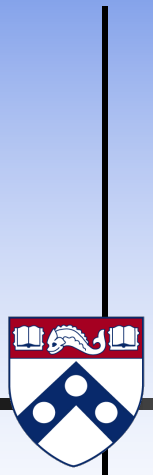


CIS 553: Networked Systems

Transport Layer

March 3, 2021

Agenda

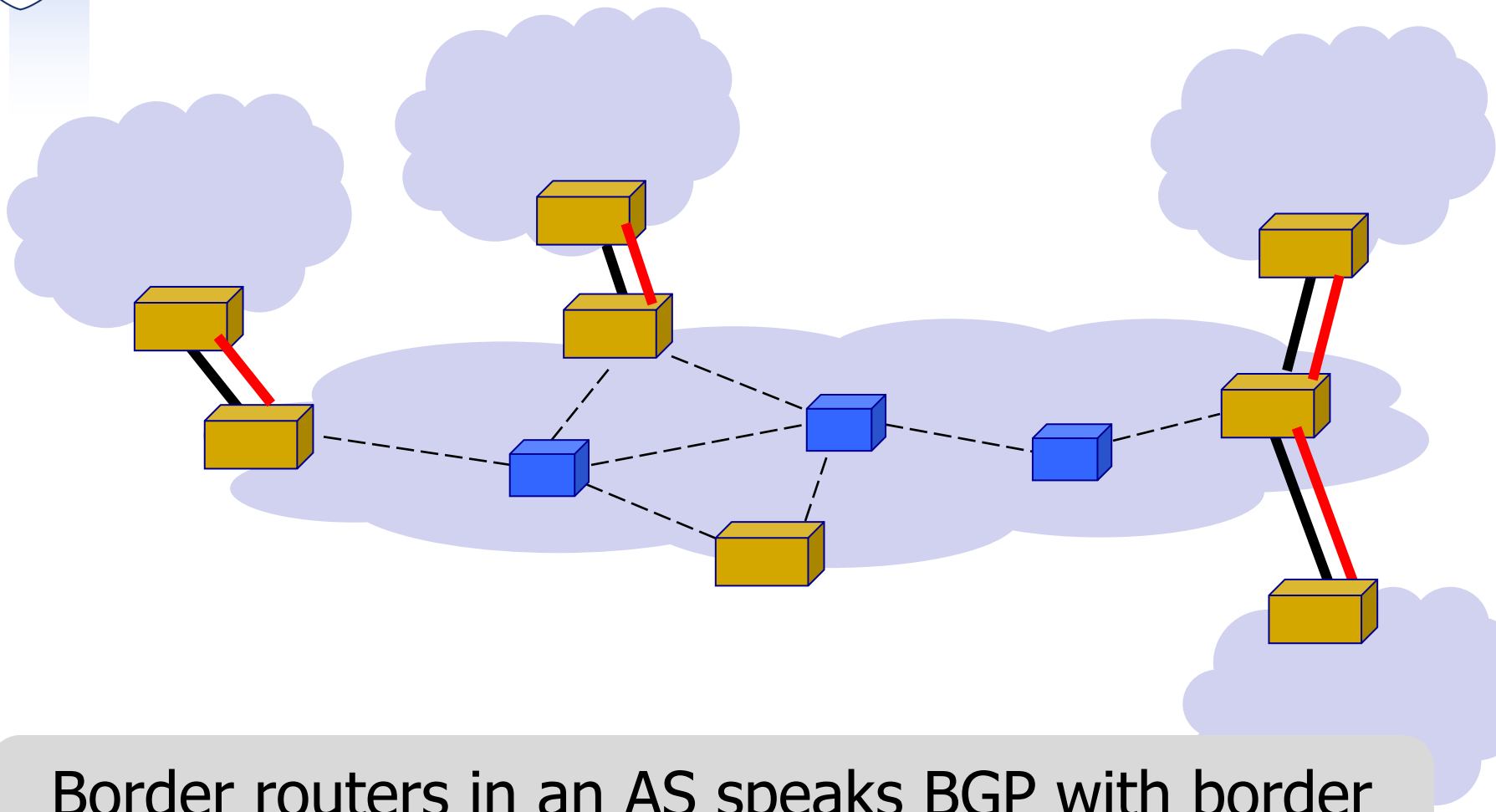
- Interdomain Routing 
  - Structure of the Internet 
  - Policy 
  - BGP 

  - Issues with BGP
- Transport Layer
  - UDP
  - TCP

eBGP, iBGP, and IGP

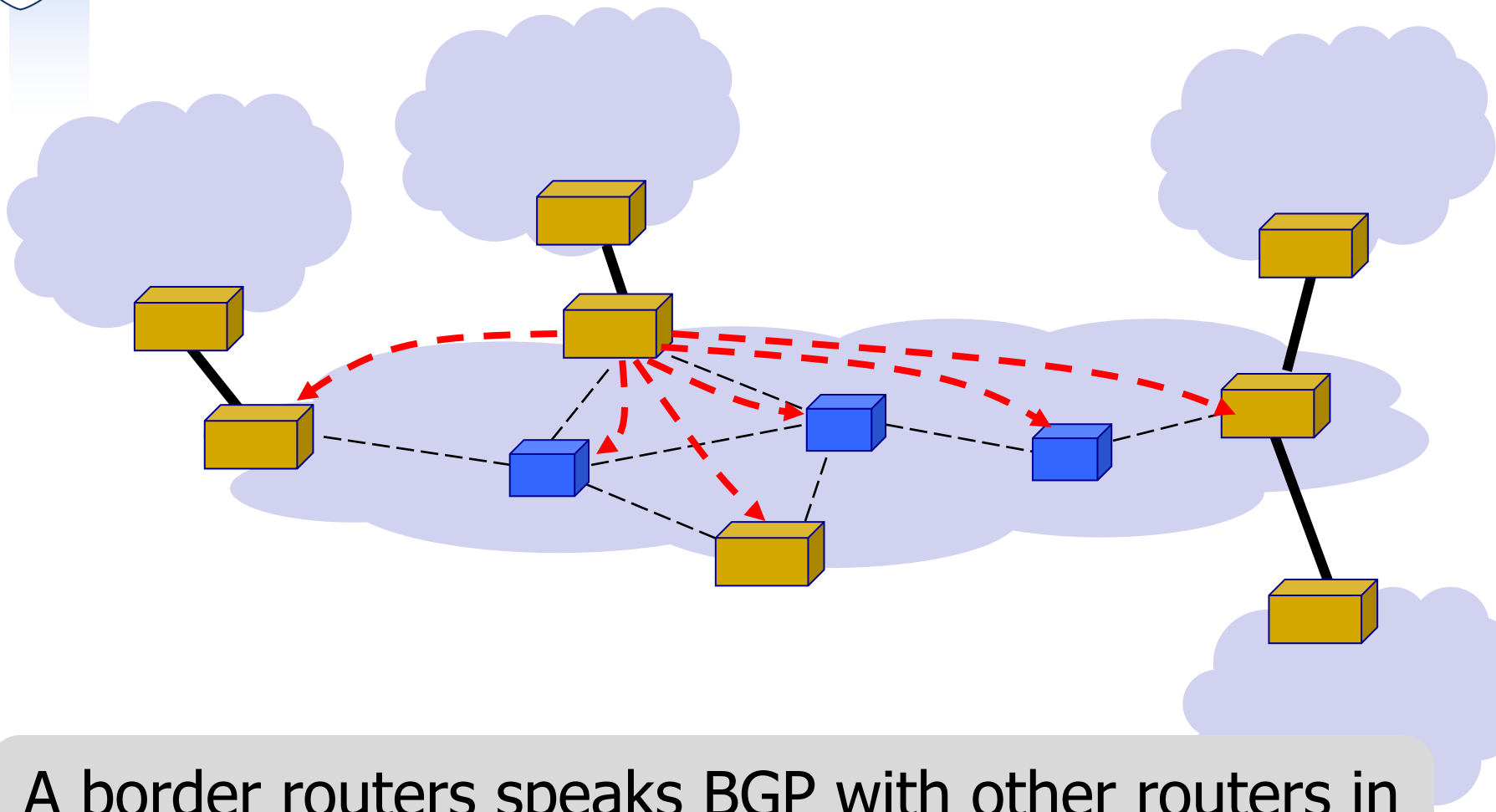
- eBGP: BGP sessions between border routers in different ASes
  - Learn routes to external destinations
- iBGP: BGP sessions between border routers and other routers within the same AS
  - Distribute externally learned routes internally
- IGP: "Interior Gateway Protocol" = Intra-domain routing protocol
  - Provide internal reachability
  - E.g., OSPF, RIP

BGP sessions: External



Border routers in an AS speaks BGP with border routers in other ASes using eBGP sessions

BGP sessions: Internal



A border routers speaks BGP with other routers in the same AS using iBGP sessions

BGP route updates

- Format <IP prefix: route attributes>
  - Attributes describe properties of the route
- Two kinds of updates
  - Announcements: new routes or changes to existing routes
  - Withdrawal: remove routes that no longer exist

Route attributes

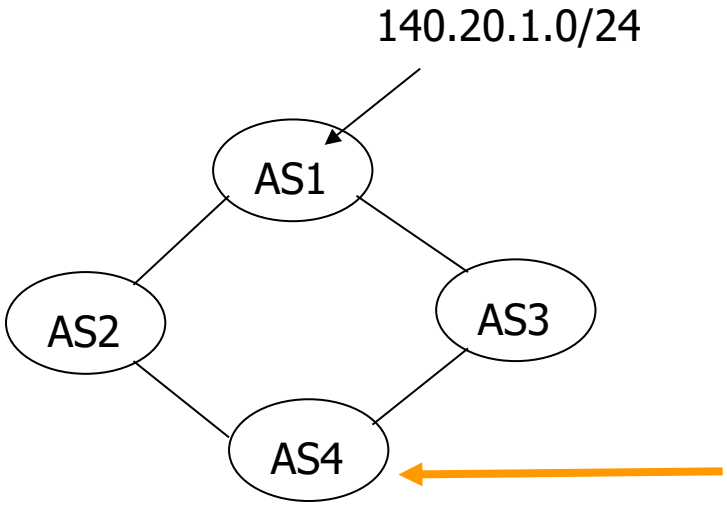
- Routes are described using attributes
  - Used in route selection/export decisions
- Some attributes are local
  - I.e., private within an AS, not included in announcements
- Some attributes are propagated with eBGP route announcements
- There are many standardized attributes in BGP
  - We will discuss a few

Attributes: (1) ASPATH

- Carried in route announcements
- Vector that lists all the ASes a route advertisement has traversed (in reverse order)

Attributes: (2) LOCAL PREF

- Local preference in choosing between different AS paths
  - Local to an AS; carried only in iBGP messages
- The higher the value the more preferred

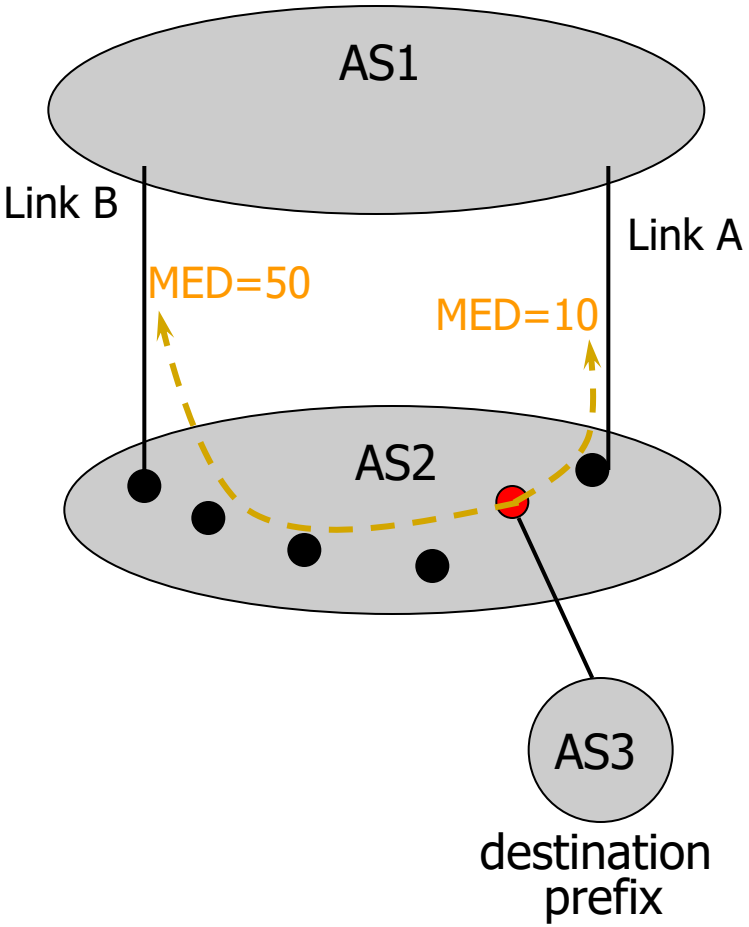


BGP table at AS4:

Destination	AS Path	Local Pref
140.20.1.0/24	AS3 AS1	300
140.20.1.0/24	AS2 AS1	100

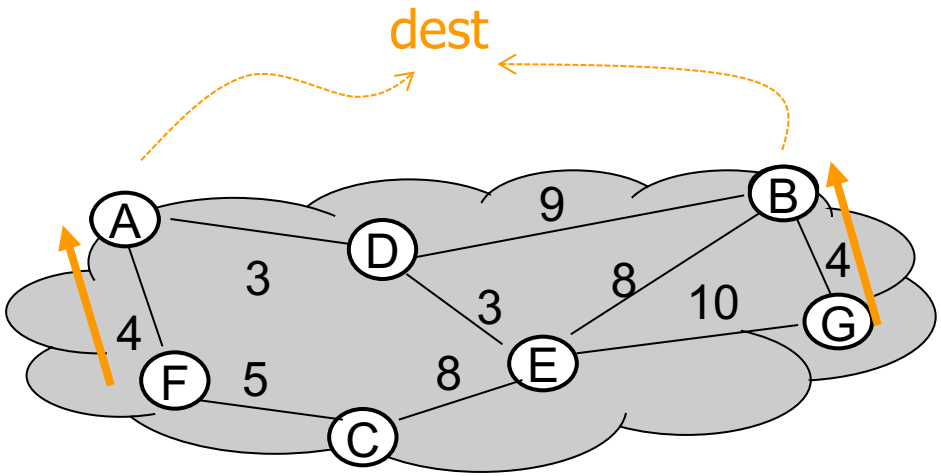
Attributes: (3) MED

- Multi-exit discriminator is used when ASes are interconnected via 2 or more links; it specifies how close a prefix is to the link it is announced on
- Lower is better
- AS that announces a prefix sets MED
- AS receiving the prefix (optionally!) uses MED to select link



Attributes: (4) IGP cost

- Used for hot-potato routing
  - Each router selects the closest egress point based on the path cost in intra-domain protocol



Using attributes

- Rules for route selection in priority order

Priority	Rule	Remarks
1	LOCAL PREF	Pick highest LOCAL PREF
2	ASPATH	Pick shortest ASPATH length
3	MED	Lowest MED preferred
4	eBGP > iBGP	Did AS learn route via eBGP (preferred) or iBGP?
5	iBGP path	Lowest IGP cost to next hop (egress router)
6	Router ID	Smallest next-hop router's IP address as tie-breaker

Issues with BGP

- Reachability
- Security
- Convergence
- Performance
- Anomalies

Agenda

- Interdomain Routing 
  - BGP 
  - Issues with BGP 
- Transport Layer 
  - UDP
  - TCP

Layer 3: Network layer

- Connects: LANs
  - All managed by different organizations
- Name of network: Internet
- Name of message: Packet
- Upper interface: A segment of data
- Lower interface: A packet of data
  - Same thing, just has an extra header

Layer 4: Transport layer

- On every host, typically not in the network
- Upper interface: Two options
  - UDP: Chunks of data
  - TCP: A stream of bytes
- Lower interface: A packet of data
- Name of message
  - UDP: Datagrams
  - TCP: Segments

What is the purpose of L4?

- Transport layer is where we "pay the piper"
  - Provide applications with good abstractions
  - Mostly without support or feedback from the network

Role of the transport layer

- Communication between processes
  - Mux and demux from/to application processes
  - Using L4 ports (NOT the same as L1 ports)

Role of the transport layer

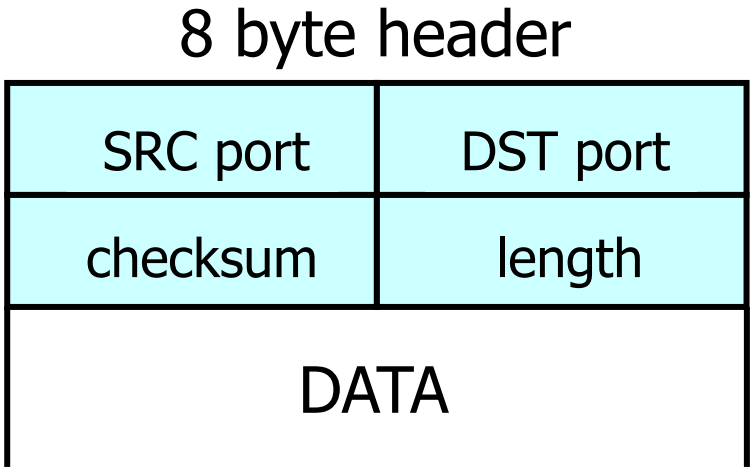
- Communication between processes
- Provide common end-to-end services for application layer [optional]
  - Reliable, in-order data delivery
  - Well-paced data delivery
    - Too fast may overwhelm the network
    - Too slow is not efficient

Role of the transport layer

- Communication between processes
- Provide common end-to-end services for app layer [optional]
- UDP and TCP are the common transport protocols
  - Also SCTP, MPTCP, SST, RDP, DCCP, …

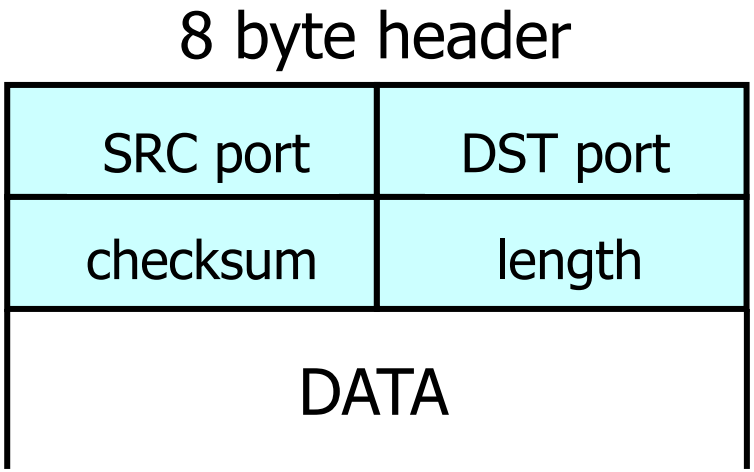
User Datagram Protocol (UDP)

- Lightweight communication between processes
  - Send and receive messages
  - Just a simple wrapper around IP
- Used by popular apps
  - Query/response for DNS
  - Real-time data in VoIP



UDP (cont'd)

- Optional error checking on the packet contents
  - (checksum field = 0 means "don't verify checksum")
- Source port is also optional
  - Useful to respond back to the sender in some cases



Why Use UDP?

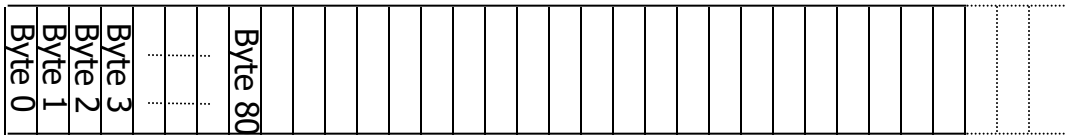
- Fine-grained control
  - UDP sends as soon as the application writes
- No connection set-up delay
  - UDP sends without establishing a connection
- No connection state
  - No buffers, parameters, sequence #s, etc.
- Small header overhead
  - UDP header is only eight-bytes long

Transmission Control Protocol (TCP)

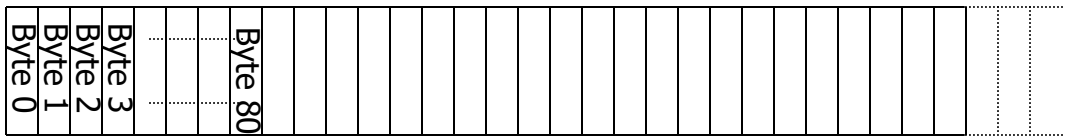
1. Stream-of-bytes service
   - Sends and receives a stream of bytes
2. Reliable, in-order delivery
   - Detect corruption, loss, and reordering
   - Reliable delivery: acknowledgments and retransmissions
3. Connection-oriented
   - Explicit set-up and tear-down of TCP connection
4. Flow control
   - Prevent overflow of the receiver's buffer space
5. Congestion control
   - Adapt to network congestion for the greater good

TCP's "stream of bytes" model

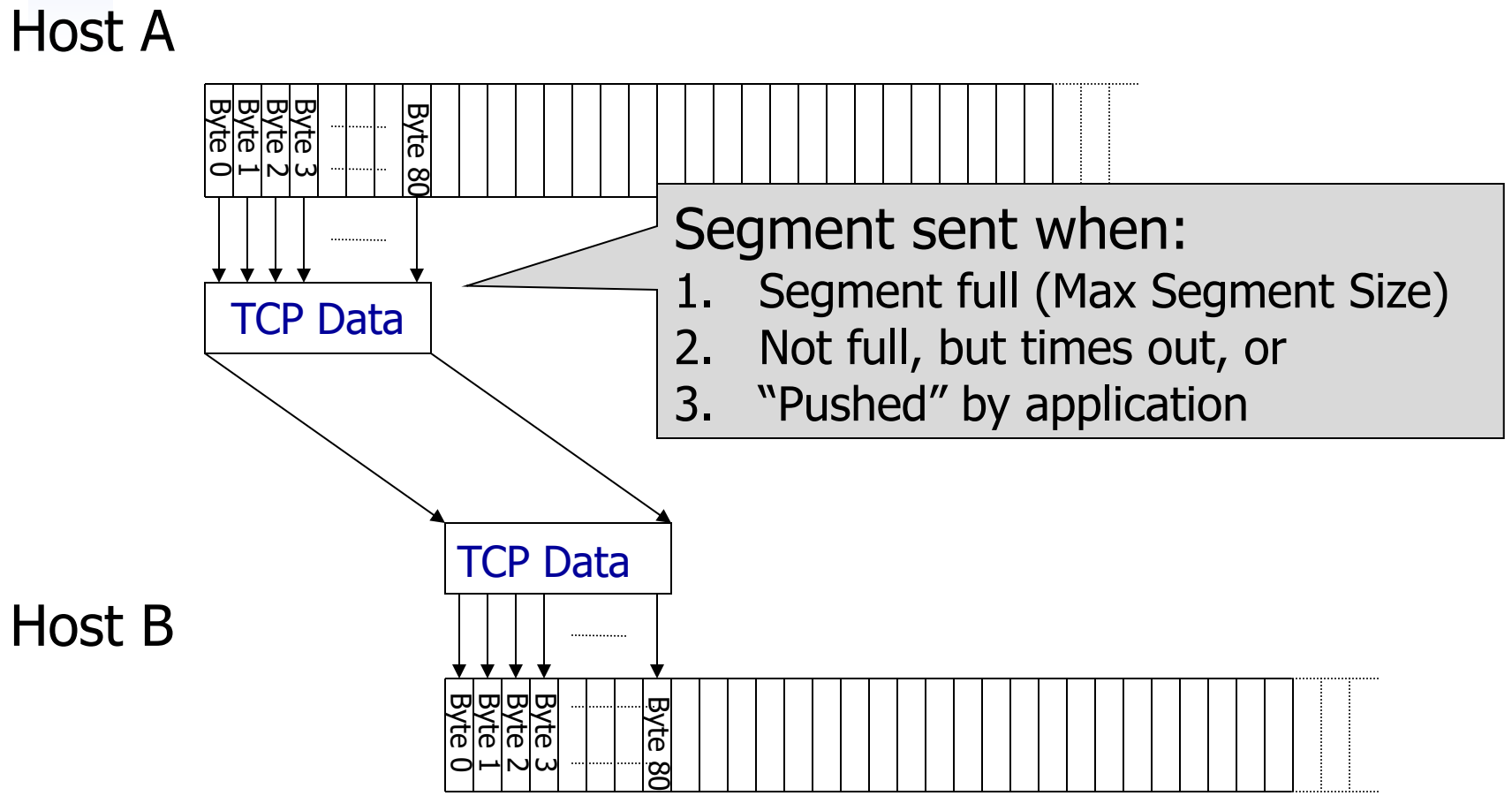
Host A



Host B



…Emulated Using TCP "Segments"



Transmission Control Protocol (TCP)

1. Stream-of-bytes service
   - Sends and receives a stream of bytes
2. Reliable, in-order delivery
   - Detect corruption, loss, and reordering
   - Reliable delivery: acknowledgments and retransmissions
3. Connection-oriented
   - Explicit set-up and tear-down of TCP connection
4. Flow control
   - Prevent overflow of the receiver's buffer space
5. Congestion control
   - Adapt to network congestion for the greater good

Challenges of Reliable Data Transfer

- Over a perfectly reliable channel? Done!
- What if packets can experience bit errors?
- What if packets can be lost?
- What if packets can be reordered?