Internet Protocols EBU5403
Live Lecture A3/A4 + tutorial

Module organiser: Richard Clegg
(r.clegg@qmul.ac.uk)
Michael Chai (michael.chai@qmul.ac.uk)
Cunhua Pan(c.pan@qmul.ac.uk)

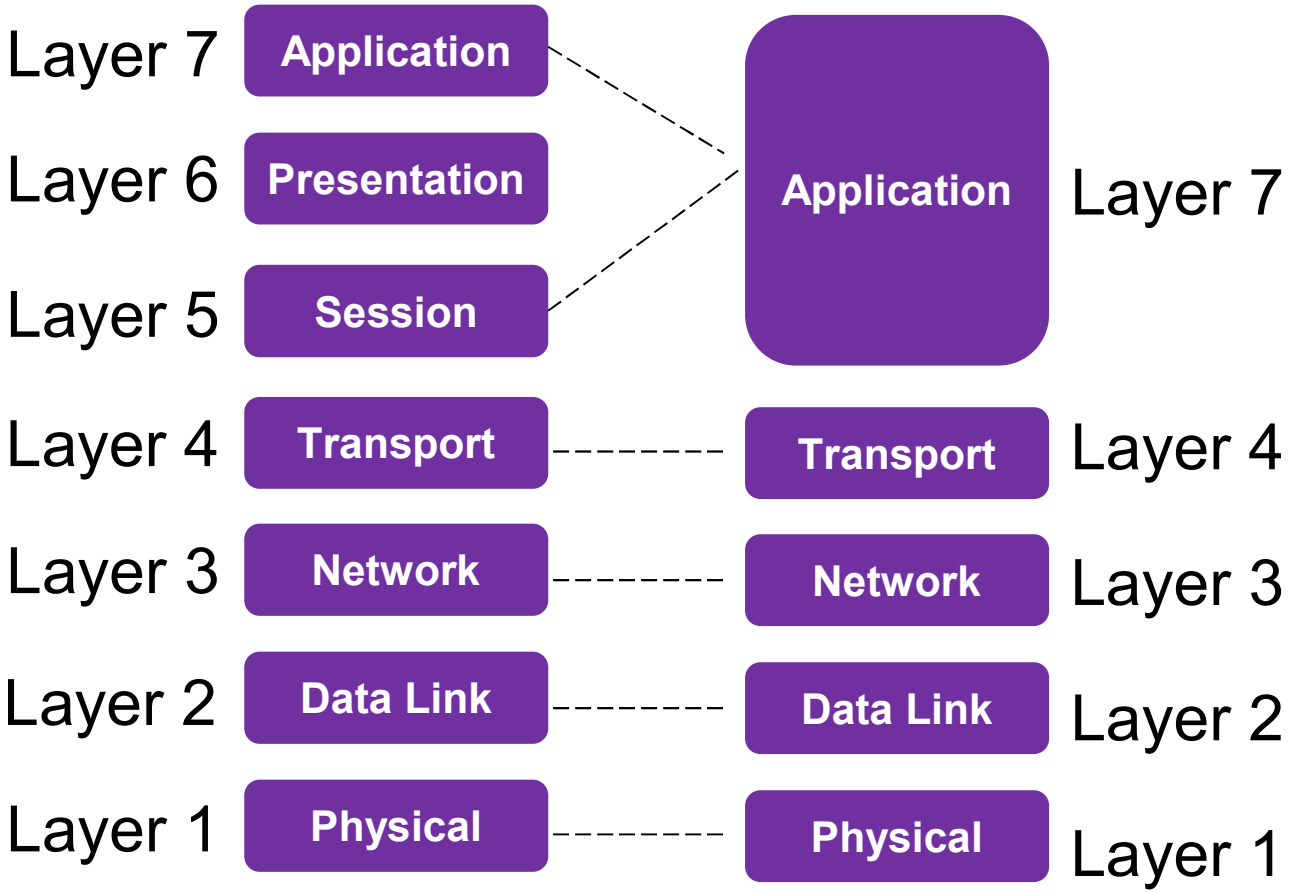
	Part 1	Part 2	Part 3	Part 4
Ecommerce + Telecoms 1	Richard Clegg		Cunhua Pan	
Telecoms 2	Michael Chai			

Go to www.menti.com and use the code 12 22 69

Structure of course

- Part A
  - Introduction to IP Networks
  - The Transport layer (part 1)
- Part B
  - The Transport layer (part II)
  - The Network layer (part I)
  - Class test
- Part C
  - The Network layer (part II)
  - The Data link layer (part I)
  - Router lab tutorial (assessed lab work after this week)
- Part D
  - The Data link layer (part II)
  - Network management and security
  - Class test

ISO/OSI (left) vs TCP/IP (right)



What layer? 什么层

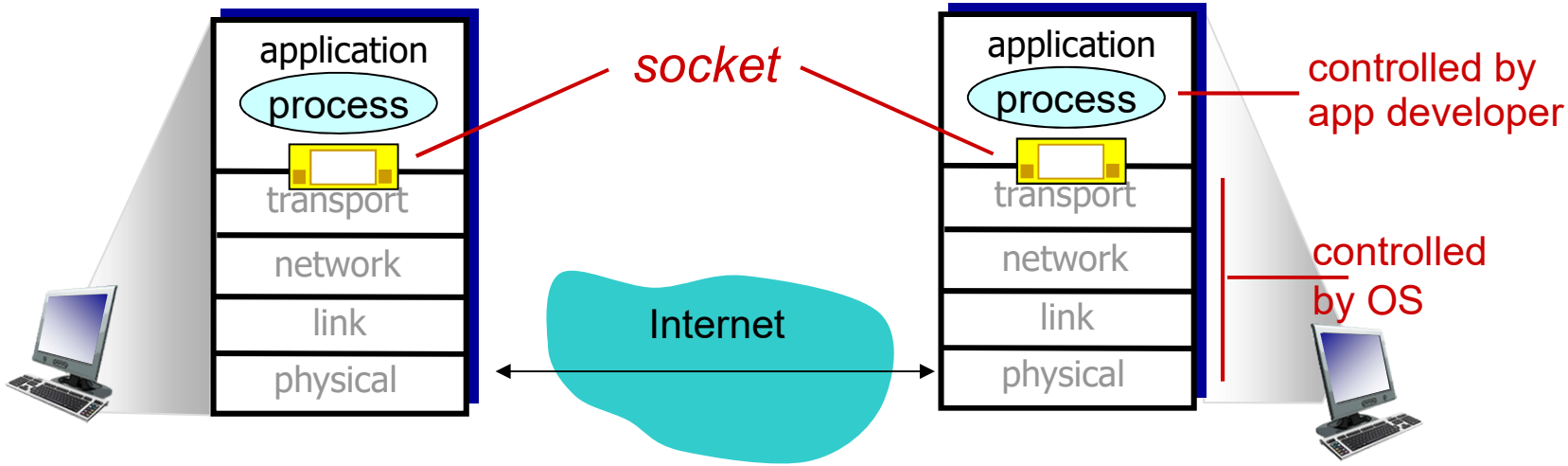
- Answer quickly using mentimeter (or chat if mentimeter is not working) the number of the layer. If you don't know take a guess as to which fits best.
- Transmission Control Protocol.
  - Transport (layer 4)
- Delivery of traffic across the whole network.
  - Network (layer 3)
- Delivery of traffic between a laptop and a WiFi router.
  - Datalink (layer 2)
- A repeater (device to boost a signal).
  - Physical (layer 1)

What layer? (PUBG -- game)

Application layer – the FUN layer!

Sockets

- process sends/receives messages to/from its socket
- socket analogous to door
  - sending process shoves message out door
  - sending process relies on transport infrastructure on other side of door to deliver message to socket at receiving process



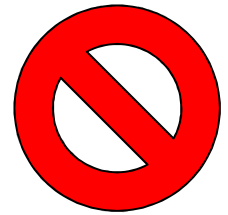
Addressing processes

- to receive messages, process must have *identifier*
- host device has unique 32-bit IP address
- *Q:* does IP address of host on which process runs suffice for identifying the process?
  - *A:* no, *many* processes can be running on same host

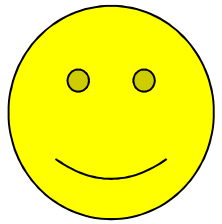
- *identifier* includes both IP address and port numbers associated with process on host.
- example port numbers:
  - HTTP server: 80
  - mail server: 25
- to send HTTP message to gaia.cs.umass.edu web server:
  - IP address: 128.119.245.12
  - port number: 80
- more shortly…

Transport layer True/False

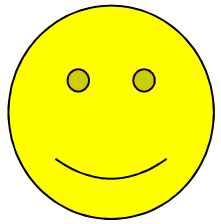
- A) Every device on the network needs a transport layer.



- B) A device must have a transport layer to run networked applications over TCP/IP

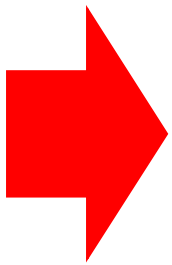


- C) If it is needed the Transport layer provides reliable in order delivery of packets.

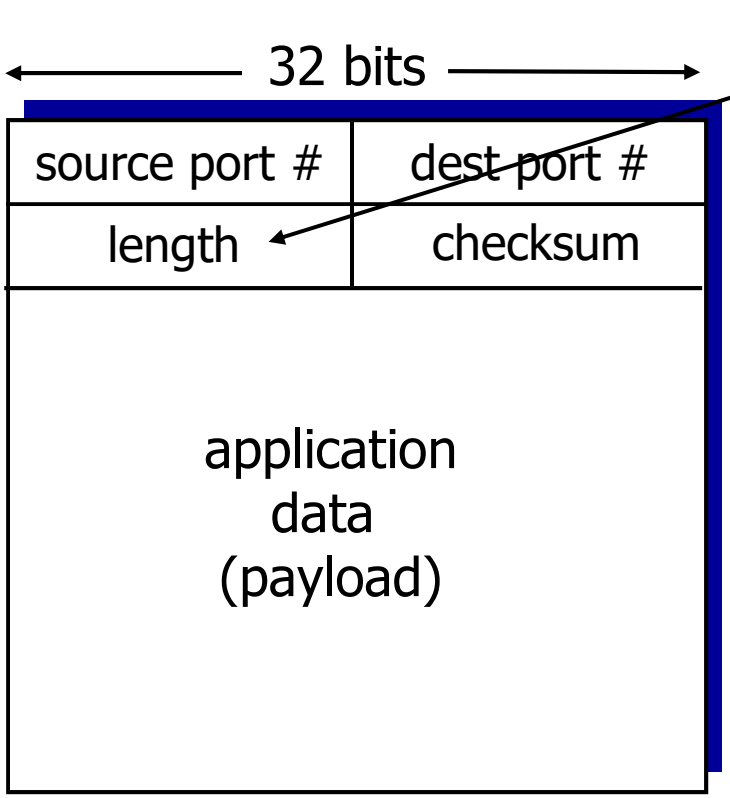


The UDP header

- A UDP header contains
  - A checksum
  - A source port
  - A destination port
  - All three of the above



UDP: segment header



length, in bytes of UDP segment, including header

UDP segment format

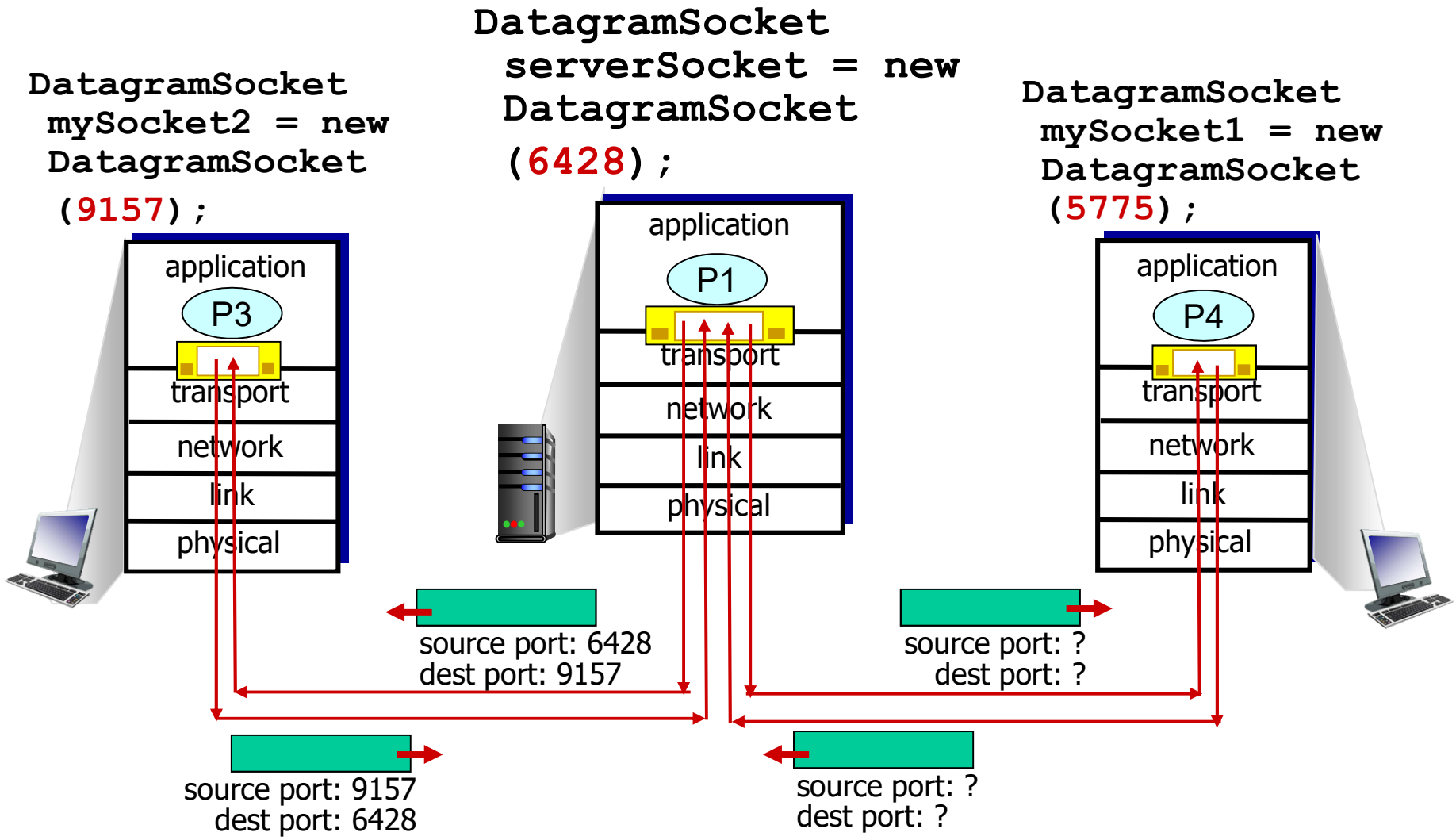
why is there a UDP?

- no connection establishment (which can add delay)
- simple: no connection state at sender, receiver
- small header size
- no congestion control: UDP can blast away as fast as desired

UDP IPs and ports

- A UDP application at IP 12.13.14.15 listening on port 80 receive a packet from 100.101.102.103 port 4324. When it replies, what is the new destination IP and destination port.
  - 100.101.102.103 : 4324

Connectionless demux: example

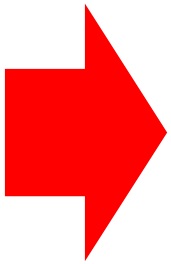


Give advantages of UDP over TCP

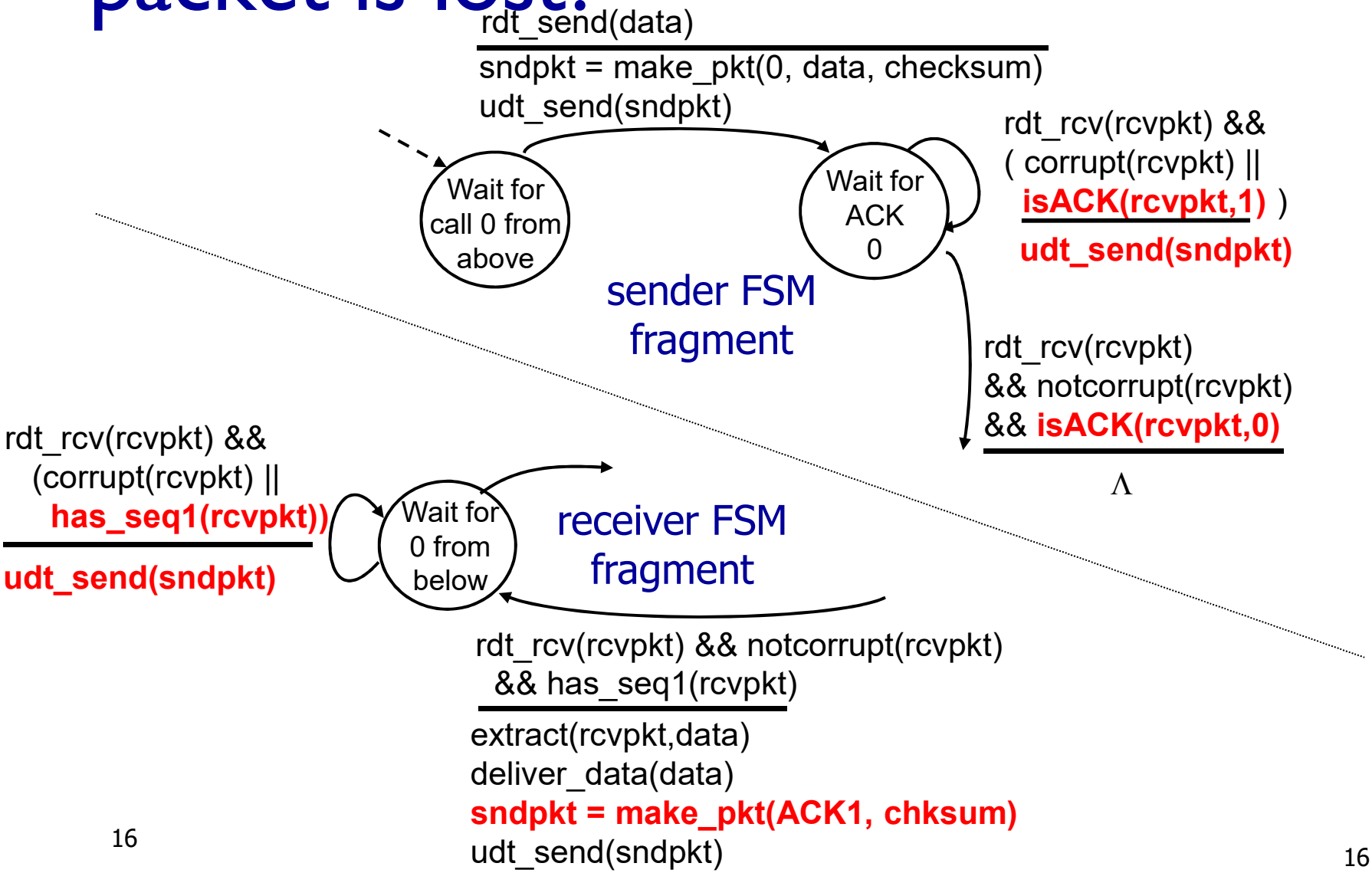
- Simple to implement
  - I just want to get something working.
- Efficient : No "overhead" (short header length)
  - Very good if I am sending short packets
- No setting up of a connection
  - I can send my data immediately – very fast
  - (Later we will see TCP handshake)
- Sometimes we don't want retransmission
  - In a live transmission might be not useful to resend data.
  - Broadcast of data to many people

Reliable data transfer 2.0

- In rdt 2.0 we send an ACK to say we received data and a NAK to say we did not. If we receive two packets correctly then one with an error we send:
  - A) ACK NAK
  - B) NAK NAK ACK
  - C) ACK ACK ACK
  - D) ACK ACK NAK



What will happen in rdt 2.2 if a packet is lost?

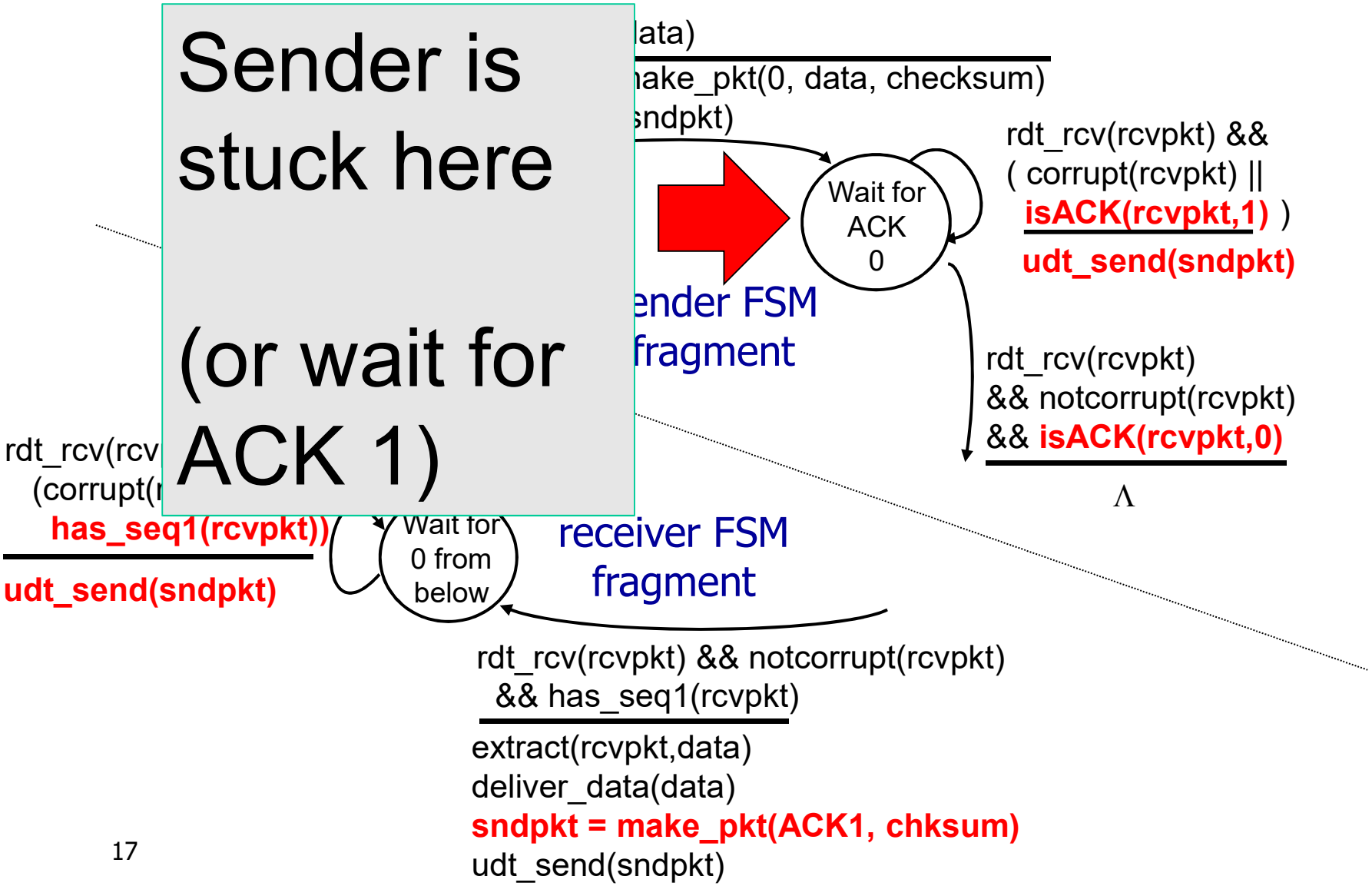


16

rdt2.2: sender, receiver fragments

Sender is stuck here

(or wait for ACK 1)



Give an advantage of rdt 3.0 over UDP

- Resends lost packets:
  - If a packet is dropped it is resent
- Resends corrupted packets:
  - If a packet is corrupted it is resent
- We will see in later lectures TCP also:
  - Puts packets in order
  - Slows sending if receiver is “full”
  - Slows sending if congestion occurs

What have we learned?

- The transport layer multiplexes and demultiplexes traffic.
- The transport layer exists on end machines to get traffic to and from a port connected to an application.
- UDP provides a simple connectionless service
- It does not correct for loss or corrupted packets.
- rdt (various protocols) can deliver reliable traffic over unreliable networks. uses ACKs and timeouts to provide a reliable service.
- rdt 3.0 deals with timeouts and corruption.