

*“Both the thing being allocated (rate) and what it is allocated among (flows) are **completely daft**—both unrealistic and impractical.”*

Food for thought ...

How to ...

- ... translate micro-benchmarks to app-level metrics?
- ... make service providers accountable?
- ... improve the performance of systems?
- ... approach fairness?

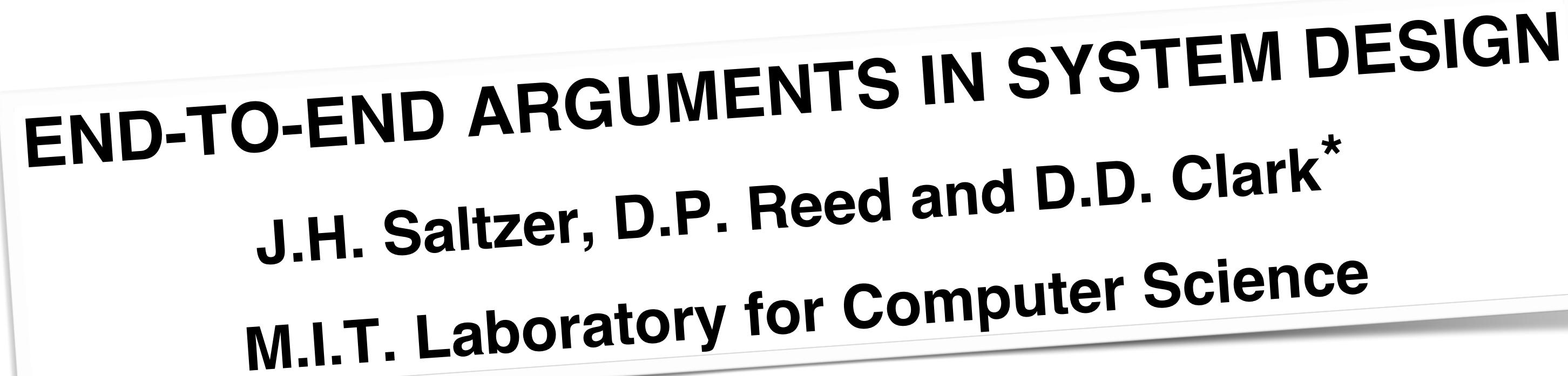
Reliability

Two particular considerations with reliability ...

- The end-to-end argument
- The fate-sharing principle

The end to end principle

IEEE ICDCS, 1981



“the end-to-end check of the file transfer application must still be implemented no matter how reliable the communication system becomes”

Does every application build its own reliability?



This is (partly) why we have:
modularity, layering and indirection!



Food for thought ...

Where should we implement ...

- Congestion control?
- Routing?
- Firewalls?
- ...

Food for thought ...

- When do we break layering?
- How do we define “end”?
 - TCP vs. application?
 - iPhone vs. iWatch?
 - Distributed applications?
- When do we break the end-to-end argument?

Survivability

End-points should be able to continue communicating without resetting conversation, even under failures.

Where should you store connection state?

“Fate-sharing principle”

Flexibility

TCP/IP vs. Everything/IP

Initially, TCP/IP was a monolith. Why did we split?

What if an application doesn't want reliability?

This is another reason we have:
modularity, layering and indirection!

Weekly reading guide

Data center applications and traffic

ACM SIGCOMM, 2015

Inside the Social Network's (Datacenter) Network

Arjun Roy, Hongyi Zeng[†], Jasmeet Bagga[†], George Porter, and Alex C. Snoeren

- Only one data point
- How did they collect measurements?
- Workload + management = observations?
- Data is available to play with; link on course-page

A perspective on networking ...

BT Technology Journal, 2006

Why the Internet only just works

M Handley

- Light reading (?)
- 10 years old, what has changed?

Read the other papers too, if you can!