TDTS21 Advanced Networking

Lecture 5: Multipath TCP …

Based on slides from J. Rexford

Revised Spring 2015 by N. Carlsson

Multipath

- Mobile user
  - WiFi and cellular at the same time
- High-end servers
  - Multiple Ethernet cards
- Data centers
  - Rich topologies with many paths

- Benefits of multipath
  - Higher throughput
  - Failover from one path to another
  - Seamless mobility

Bringing Multipath to the End Host

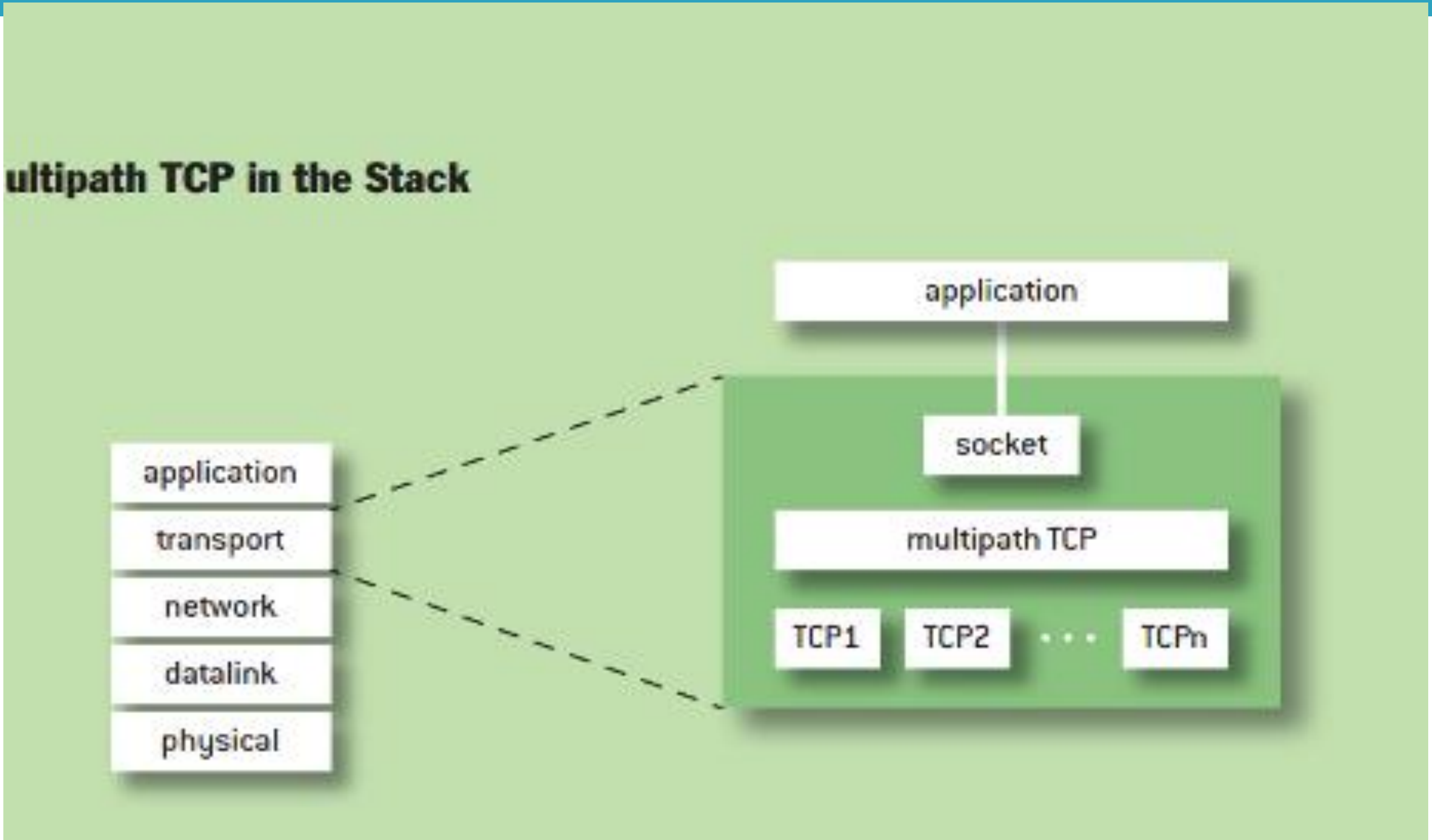
- Multiple addresses
  - One or more addresses at an end host
  - E.g., one per interface card
- Multiple paths
  - Sequence of links between sender and receiver
  - E.g., four-tuple of source and dest address and port
- Multiple subflows
  - Flow of TCP segments over an individual path
  - All associated with a single TCP connection

Keeping the Same Socket API

- Backwards compatibility with existing apps
  - Present the same socket API and expectations

- Establish the TCP connection in the same way
  - Create a socket to a single remote IP address/port
  - … and then add more subflows to the connection
- Work in all scenarios where regular TCP works
  - If a subflow fails, the connection should continue
  - … as long as some other subflow has connectivity

MPTCP in the Network Stack

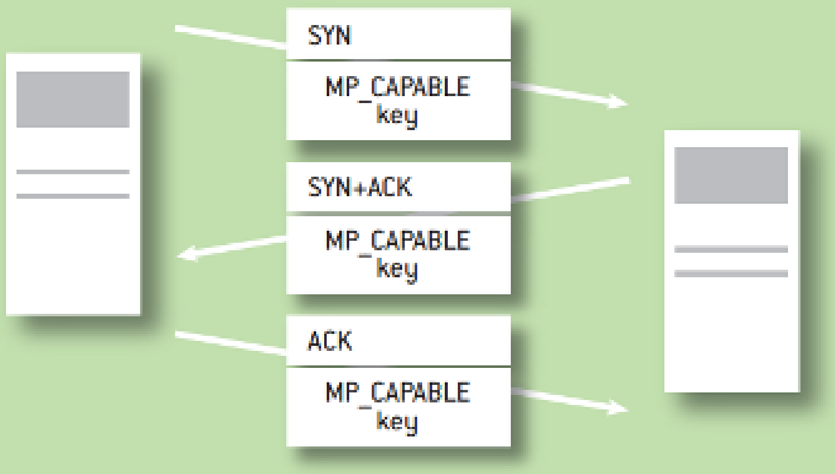


From http://queue.acm.org/detail.cfm?id=2591369

Negotiating MTTCP Capability

- How do end-points know they both speak MPTCP?
  - During the 3-way SYN/SYN-ACK/ACK handshake

FIGURE 2 **Connection Establishment**



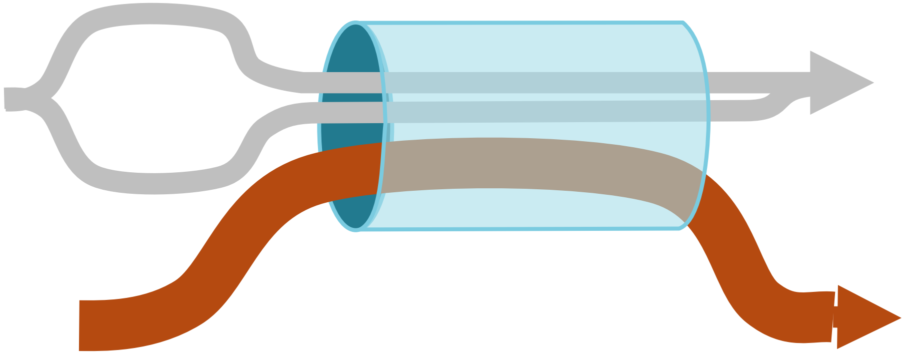
Use of Multipath TCP in iOS 7

- Multipath TCP in iOS 7
  - Primary TCP connection over WiFi
  - Backup TCP connection over cellular data
- Failover
  - If WiFi becomes unavailable…
  - … iOS 7 will use the cellular data connection
- For destinations controlled by Apple
  - E.g., Siri

MULTIPATH CONGESTION CONTROL

Slides from Damon Wischik

Goal #1: Fair at Shared Bottlenecks

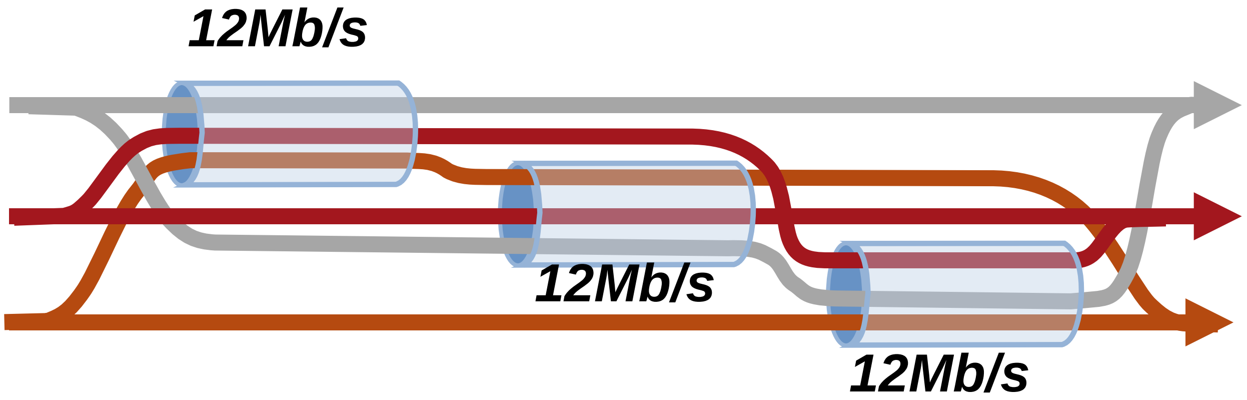


A multipath TCP flow with two subflows

Regular TCP

To be fair, Multipath TCP should take as much capacity as TCP at a bottleneck link, no matter how many paths it is using.

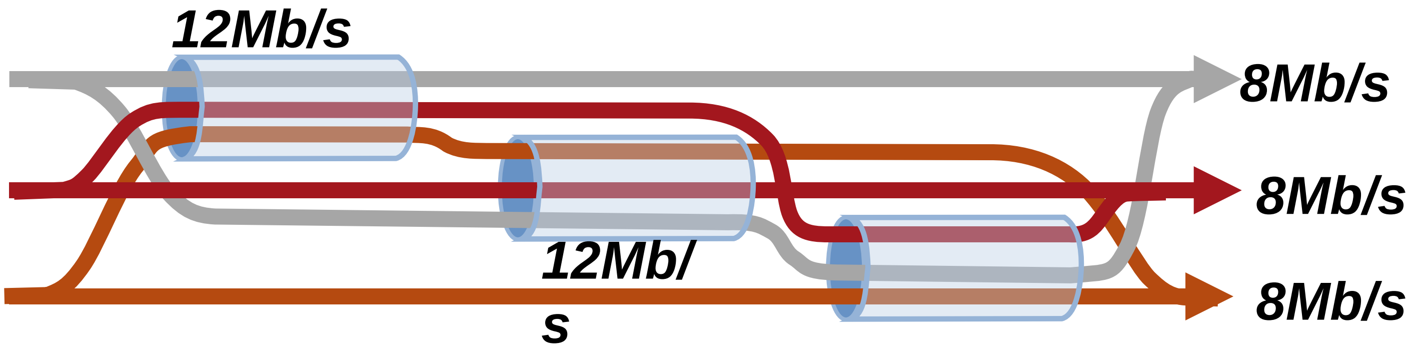
Use Efficient Paths



Each flow has a choice of a 1-hop and a 2-hop path.

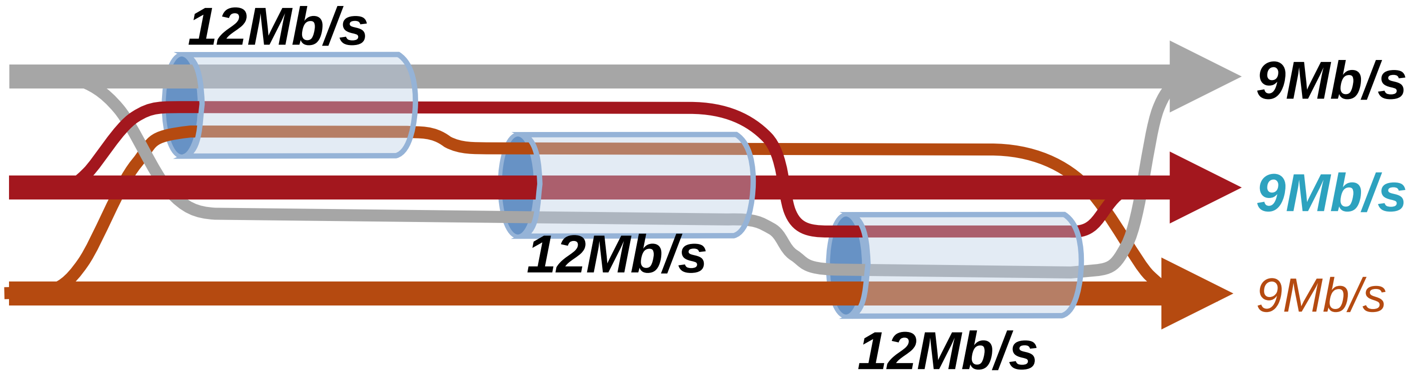
How should split its traffic?

Use Efficient Paths



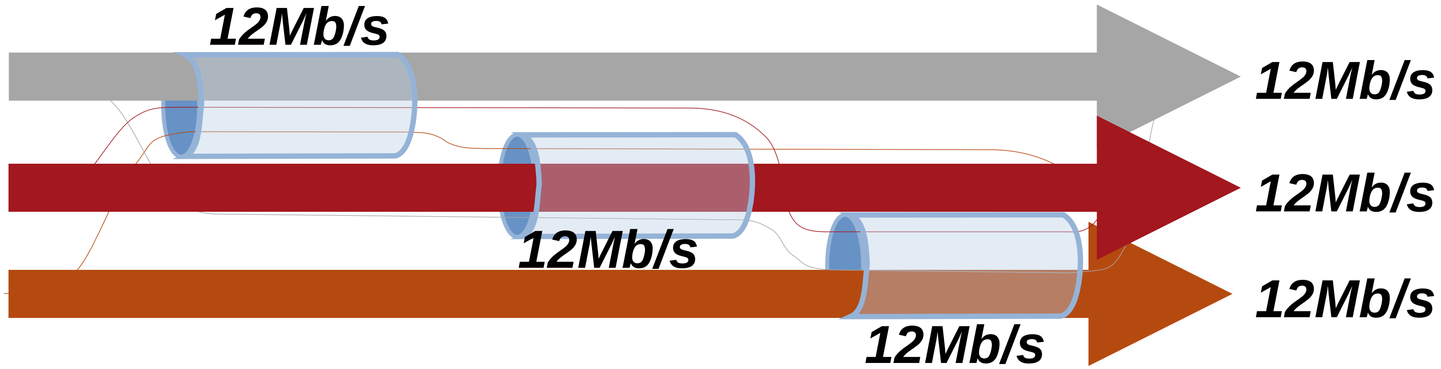
If each flow split its traffic 1:1 ...

Use Efficient Paths



If each flow split its traffic 2:1 ...

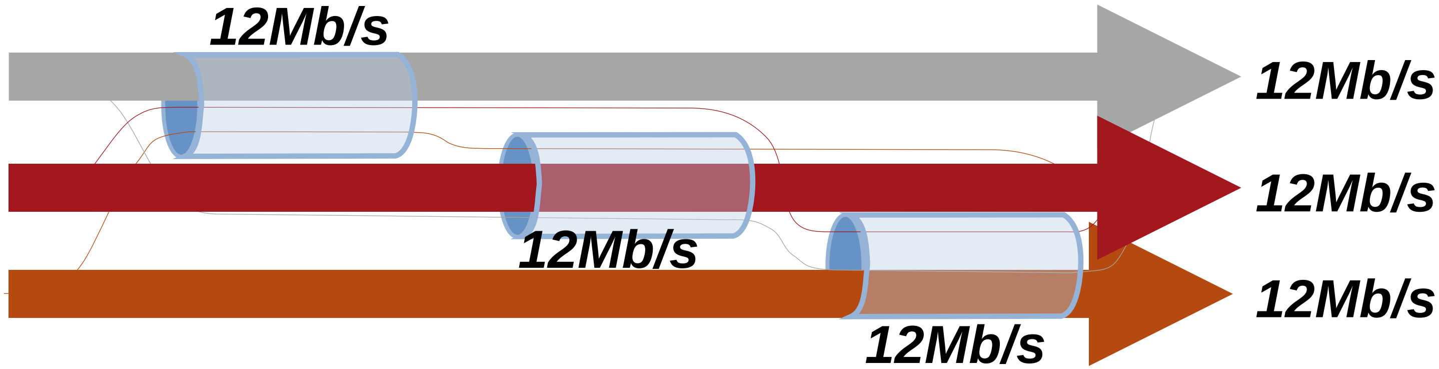
Use Efficient Paths



Better: Each connection on a one-hop path

Each connection should send all traffic on the least-congested paths

Use Efficient Paths



Better: Each connection on a one-hop path

Each connection should send all traffic on the least-congested paths

But keep some traffic on the alternate paths as a probe