

Lecture 13: Midterm Review, Network Layer Overview

COMP 332, Spring 2024

Victoria Manfredi



Acknowledgements: materials adapted from Computer Networking: A Top Down Approach 7th edition: ©1996-2016, J.F Kurose and K.W. Ross, All Rights Reserved as well as from slides by Abraham Matta at Boston University, and some material from Computer Networks by Tannenbaum and Wetherall.

Today

Announcements

- Midterm Wed. in class

Midterm

- How is everybody feeling about it?
- We can just make today a review class ...

Network layer

- Overview

Wireshark

**WHAT IS THE UPPER LAYER
PROTOCOL?**

Directing bits, frames, packets, segments ...

- ▼ Frame 30: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0
 - > Interface id: 0 (en0)
 - Encapsulation type: Ethernet (1)
 - Arrival Time: Mar 22, 2024 09:06:06.881620000 EDT
 - [Time shift for this packet: 0.000000000 seconds]
 - Epoch Time: 1711112766.881620000 seconds
 - [Time delta from previous captured frame: 0.016491000 seconds]
 - [Time delta from previous displayed frame: 0.016491000 seconds]
 - [Time since reference or first frame: 0.273634000 seconds]
 - Frame Number: 30
 - Frame Length: 66 bytes (528 bits)
 - Capture Length: 66 bytes (528 bits)
 - [Frame is marked: False]
 - [Frame is ignored: False]
 - [Protocols in frame: eth:ethertype:ip:tcp]
 - [Coloring Rule Name: TCP]
 - [Coloring Rule String: tcp]
- ▼ Ethernet II, Src: Motorola_f6:83:2b (38:80:df:f6:83:2b), Dst: 88:66:5a:28:6e:b1 (88:66:5a:28:6e:b1)
 - > Destination: 88:66:5a:28:6e:b1 (88:66:5a:28:6e:b1)
 - > Source: Motorola_f6:83:2b (38:80:df:f6:83:2b)
 - Type: IPv4 (0x0800)
- ▼ Internet Protocol Version 4, Src: 17.248.202.64 (17.248.202.64), Dst: 192.168.0.11 (192.168.0.11)
 - 0100 = Version: 4
 - 0101 = Header Length: 20 bytes (5)
 - > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
 - Total Length: 52
 - Identification: 0xa0df (41183)
 - > Flags: 0x02 (Don't Fragment)

Directing bits, frames, packets, segments ...

```

  ▾ Ethernet II, Src: Motorola_f6:83:2b (38:80:df:f6:83:2b), Dst: 88
    > Destination: 88:66:5a:28:6e:b1 (88:66:5a:28:6e:b1)
    > Source: Motorola_f6:83:2b (38:80:df:f6:83:2b)
    Type: IPv4 (0x0800)
  ▾ Internet Protocol Version 4, Src: 17.248.202.64 (17.248.202.64),
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
    > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 52
    Identification: 0xa0df (41183)
    > Flags: 0x02 (Don't Fragment)
    Fragment offset: 0
    Time to live: 52
    Protocol: TCP (6)
    Header checksum: 0x08f9 [validation disabled]
    [Header checksum status: Unverified]
    Source: 17.248.202.64 (17.248.202.64)
    Destination: 192.168.0.11 (192.168.0.11)
    [Source GeoIP: Unknown]
    [Destination GeoIP: Unknown]
  ▾ Transmission Control Protocol, Src Port: 443, Dst Port: 53603, S
    Source Port: 443
    Destination Port: 53603
    [Stream index: 0]
    [TCP Segment Len: 0]
    Sequence number: 7675 (relative sequence number)

```

Network Layer

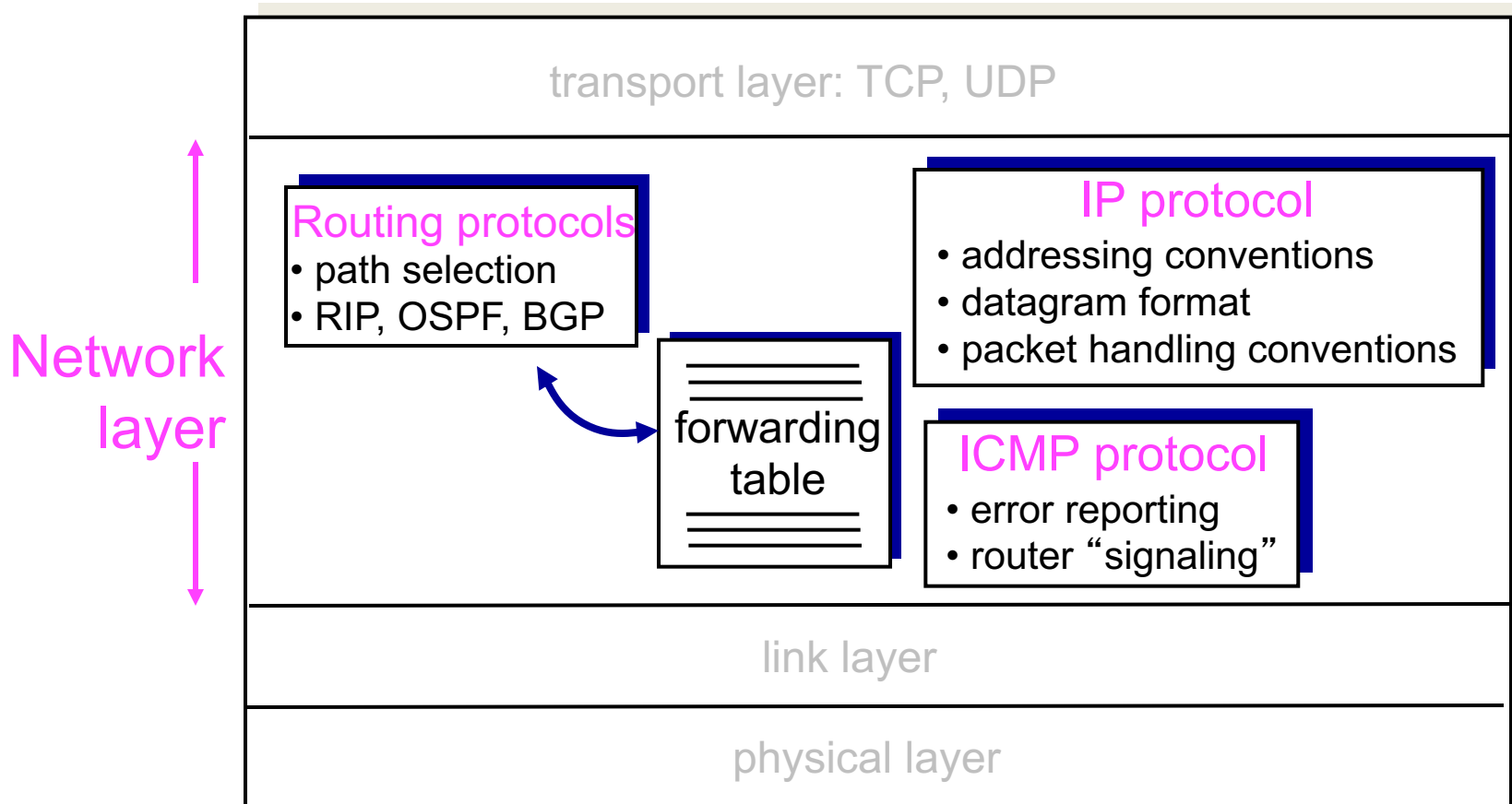
OVERVIEW

5-layer Internet protocol stack

Layer	Service provided to upper layer	Protocols	Unit of information
5 Application	<ul style="list-style-type: none">Support network applications	FTP, DNS, SMTP, HTTP	Message 1 message may be split into multiple segments
4 Transport	<ul style="list-style-type: none">Deliver messages to app endpointsFlow controlReliability	TCP (reliable) UDP (best-effort)	Segment (TCP) Datagram (UDP) 1 segment may be split into multiple packets
3 Network	<ul style="list-style-type: none">Route segments from source to destination host	IP (best-effort) Routing protocols	Packet (TCP) Datagram (UDP)
2 Link	<ul style="list-style-type: none">Move packet over link from one host to next host	Ethernet, 802.11	Frame MTU is 1500 bytes
1 Physical	<ul style="list-style-type: none">Move individual bits in frame from one host to next“bits on wire”	Ethernet phy 802.11 phy Bluetooth phy DSL	Bit

Internet's network layer

Network layer functions on hosts and routers



Network layer

Goal

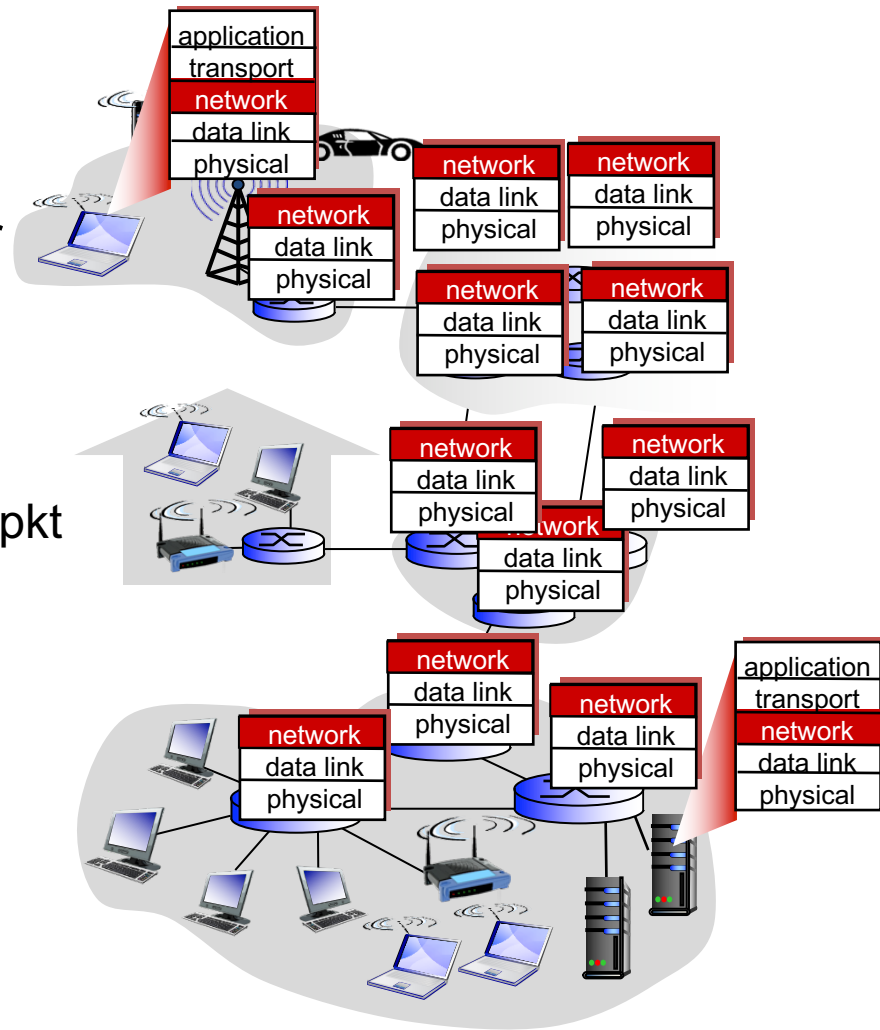
- move pkt from one host to another

How done on Internet?

- routers
 - examine header fields in every IP pkt
 - determines outgoing link

Internet e2e argument

- some functionality only properly implemented in end systems
- smart hosts vs. dumb routers



Network layer is in every host and router on Internet

Encapsulation and decapsulation

Sender

- encapsulates segments into packets, puts src, dest IP in IP pkt hdr

Receiver

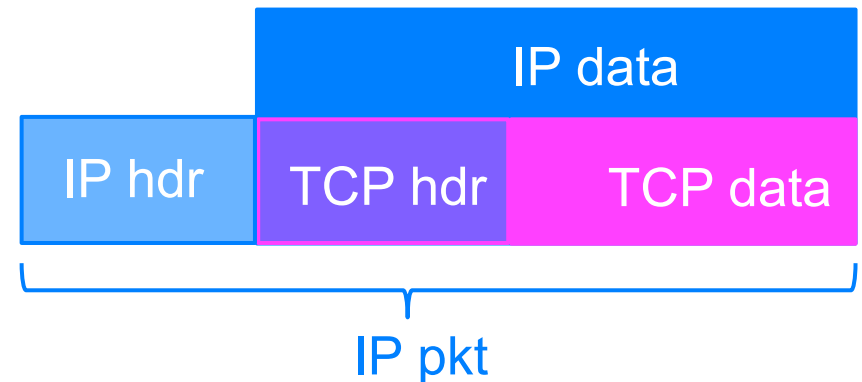
- decapsulates packets into segments, delivers to transport layer

Max length of IP packet in bytes

- MTU: Maximum Transmission Unit
- 1500 bytes if Ethernet used as link layer protocol

Max length of TCP data in bytes

- MSS: Maximum Segment Size
- $MSS = MTU - IP\ hdr - TCP\ hdr$
 - TCP header ≥ 20 bytes



Division of network layer functionality

1. Control plane

- comprises traffic only between routers, to compute routes between src and dst
- network-wide: routers run routing algorithms

2. Data plane

- comprises traffic between end hosts, forwarded by routers
- forwarding table set based on routes computed in control plane
- local: each router stores, forwards packets

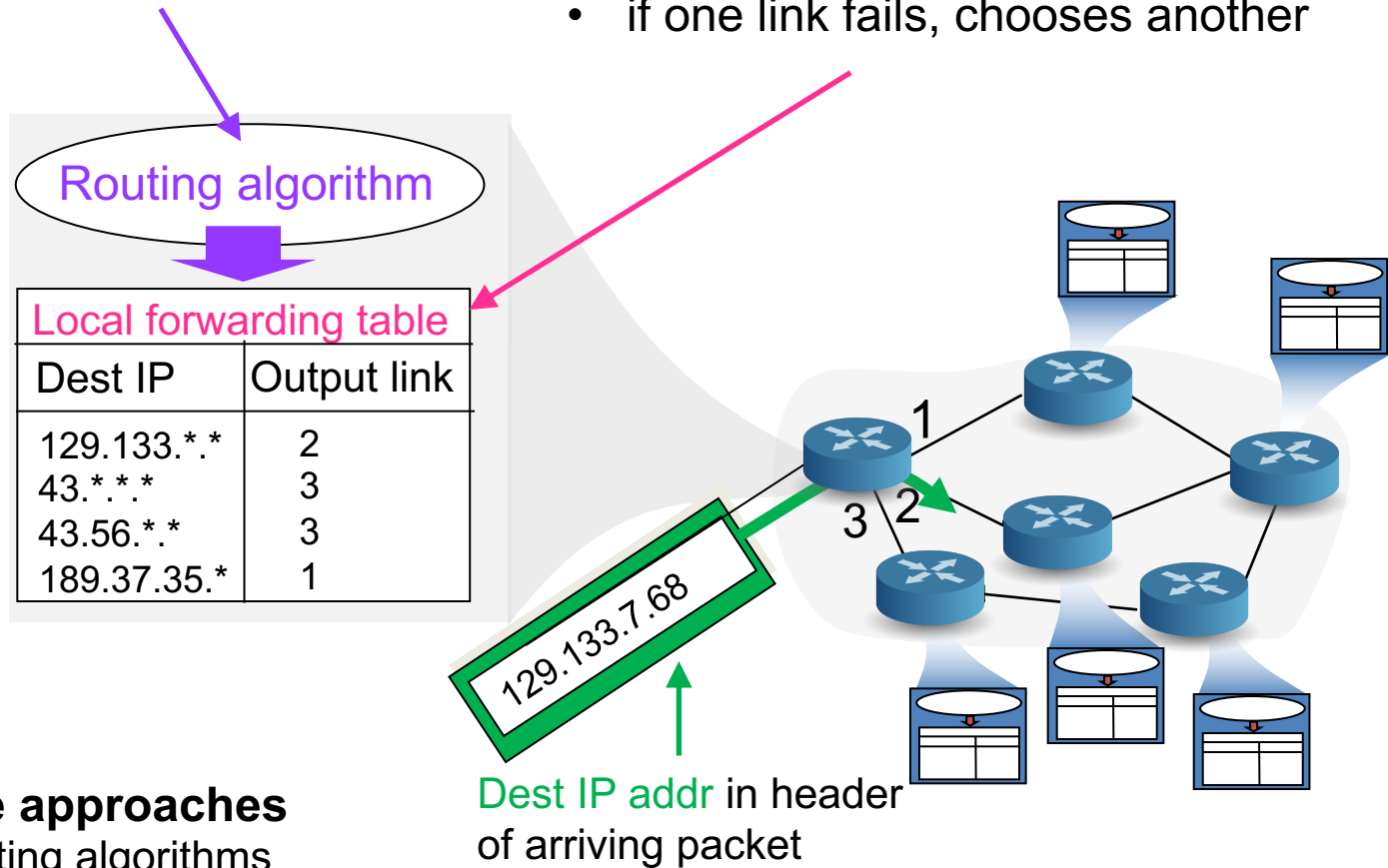
Interplay between routing and forwarding

Routing (slower time scale)

- routers view Internet as **graph**
- run **shortest path algorithms**

Forwarding (faster time scale)

- routers use paths to choose best **output link** for packet **destination IP address**
- if one link fails, chooses another

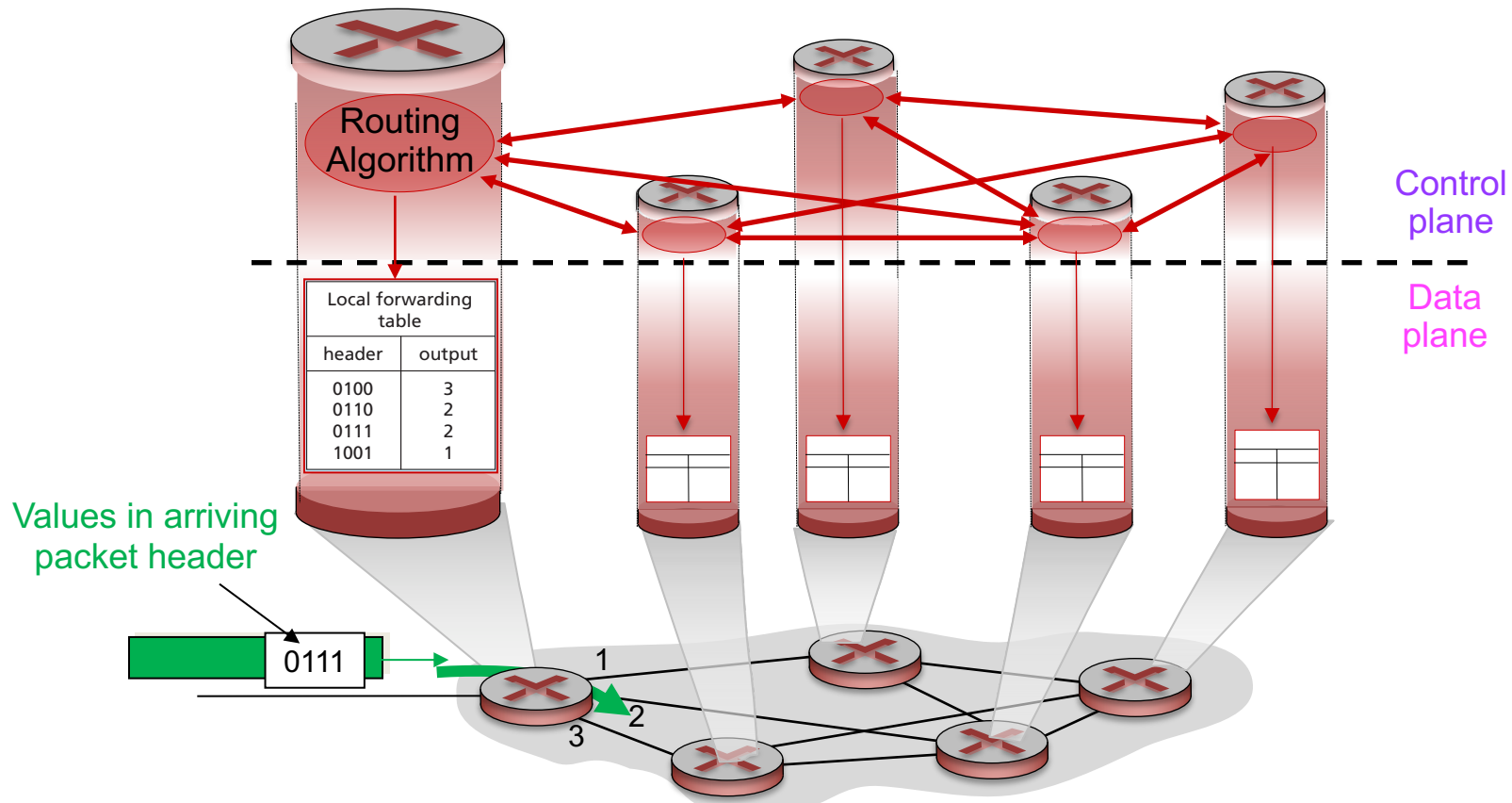


2 control-plane approaches

1. traditional routing algorithms implemented in routers
2. software-defined networking (SDN) implemented in (remote) servers

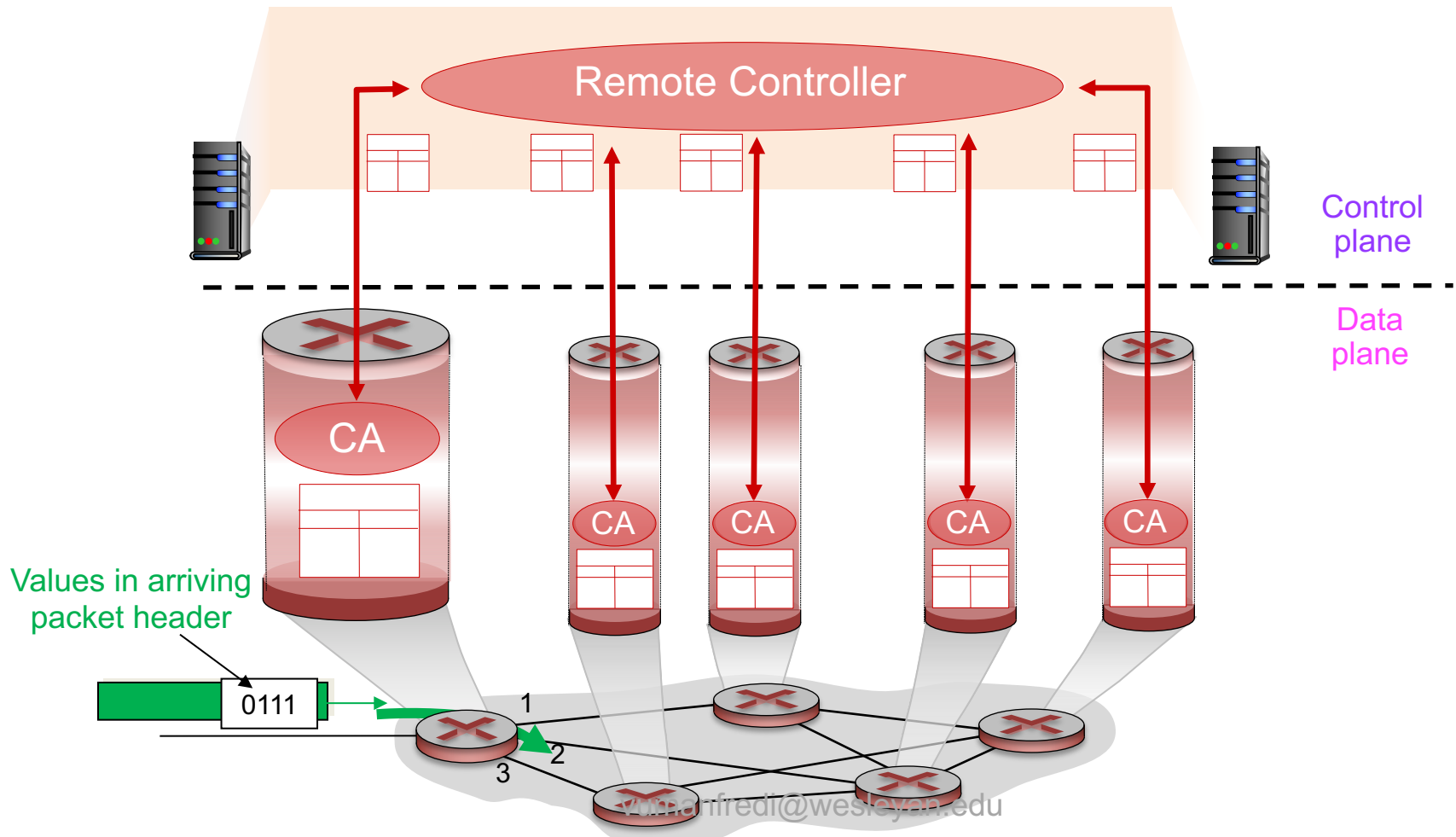
Approach 1: per-router control plane

Individual routing algorithm **components in each and every router** interact in the control plane



Approach 2: logically centralized control plane

A distinct (typically remote) controller interacts with local control agents (CAs)



Network layer service model

Q: What **service model** does network layer provide to transport layer for moving packets from sender to receiver?

Example services

– individual packets

- guaranteed delivery
- guaranteed delivery with less than 40 ms delay

– flow of packets

- in-order packet delivery
- guaranteed minimum bandwidth to flow
- restrictions on changes in inter-packet spacing

Network layer service models

Network Architecture	Service Model	Guarantees ?				Congestion feedback
		Bandwidth	Loss	Order	Timing	
Internet	best effort	none	no	no	no	no (inferred via loss)
ATM	CBR	constant rate	yes	yes	yes	no congestion
ATM	VBR	guaranteed rate	yes	yes	yes	no congestion
ATM	ABR	guaranteed minimum	no	yes	no	yes
ATM	UBR	none	no	yes	no	no

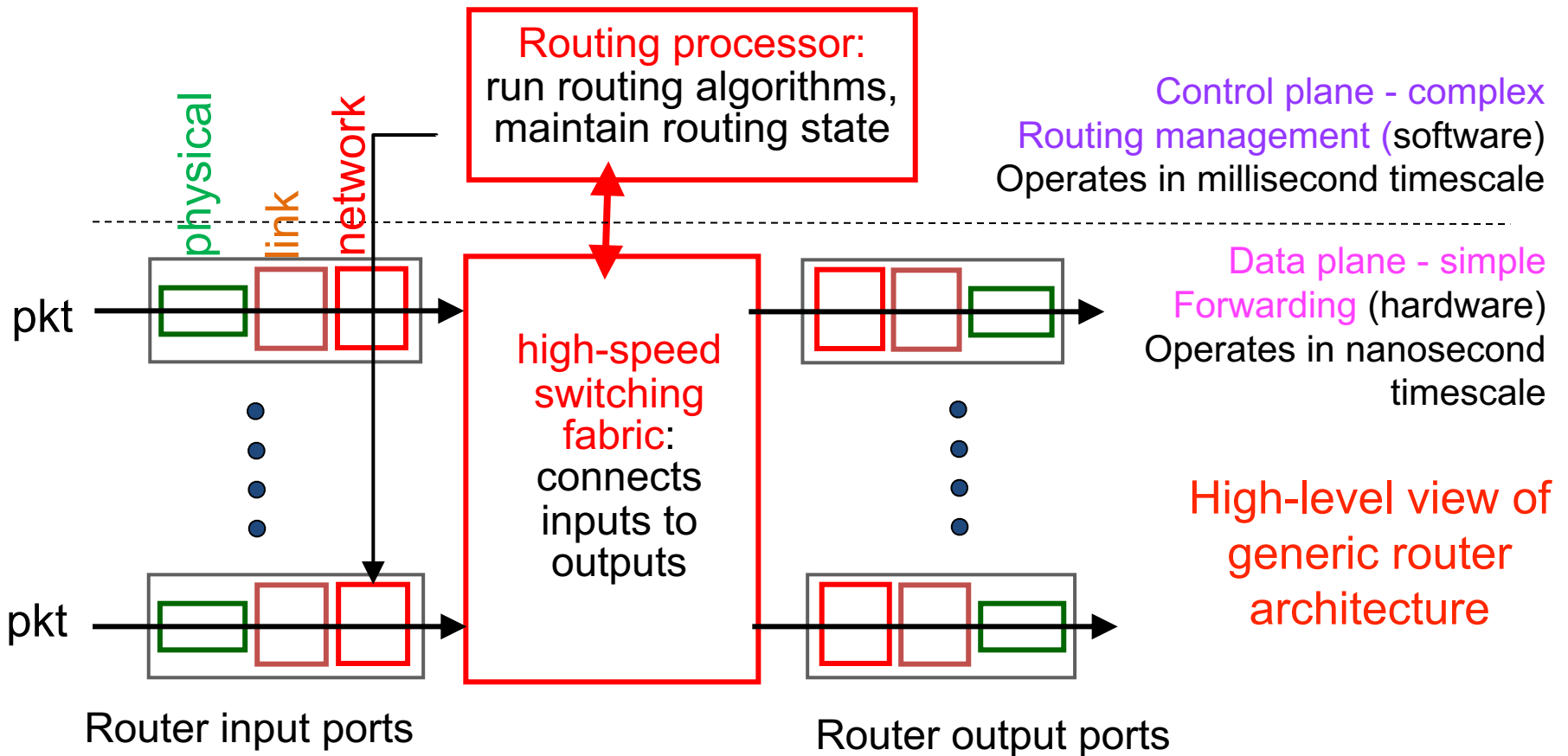
ATM: Asynchronous Transfer Mode
e.g., used in public switched telephone network

Network Layer

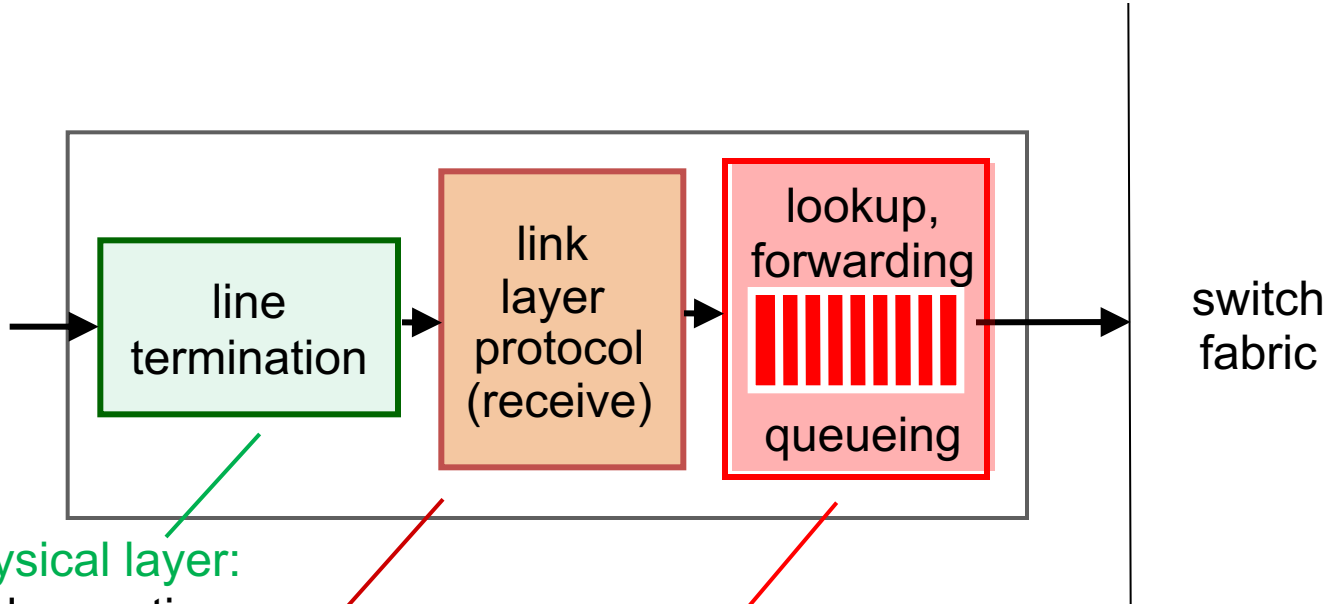
WHAT'S INSIDE A ROUTER?

What does a router need to do?

Run routing protocols (control) and store and forward pkts (data)



Input port functions



Physical layer:
bit-level reception,
terminate phys. conn.

Data link layer:
e.g., Ethernet processing,
error-checking, de-capsulation,

Network layer

- validate/update checksum, decrement TTL
- **switching**: use header field values, lookup output port
- **queue**: if packets arrive faster than forwarding rate into switch fabric

Switching fabrics

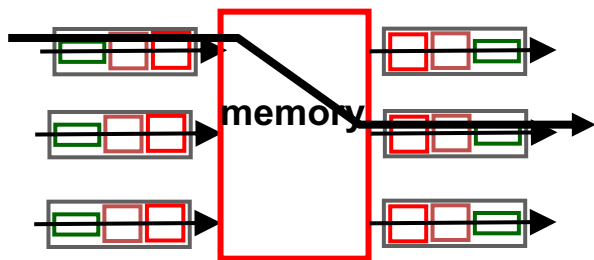
Transfer packet

- from **input** buffer to appropriate **output** buffer

Switching rate

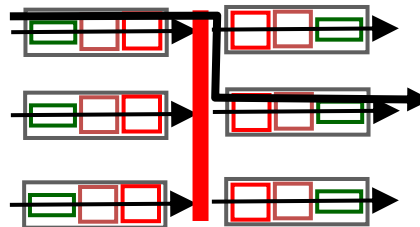
- rate at which packets can be **transferred** from inputs to outputs
- N inputs: switching rate = N x line rate desirable

3 types of switching fabrics



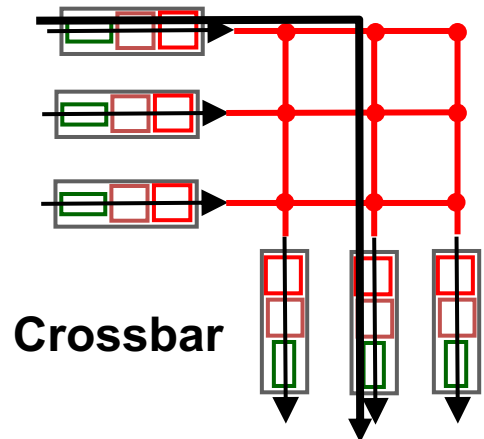
Memory

Speed limited by
memory bandwidth



Bus

Speed limited by
bus contention



Crossbar

Forward multiple
pkts in parallel

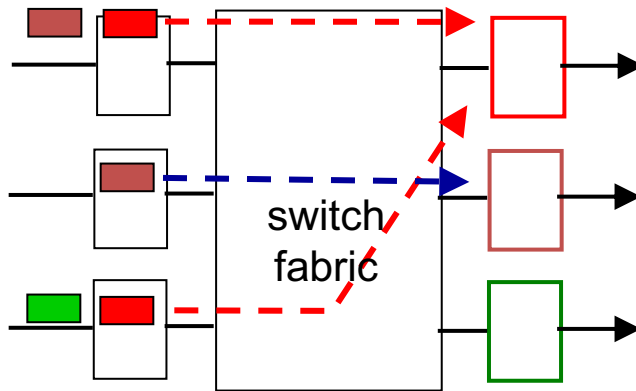
Contention at input ports

If switching fabric slower than input ports combined

- queueing may occur at input queues
- queueing delay and loss due to input buffer overflow!

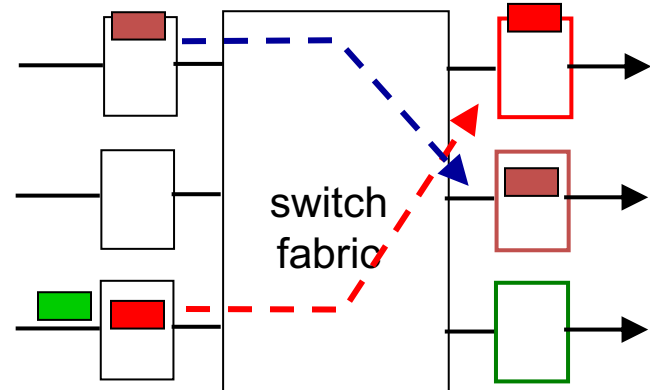
Head-of-the-Line (HOL) blocking

- queued pkt at front of queue prevents others from moving forward



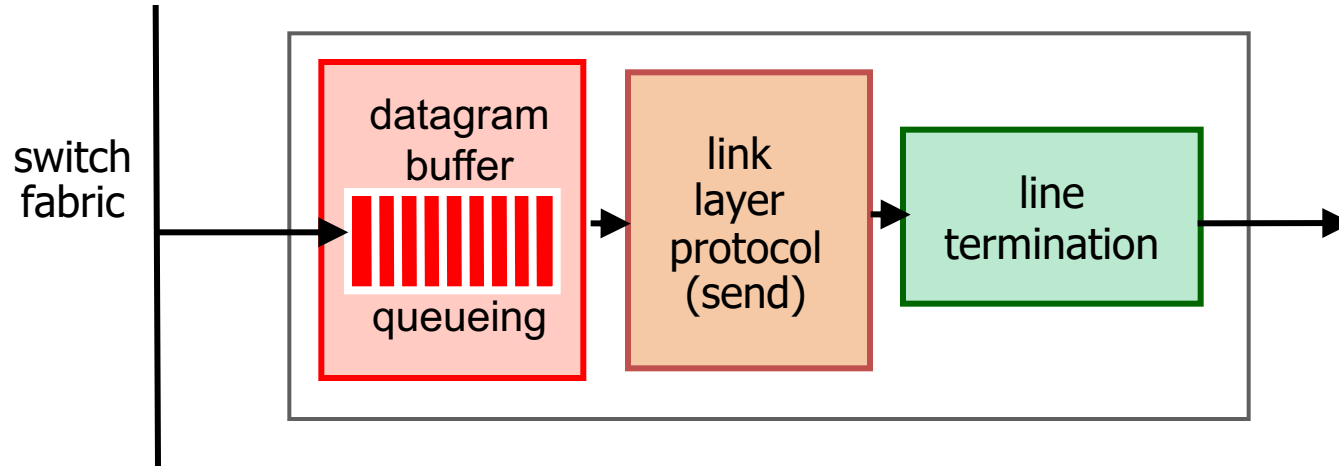
Output port contention: only one red packet can be transferred.

Lower red packet is blocked



One packet time later: green packet experiences HOL blocking

Contention at output ports



Buffering

- when packets arrive from fabric faster than transmission rate
- **packet loss**: due to congestion, lack of buffers

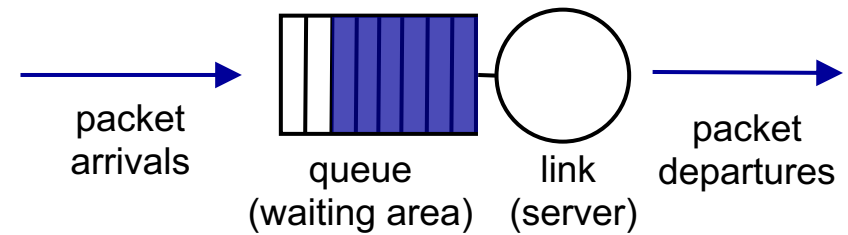
Scheduling

- chooses next among queued packets to transmit on link
- **net neutrality**: who gets best performance

Scheduling mechanisms

FIFO (first in first out)

- send in order of arrival to queue



Priority

- multiple classes, with different priorities (e.g., based on hdr info)
 - send highest priority queued packet

Round robin scheduling

- multiple classes, cyclically scan class queues
 - send one packet from each class (if available)

Weighted fair queueing

- generalized round robin
 - each class gets weighted amount of service in each cycle

In practice: hardware queues use FIFO,
need software to do priority

Network Layer

INTERNET PROTOCOL

Internet Protocol (IP)

THE network layer protocol of the Internet

- protocol your device **must** implement to run on Internet
- RFC published ~1980

Provides

- best effort service
 - to get pkts from one end host to another across many interconnected networks using dst IP address in IP hdr
- addressing
 - format and usage of addresses
- fragmentation
 - e.g., if pkt size exceeds Ethernet MTU of 1500 bytes
- some error detection

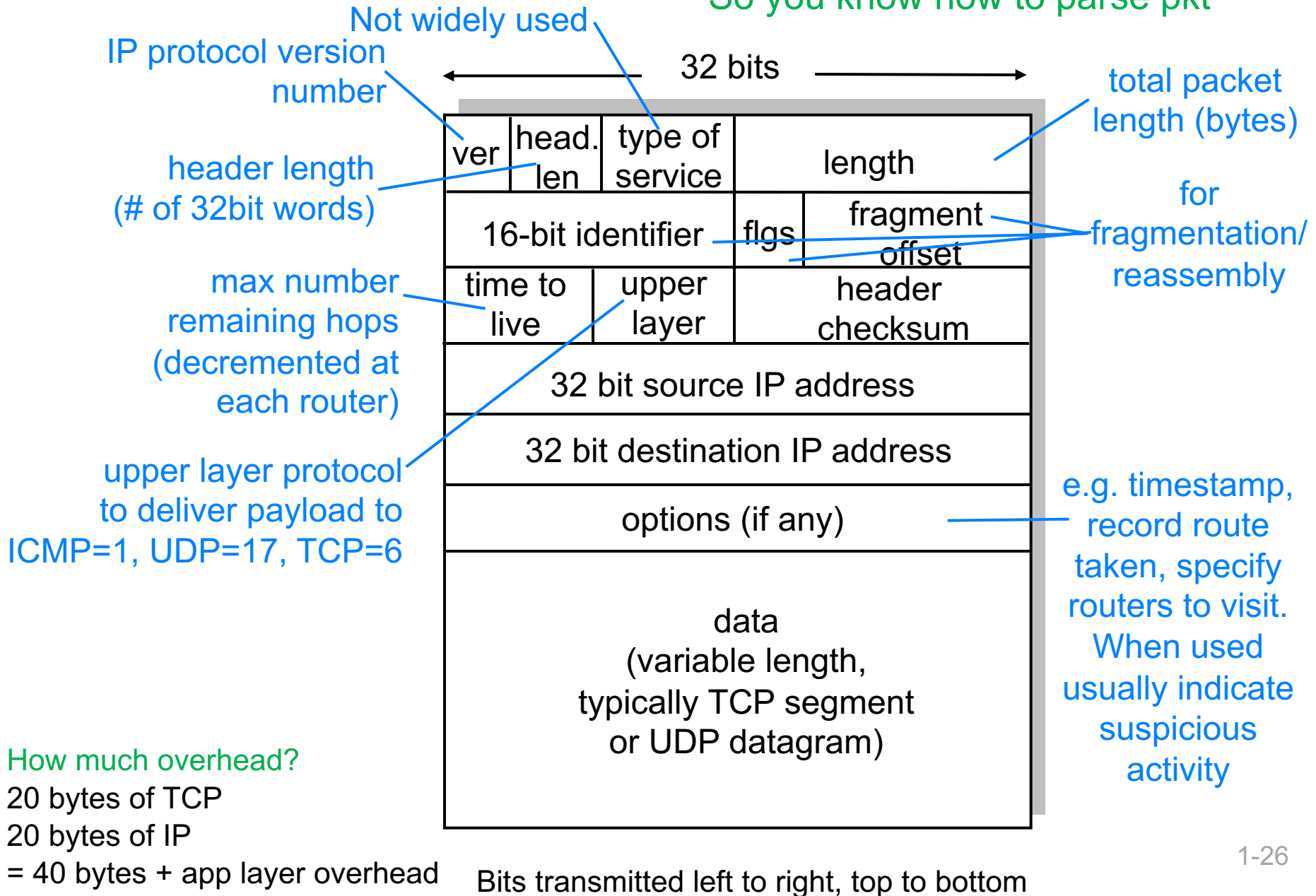
Q: what does IP not provide?

- QoS, reliability, ordering, persistent state for e2e flows, connections

IP packet format

Q: Why is version 1st?

So you know how to parse pkt



Wireshark

Look at IP headers and ping/traceroute