Prudentia: a Watchdog for Internet Fairness

Justine Sherry
Assistant Professor
Carnegie Mellon University

sherry@cs.cmu.edu
@justinesherry

Carnegie Mellon University Computer Science Department

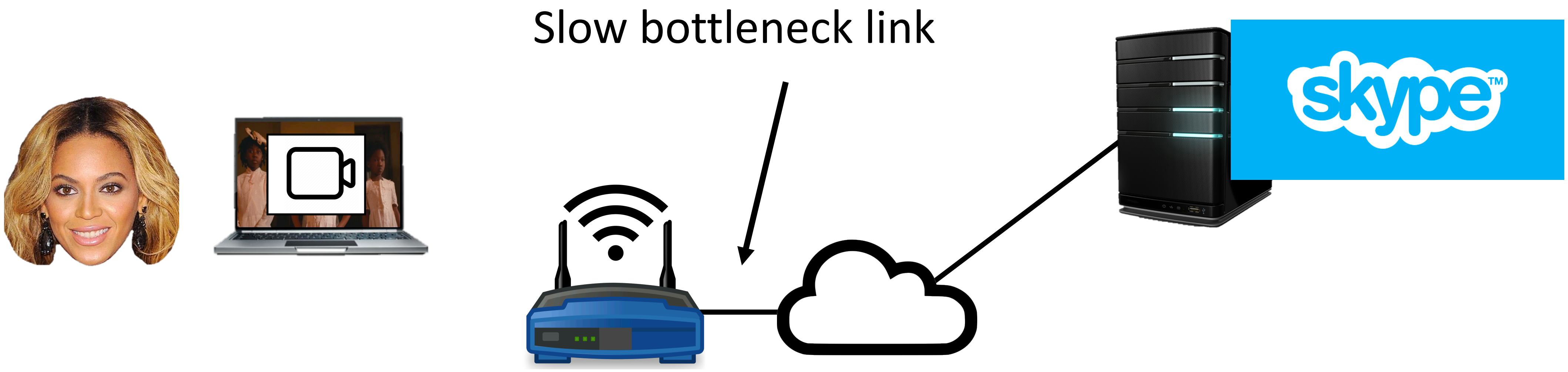
Carnegie Mellon University Security and Privacy Institute CyLab

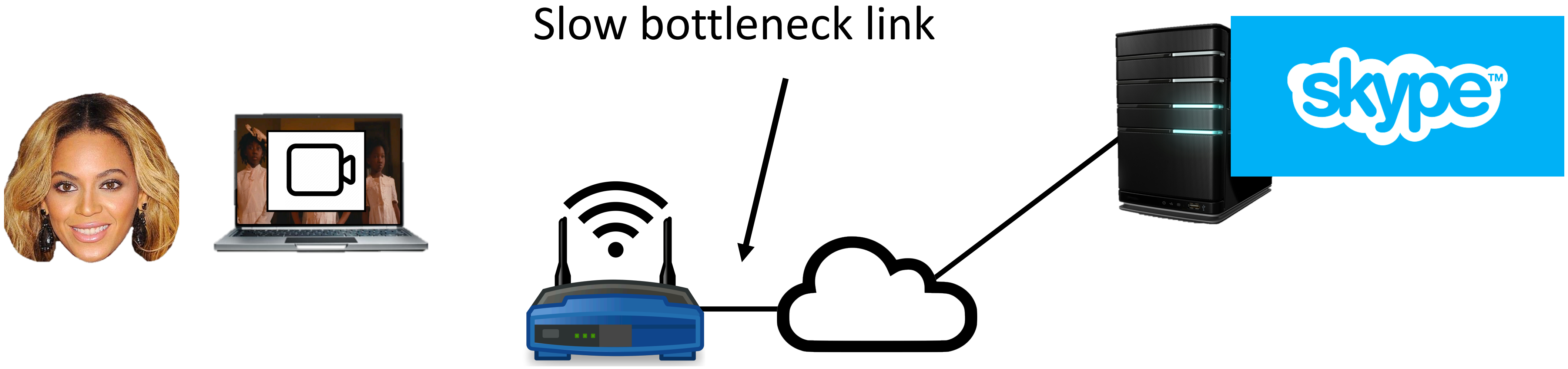
Isolation is a fundamental security primitive for computer systems.

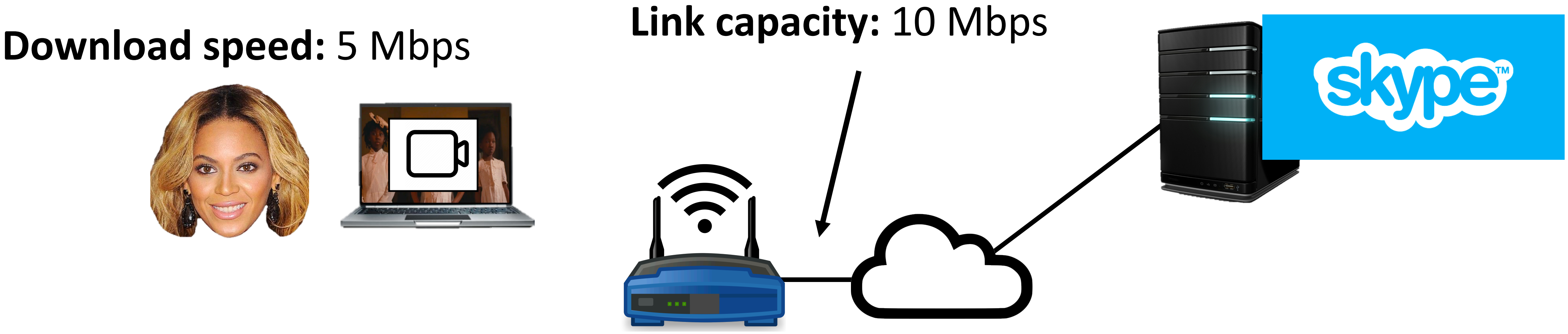
Performance Isolation: when many users *share* a computer system, a malicious or buggy user should not be able to consume system resources set aside for other users

- Processor Sharing
- Disk Partitions
- Time Division Multiplexing

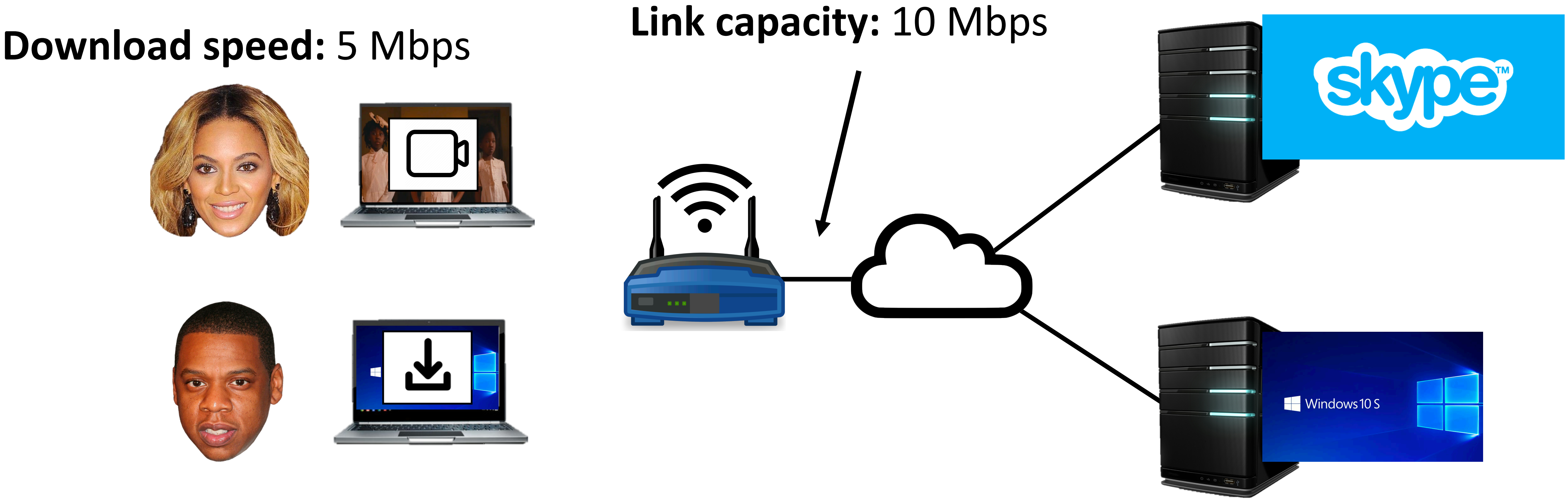
The Internet was deployed with little thought for isolation between users ☹

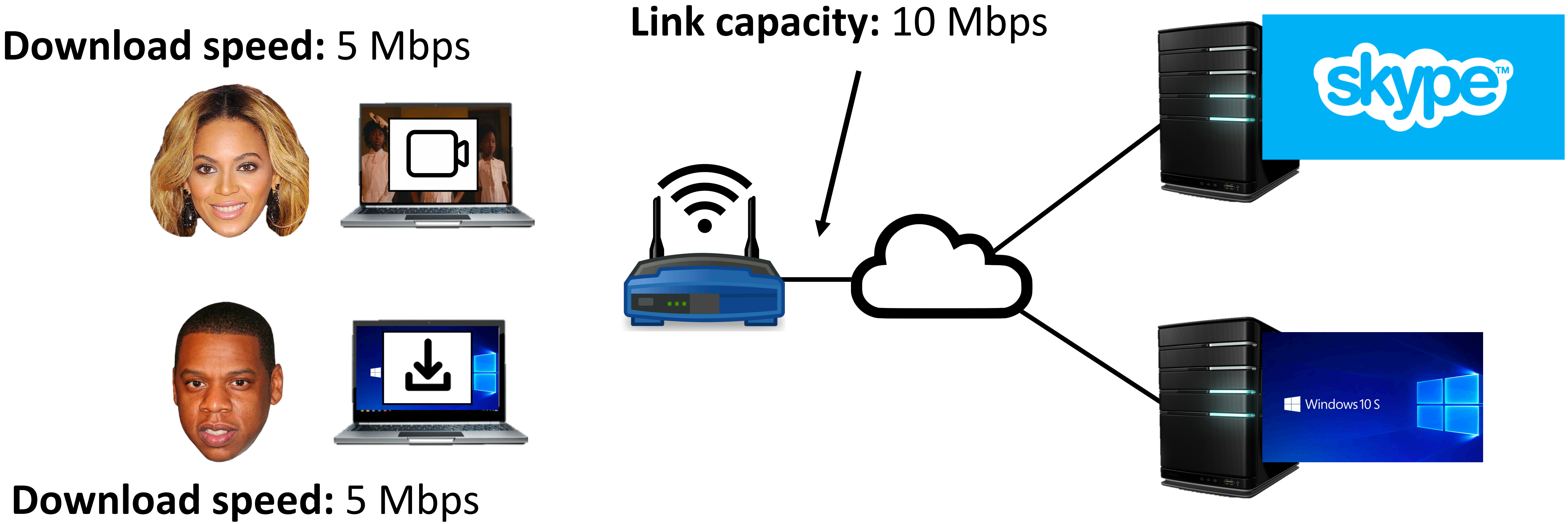

Slow bottleneck link
skype

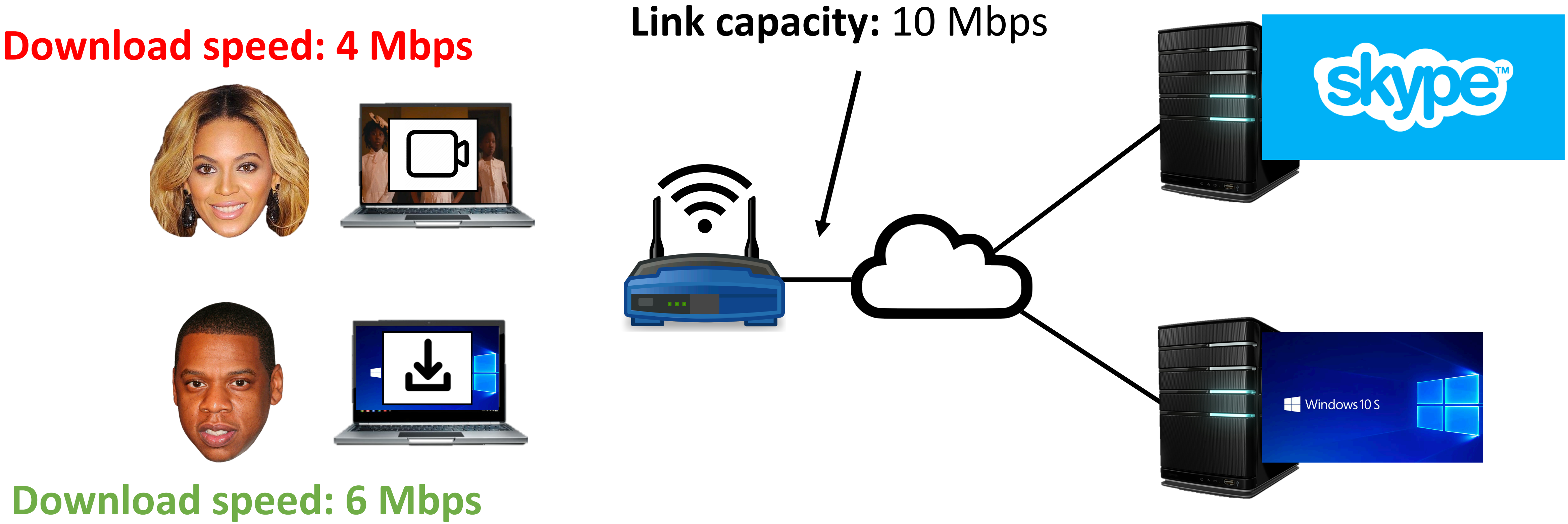

Slow bottleneck link
skype

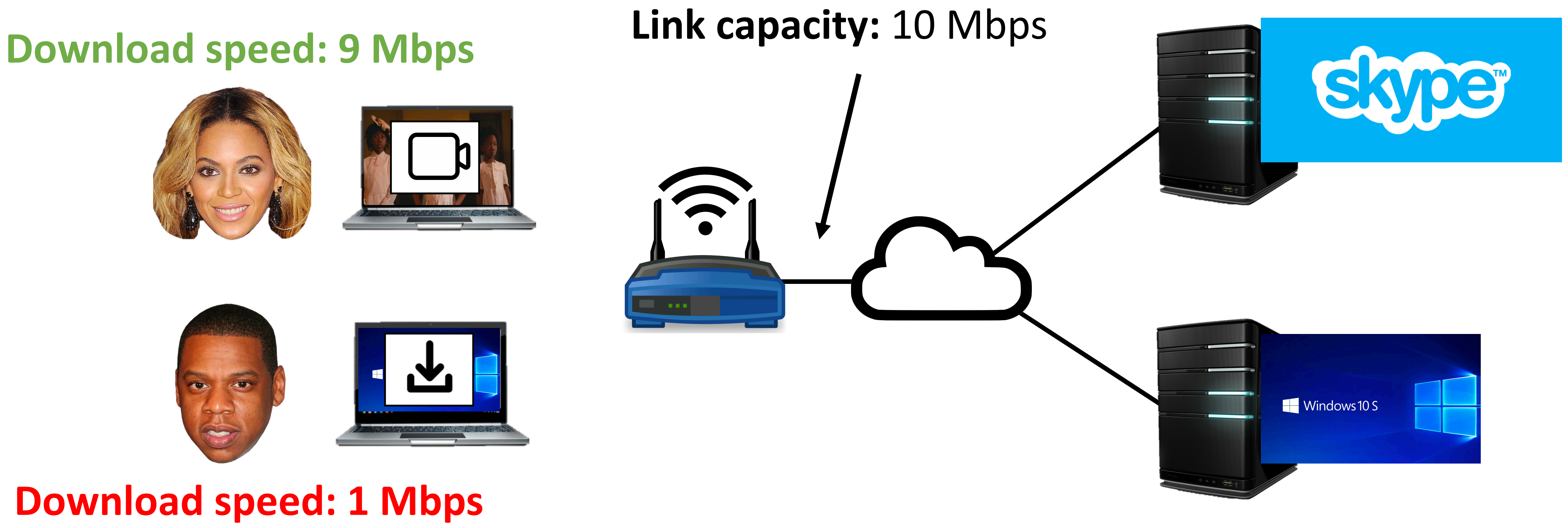


Download speed: 5 Mbps
Link capacity: 10 Mbps
skype

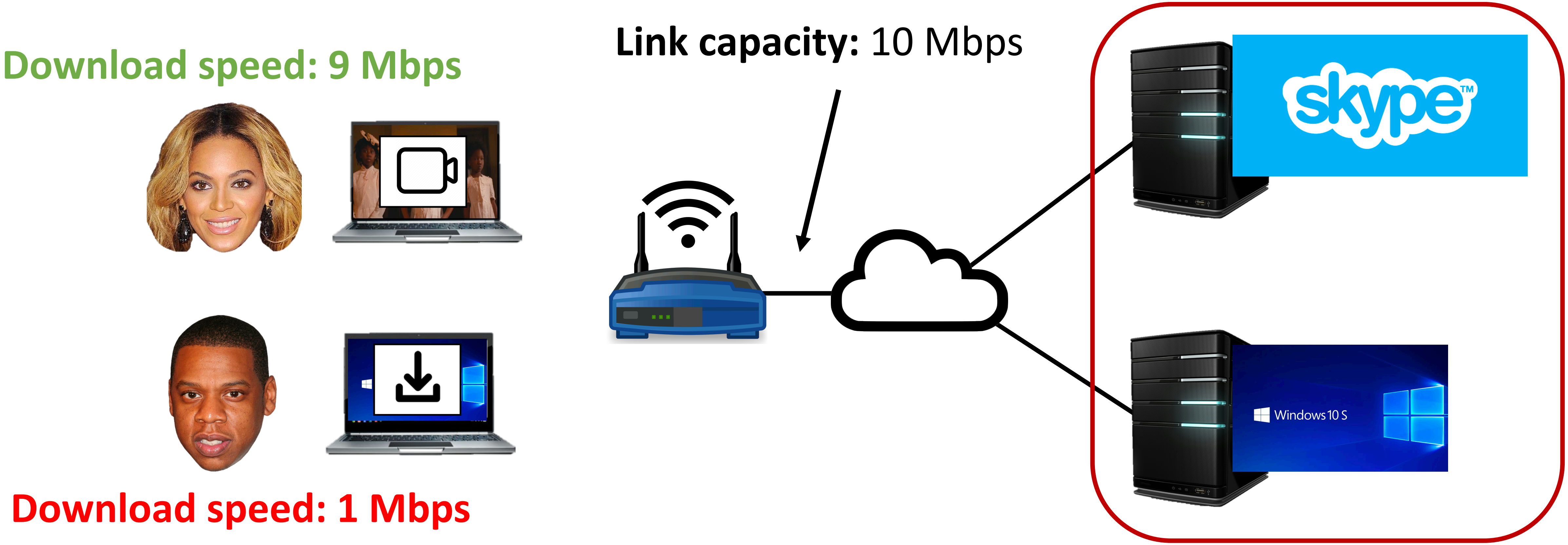

Download speed: 5 Mbps
Link capacity: 10 Mbps
skype
Windows 10 S


Download speed: 5 Mbps
Link capacity: 10 Mbps
skype
Windows 10 S
Download speed: 5 Mbps


Download speed: 4 Mbps
Link capacity: 10 Mbps
skype
Windows 10 S
Download speed: 6 Mbps


Download speed: 9 Mbps
Link capacity: 10 Mbps
skype™
Windows 10 S
Download speed: 1 Mbps

Congestion Control Algorithm: code that runs at the sender to decide when to speed up transmission and when to slow down

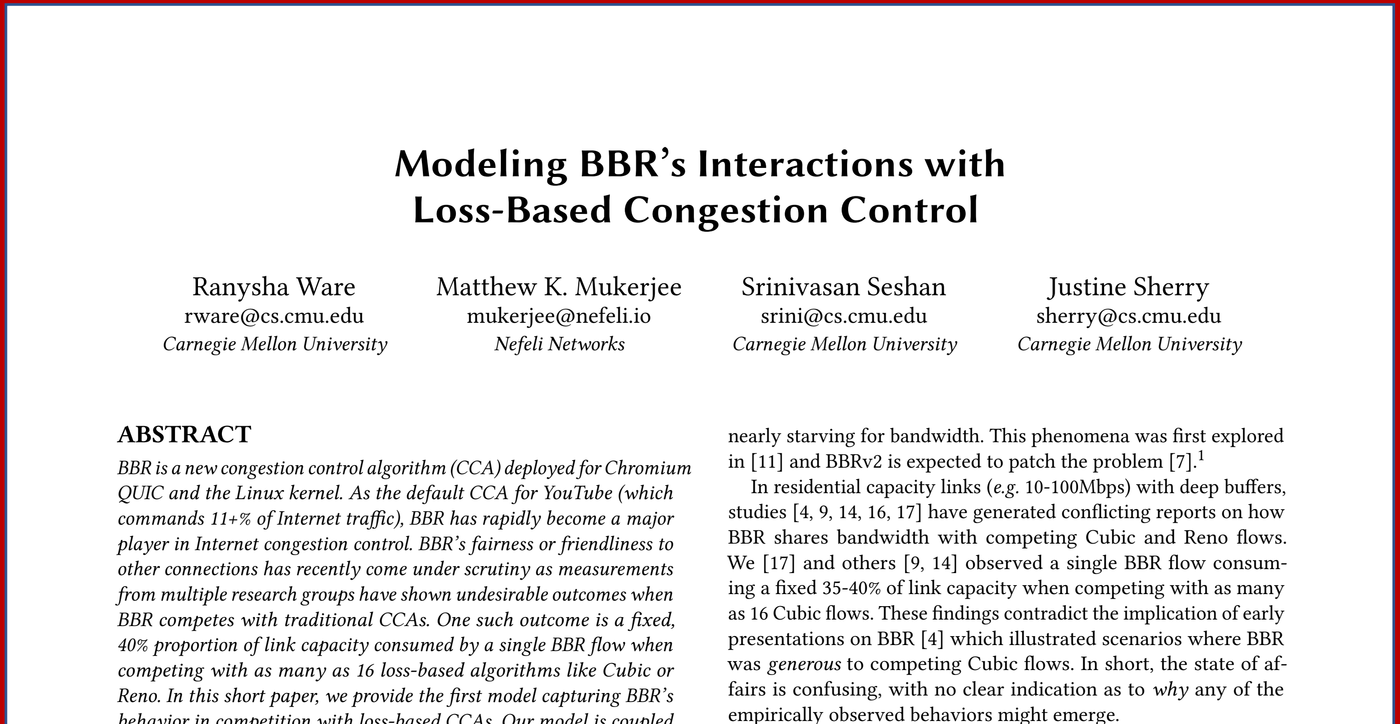


Congestion Control Algorithm: code that runs at the sender to decide when to speed up transmission and when to slow down

New Reno	Cubic	BBR
Copa	PCC Vivace	Compound

Unfair outcomes can occur when different congestion control algorithms compete.

Unfair outcomes can occur when different congestion control algorithms compete.



Modeling BBR's Interactions with Loss-Based Congestion Control

Ranysha Ware
rware@cs.cmu.edu
Carnegie Mellon University

Matthew K. Mukerjee
mukerjee@nefeli.io
Nefeli Networks

Srinivasan Seshan
srini@cs.cmu.edu
Carnegie Mellon University

Justine Sherry
sherry@cs.cmu.edu
Carnegie Mellon University

ABSTRACT

BBR is a new congestion control algorithm (CCA) deployed for Chromium QUIC and the Linux kernel. As the default CCA for YouTube (which commands 11+% of Internet traffic), BBR has rapidly become a major player in Internet congestion control. BBR's fairness or friendliness to other connections has recently come under scrutiny as measurements from multiple research groups have shown undesirable outcomes when BBR competes with traditional CCAs. One such outcome is a fixed, 40% proportion of link capacity consumed by a single BBR flow when competing with as many as 16 loss-based algorithms like Cubic or Reno. In this short paper, we provide the first model capturing BBR's behavior in competition with loss-based CCAs. Our model is coupled

nearly starving for bandwidth. This phenomena was first explored in [11] and BBRv2 is expected to patch the problem [7].[1]

In residential capacity links (*e.g.* 10-100Mbps) with deep buffers, studies [4, 9, 14, 16, 17] have generated conflicting reports on how BBR shares bandwidth with competing Cubic and Reno flows. We [17] and others [9, 14] observed a single BBR flow consuming a fixed 35-40% of link capacity when competing with as many as 16 Cubic flows. These findings contradict the implication of early presentations on BBR [4] which illustrated scenarios where BBR was *generous* to competing Cubic flows. In short, the state of affairs is confusing, with no clear indication as to *why* any of the empirically observed behaviors might emerge.

BBR Congestion-Based Congestion Control

MEASURING BOTTLENECK BANDWIDTH AND ROUND-TRIP PROPAGATION TIME

NEAL CARDWELL
YUCHUNG CHENG
C. STEPHEN GUNN
SOHEIL HASSAS YEGANEH
VAN JACOBSON

By all accounts, today's Internet is not moving data as well as it should. Most of the world's cellular users experience delays of seconds to minutes; public Wi-Fi in airports and conference venues is often worse. Physics and climate researchers need to exchange petabytes of data with global collaborators but find their carefully engineered multi-Gbps infrastructure often delivers at only a few Mbps over intercontinental distances.[6]

These problems result from a design choice made when TCP congestion control was created in the 1980s—interpreting packet loss as "congestion."[13] This equivalence was true at the time but was because of technology limitations, not first principles. As NICs (network interface controllers) evolved from Mbps to Gbps and memory chips from KB to GB, the relationship between packet loss and congestion became more tenuous.

Today TCP's loss-based congestion control—even with the current best of breed, CUBIC[11]—is the primary cause of these problems. When bottleneck buffers are large,

In 2016, Google released a new congestion control algorithm called **BBR**.

They open-sourced the algorithm, and deployed it as the default CCA for YouTube

Early measurement studies suggested that BBR was **generous** to traditional, widely-deployed algorithms like Cubic and Reno.

BBR is fair to Cubic in deep-buffered networks.

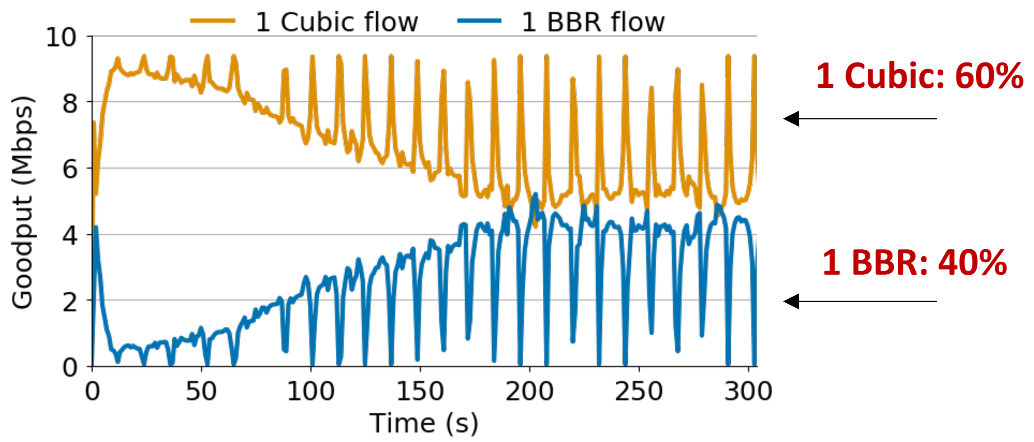


Figure: 1 BBR vs. 1 Cubic.
(10 Mbps network, 32 BDP queue)

Reference: N. Cardwell, et.al. 2016. BBR: Congestion control. In Presentation at IETF97
.

But measurements in our testbed showed that sometimes BBR was quite **unfair** to traditional algorithms.

BBR is fair to Cubic in deep-buffered networks.

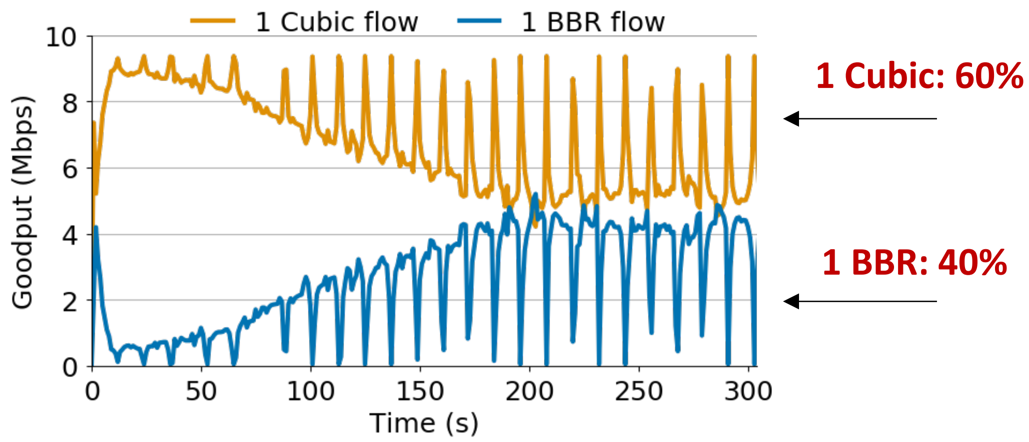


Figure: 1 BBR vs. 1 Cubic. (10 Mbps network, 32 BDP queue)

BBR is unfair to Cubic in deep-buffered networks.

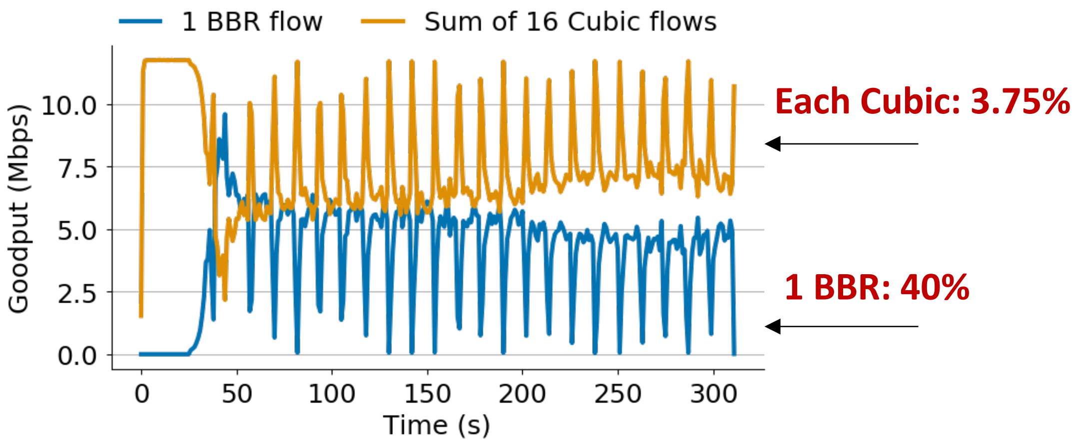


Figure: 1 BBR vs. 16 Cubic. (10 Mbps network, 32 BDP queue)

Reference: Ware et. al. The Battle for Bandwidth: Fairness and Heterogenous Congestion Control. Poster at NSDI 2018.

But measurements in our testbed showed that sometimes BBR was quite **unfair** to traditional algorithms.

BBR is fair to Cubic in deep-buffered networks.

BBR is unfair to Cubic in deep-buffered networks.

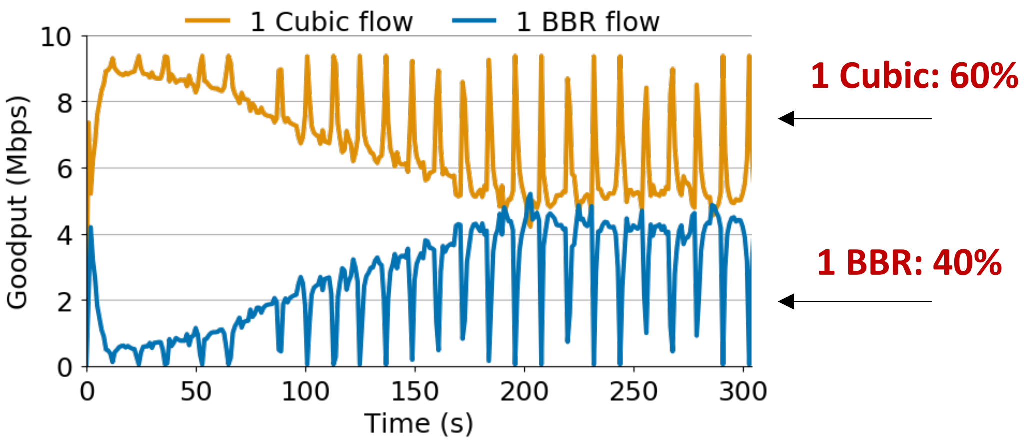


Figure: 1 BBR vs. 1 Cubic.
(10 Mbps network, 32 BDP queue)

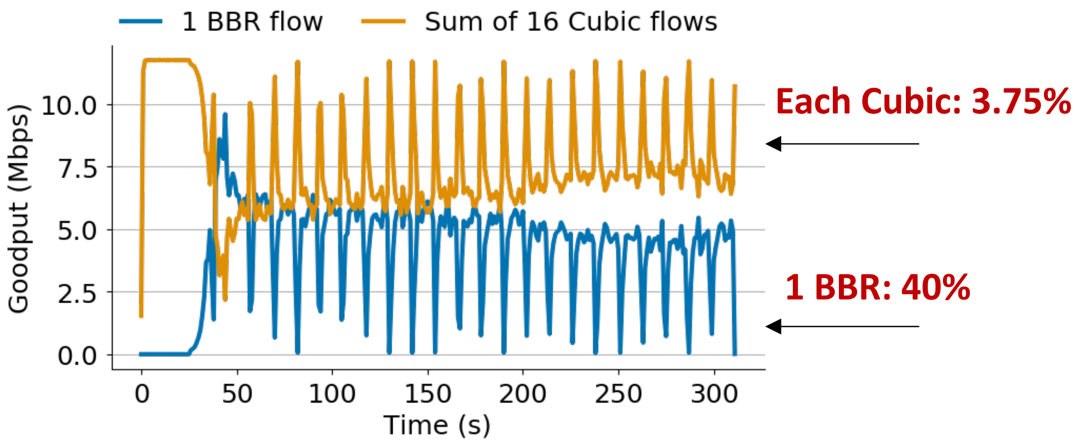


Figure: 1 BBR vs. 16 Cubic.
(10 Mbps network, 32 BDP queue)

How can we explain these results?

We derived the first **mathematical models** to understand BBR's behavior.

Mathis equation for TCP Reno's throughput

$$BW < \left(\frac{MSS}{RTT}\right)\frac{1}{\sqrt{p}}$$

Padhye equation for TCP Reno's throughput

$$B(p) \approx \min\left(\frac{W_{max}}{RTT}, \frac{1}{RTT\sqrt{\frac{2bp}{3}} + T_0 \min\left(1, 3\sqrt{\frac{3bp}{8}}\right) p(1+32p^2)}\right)$$

We derived the first **mathematical models** to understand BBR's behavior.

Mathis equation for TCP Reno's throughput

$$BW < \left(\frac{MSS}{RTT}\right)\frac{1}{\sqrt{p}}$$

Padhye equation for TCP Reno's throughput

$$B(p) \approx \min\left(\frac{W_{max}}{RTT}, \frac{1}{RTT\sqrt{\frac{2bp}{3}} + T_0 \min\left(1, 3\sqrt{\frac{3bp}{8}}\right) p(1+32p^2)}\right)$$

Our equation for BBR's throughput

Can we build a model?

We derived the first **mathematical models** to understand BBR's behavior.

Mathis equation for TCP Reno's throughput

$$BW < \left(\frac{MSS}{RTT}\right)\frac{1}{\sqrt{p}}$$

Padhye equation for TCP Reno's throughput

$$B(p) \approx \min\left(\frac{W_{max}}{RTT}, \frac{1}{RTT\sqrt{\frac{2bp}{3}} + T_0 \min\left(1, 3\sqrt{\frac{3bp}{8}}\right) p(1+32p^2)}\right)$$

Our equation for BBR's throughput

$$BBR_{frac} = \left(1 - \frac{1}{2} + \frac{1}{2X} + \frac{4N}{q}\right) \times \left(1 - \left(\frac{q}{c} + .2 + l\right) \times \frac{1}{10}\right)$$

Our model can predict BBR's throughput when competing against Cubic flows with a median error of 5% and against Reno with a median of 8%.

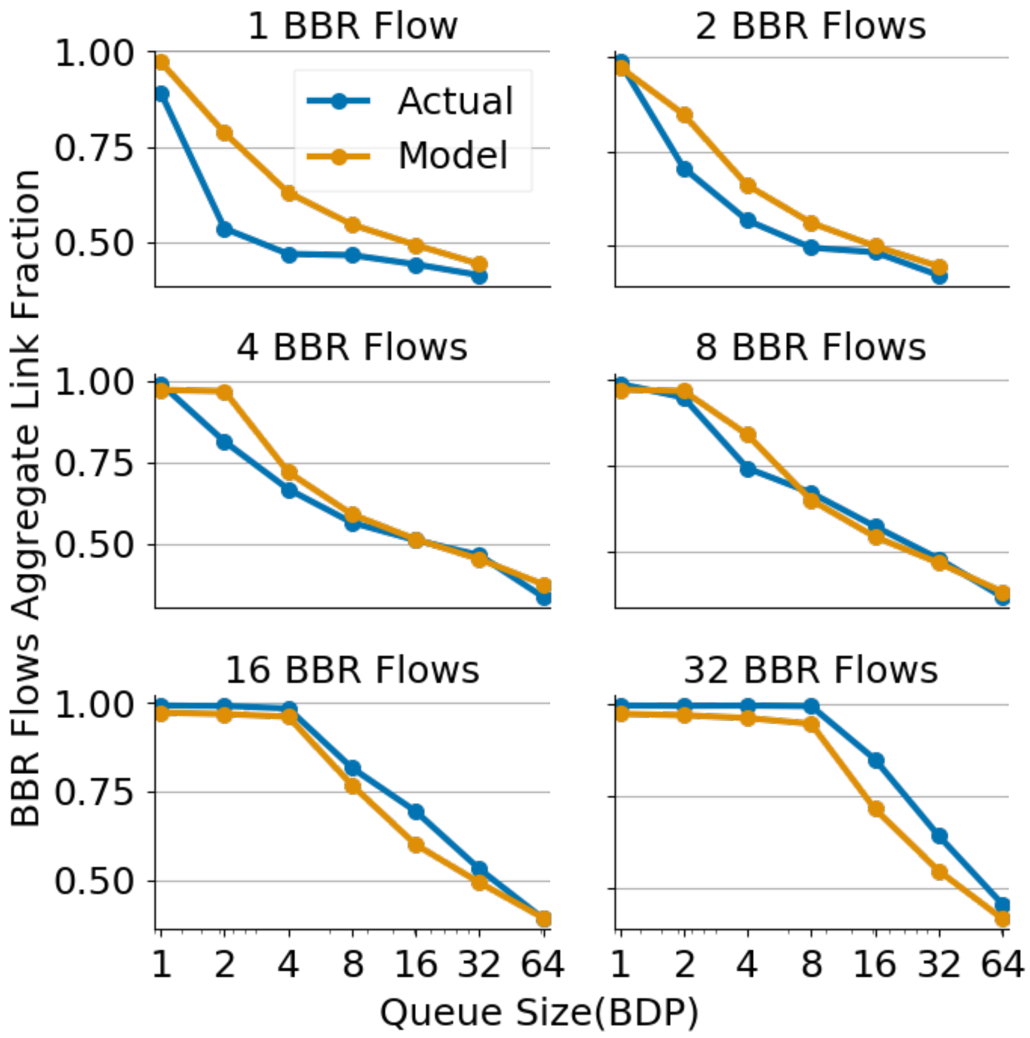


Figure: N BBR vs. 1 Cubic (10 Mbps network)

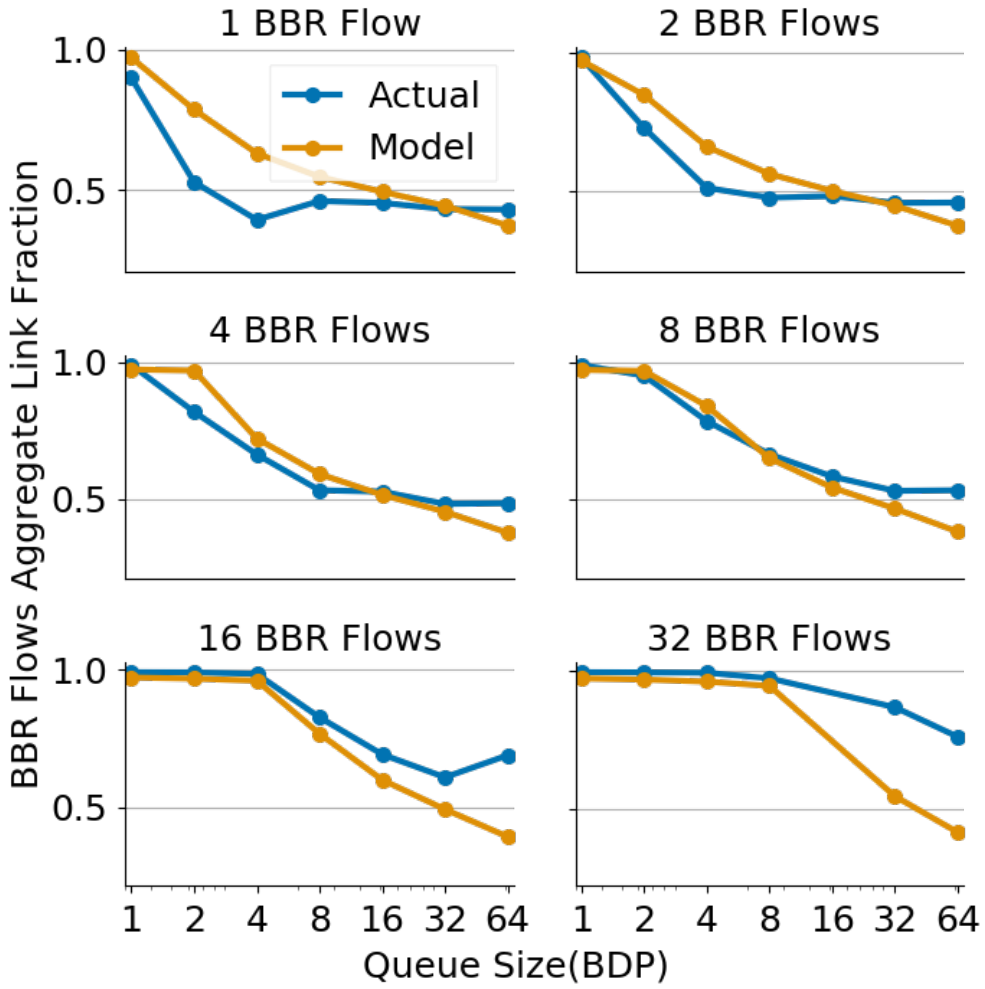


Figure: N BBR vs. 1 Reno (10 Mbps network)

Our model shows BBR's throughput **does not** depend on the number of competing loss-based flows.

Mathis equation for TCP Reno's throughput

$$BW < \left(\frac{MSS}{RTT}\right)\frac{1}{\sqrt{p}}$$

Padhye equation for TCP Reno's throughput

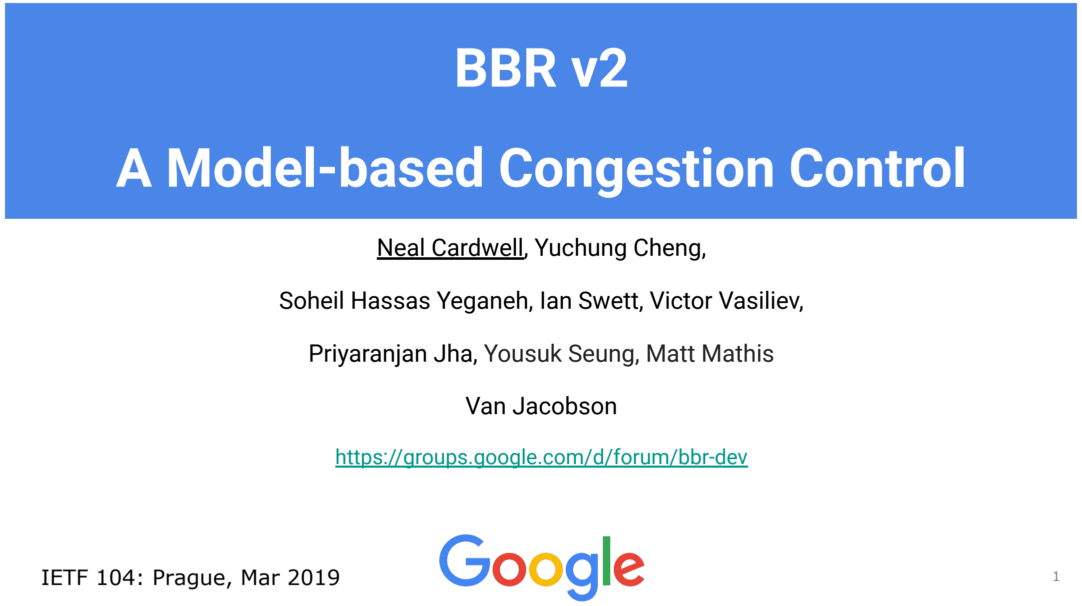
$$B(p) \approx \min\left(\frac{W_{max}}{RTT}, \frac{1}{RTT\sqrt{\frac{2bp}{3}} + T_0 \min\left(1, 3\sqrt{\frac{3bp}{8}}\right) p(1+32p^2)}\right)$$

Our equation for BBR's throughput

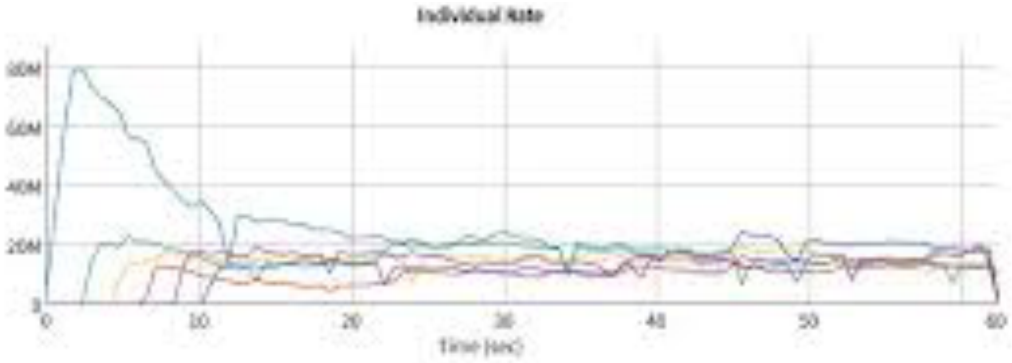
$$BBR_{frac} = \left(1 - \frac{1}{2} + \frac{1}{2X} + \frac{4N}{q}\right) \times \left(1 - \left(\frac{q}{c} + .2 + l\right) \times \frac{1}{10}\right)$$

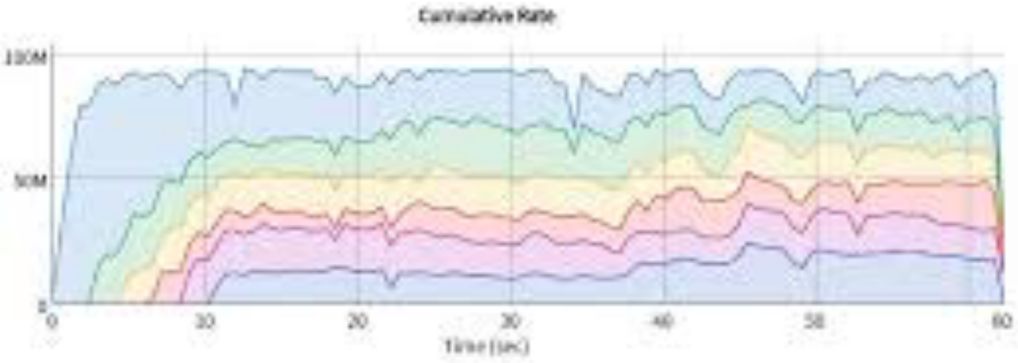
None of these variables depend on the number of loss-based flows!

Our insights informed the designers of BBR who came out with a new and improved version of BBR– in part designed to be **more fair** to competitors.



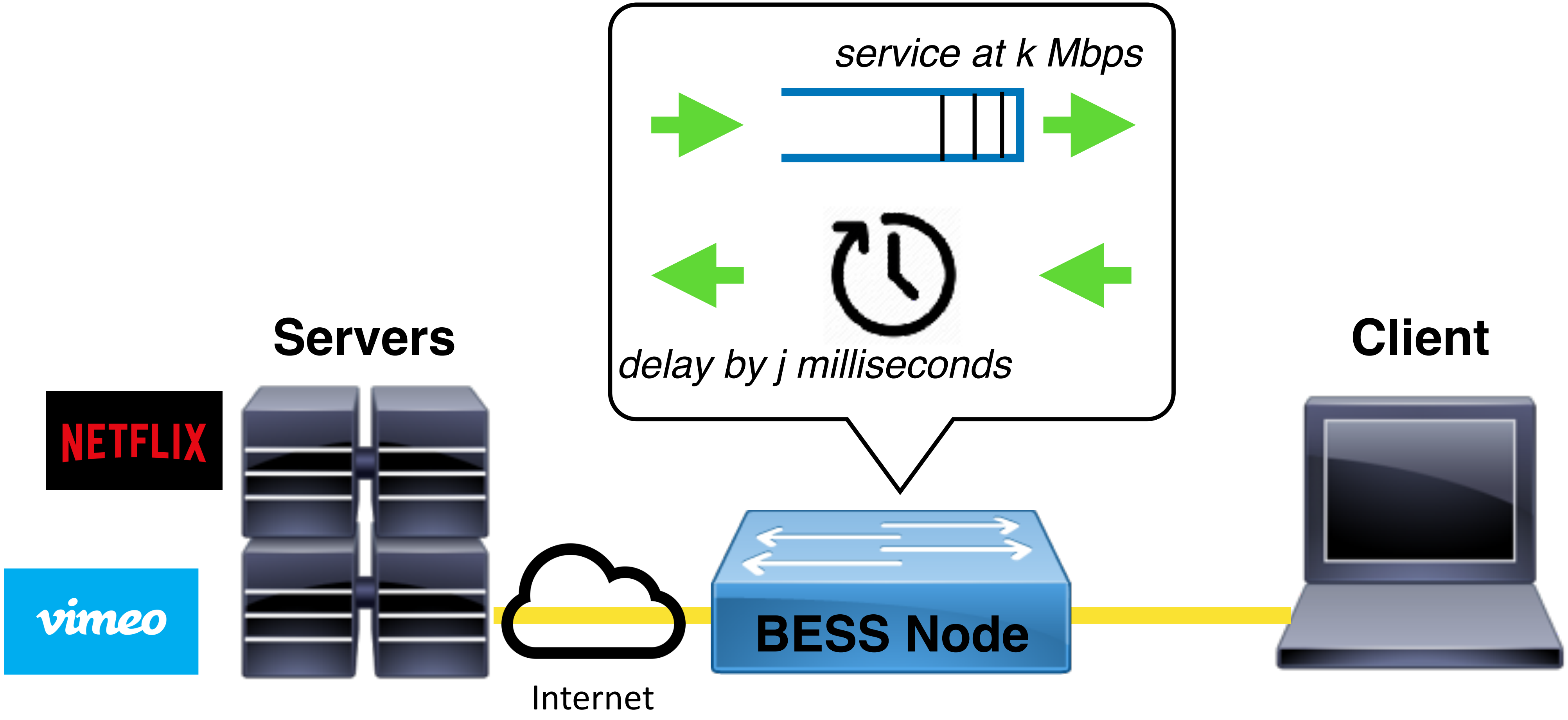
Hooray!





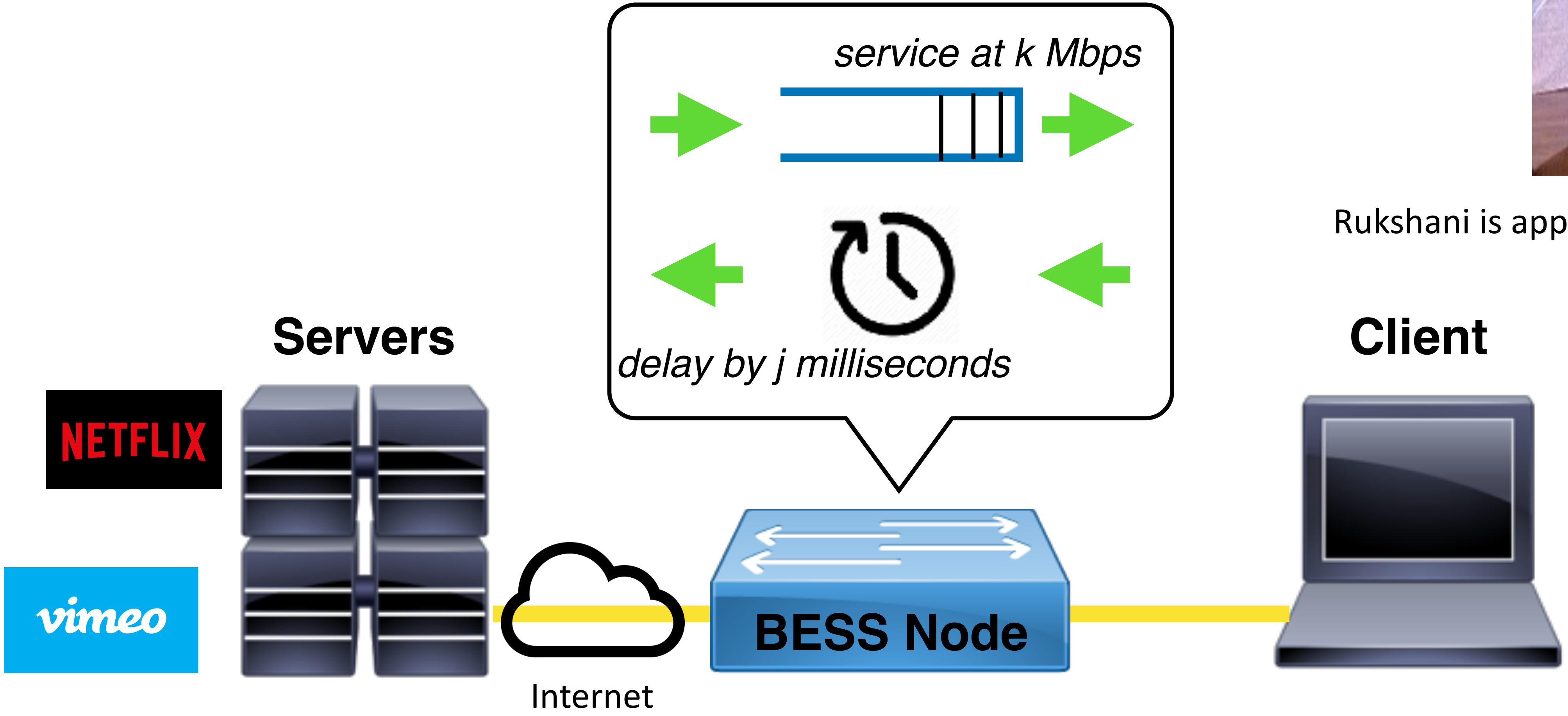
Except… what about novel CCAs and services that are **not** open sourced?

Prudentia: a testbed for measuring Internet services' fairness and the *harm* that competing services cause each other

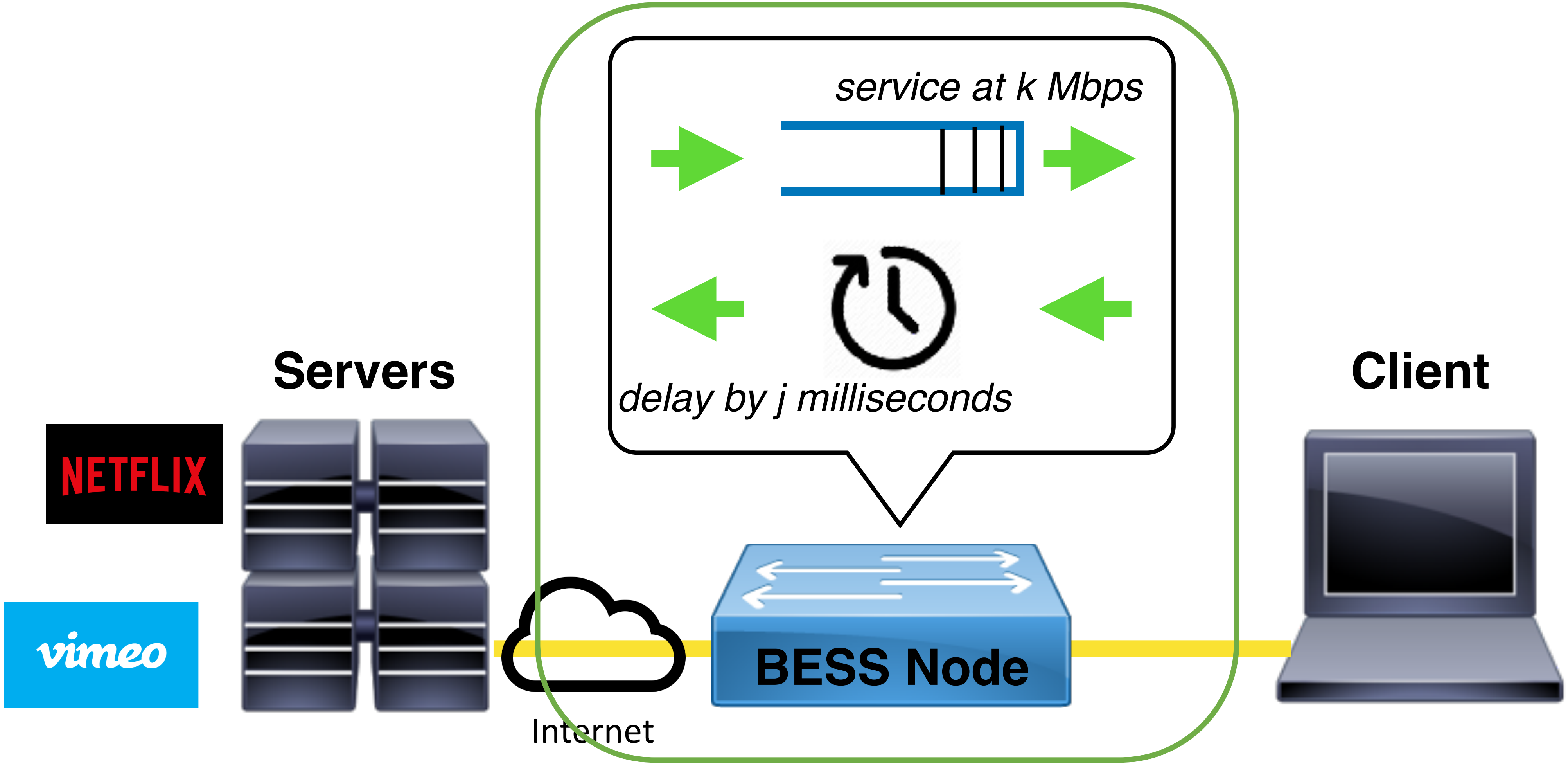


Prudentia: a testbed for measuring Internet services' fairness and the *harm* that competing services cause each other

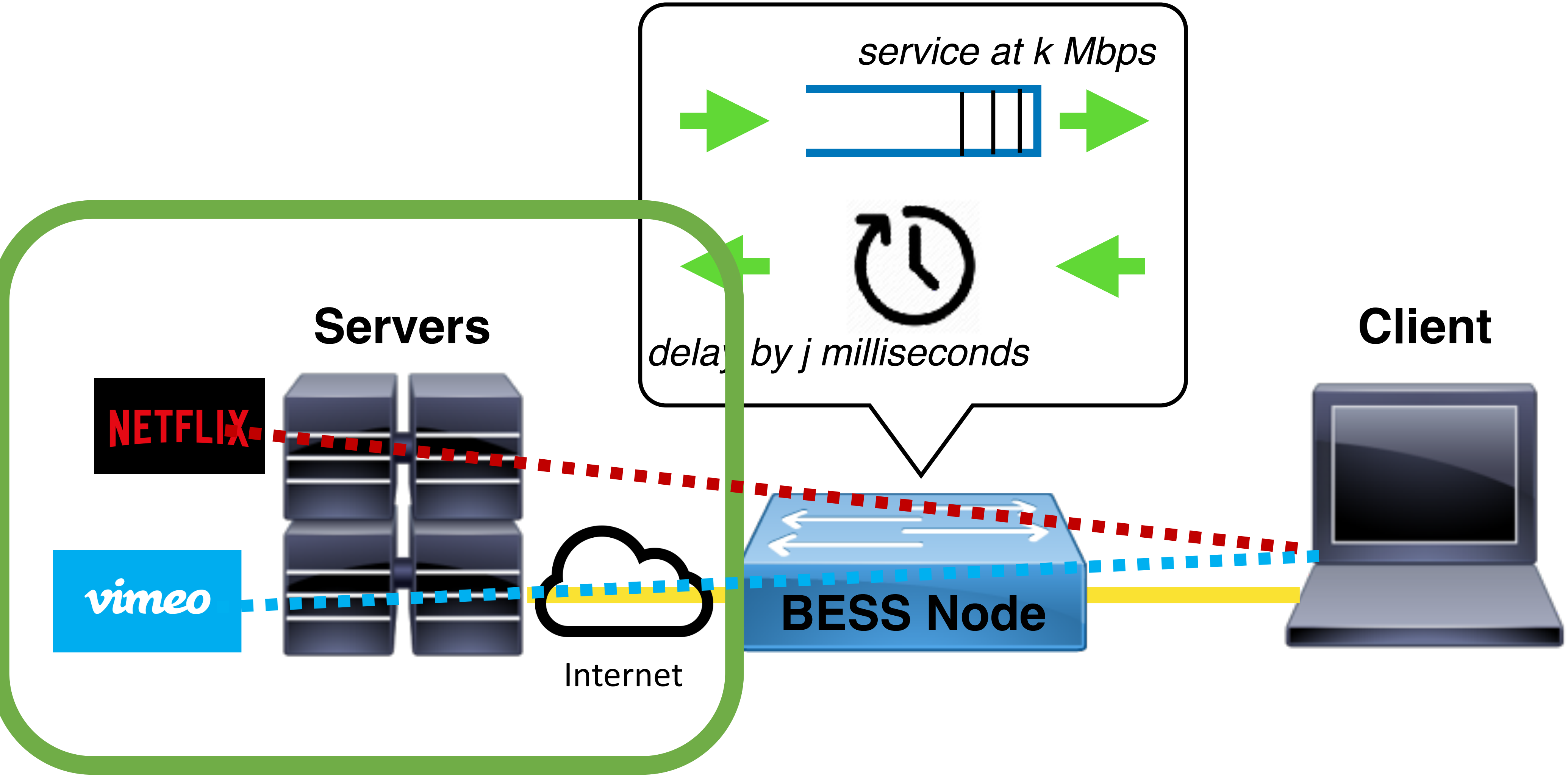
Rukshani is applying for PhD programs!



Programmable Switch: lets us emulate different network conditions, making the network behave like a home broadband connection or a 3G cellular link



Public Internet Services: we connect to the Internet "through" our emulated network to various public Internet services forcing their traffic to share the same congested bottleneck



Which video platform gets better service when the two compete for bandwidth?

Do some services cause more "harm" or damage than others?

How can we decide that a new Internet service is “too harmful”?

Prudentia: a testbed for measuring Internet services' fairness and the *harm* that competing services cause each other

Come to next year's Cylab Conference to learn what we find ☺

Justine Sherry
Assistant Professor
Computer Science Department & Cylab

sherry@cs.cmu.edu
@justinesherry