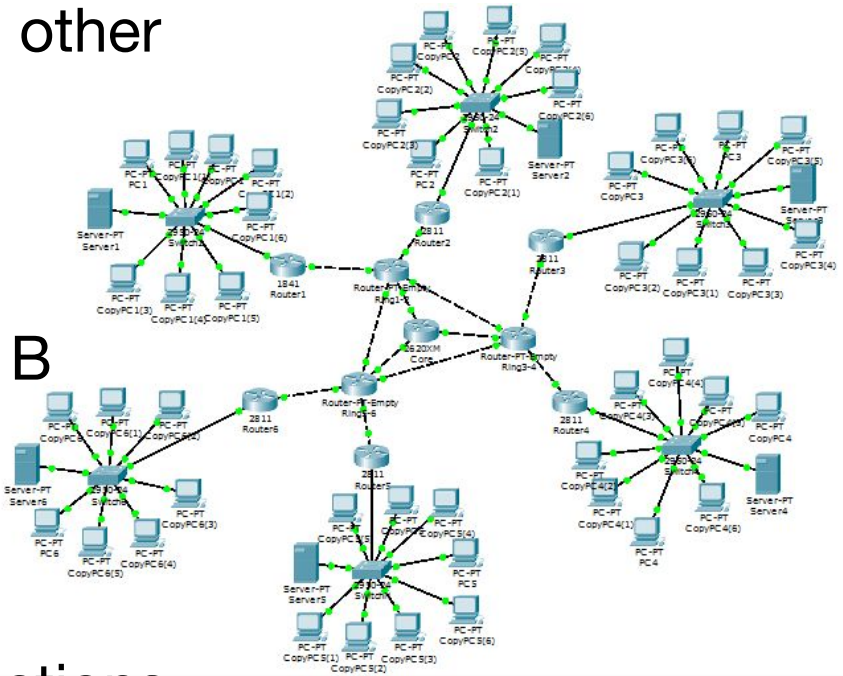
Networking Basics

With slides from

- Dave Levin 414-spring2016
- Michelle Mazurek 414-fall2016

The network is “dumb”

- End-hosts on periphery of network
  - Not physically connected to each other
  - But can still communicate!
- Routers: Interior nodes that
  - “Route”: *determine how* to get to B
  - “Forward”: traffic from A to B
- Principle: Routers are stateless
  - Don’t know about ongoing connections
  - Send each packet to best “next hop” known



Weak analogy: The postal system

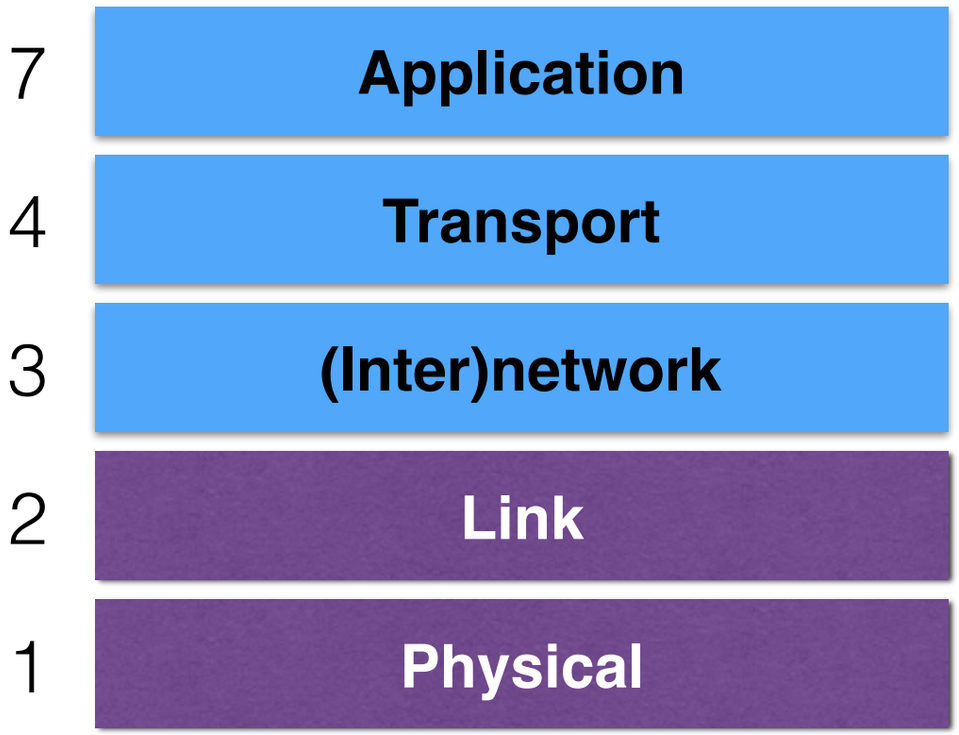
Postal system analogy

- Messages are self-contained
  - Mail: Content in an envelope
  - Internet: Data in a packet
- Routers forward based on destination address
  - Post: zip code, then street, then building, then apartment number (then the right individual)
  - Internet: Progressively smaller blocks of IP addresses, then your computer (then the right application)
- Simple, robust
  - More sophisticated things at the *ends of the network*

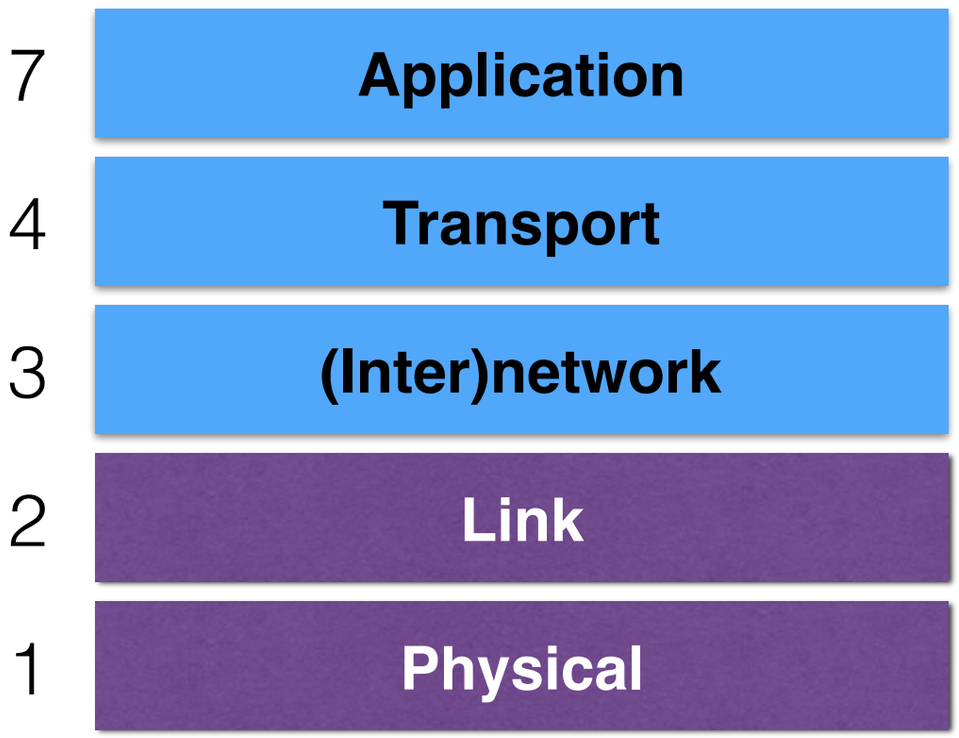
Internet layers

- The Internet is structured in layers
- Each layer uses the services provided by the layer below and provides service to the layer above
- Protocol: interactions between nodes in a layer
  - msg format, what to do at sends and receives
  - eg: IP, TCP, http
- Interface: interactions between layers in a node
  - functions called and values returned
  - eg: networking sockets

Internet layering = “Protocol stack”



Layer 1: Physical layer

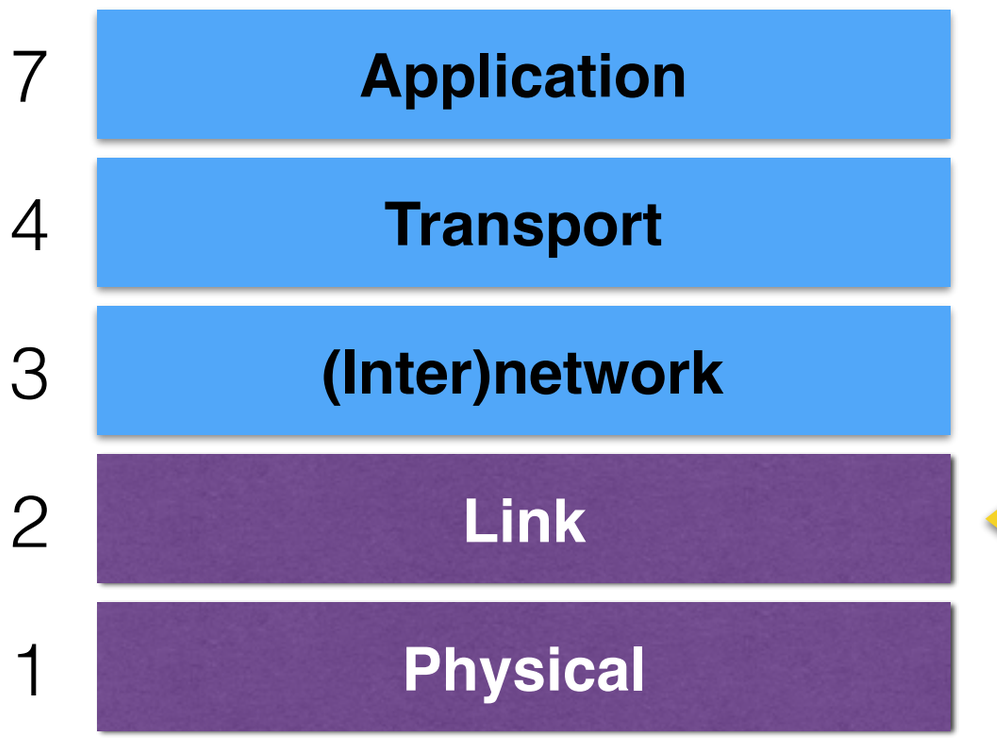


- Encoding of bits to send over a ***single*** physical link
- Examples:
  - Voltage levels
  - RF modulation
  - Photon intensities

Physical layer:
transmitting a single bit
over a physical link
(though not necessarily *wired* link)

End-host C

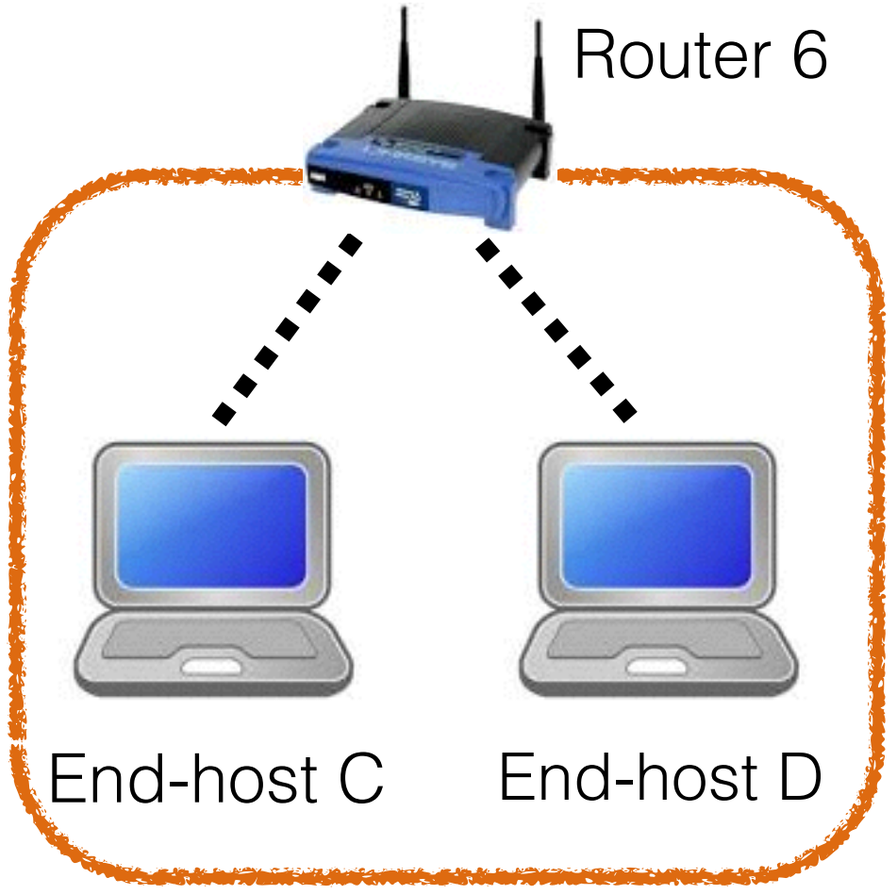
Layer 2: Link layer



- Framing and transmission of a collection of bits into individual **messages** sent across a single subnetwork (one physical topology)
- Provides **local** addressing (MAC)
- May involve multiple *physical links*
- Often the technology supports **broadcast**: every "node" connected to the subnet receives
- Examples:
  - Modern Ethernet
  - WiFi (802.11a/b/g/n/etc)

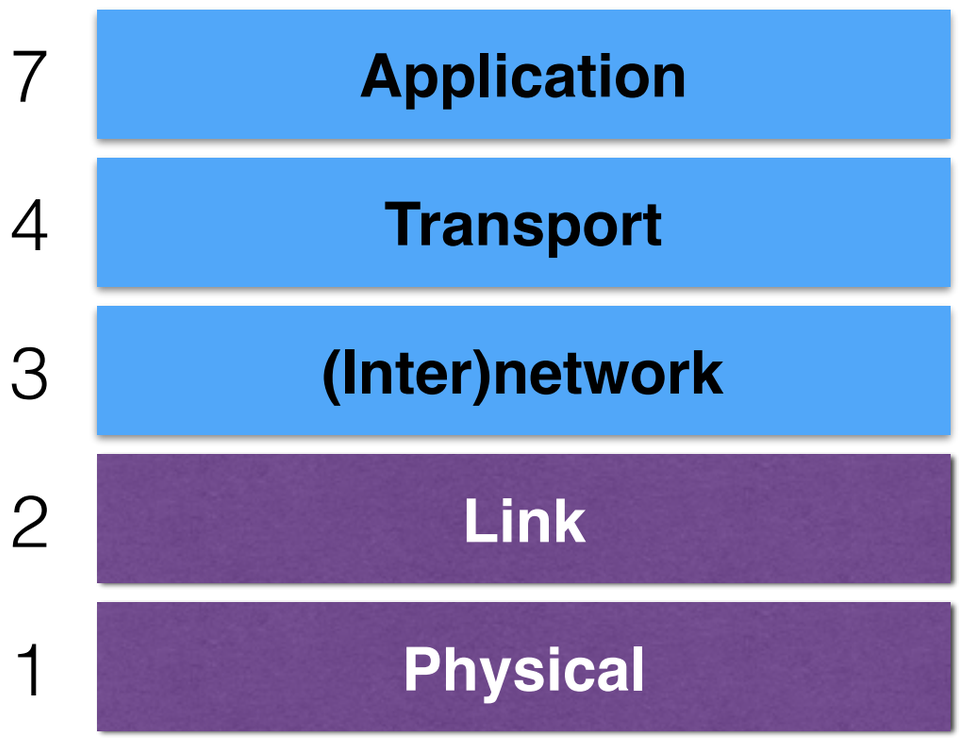
Link layer

- transmitting messages
- over a *subnet*
- src/dst identified by globally unique **MAC addrs**



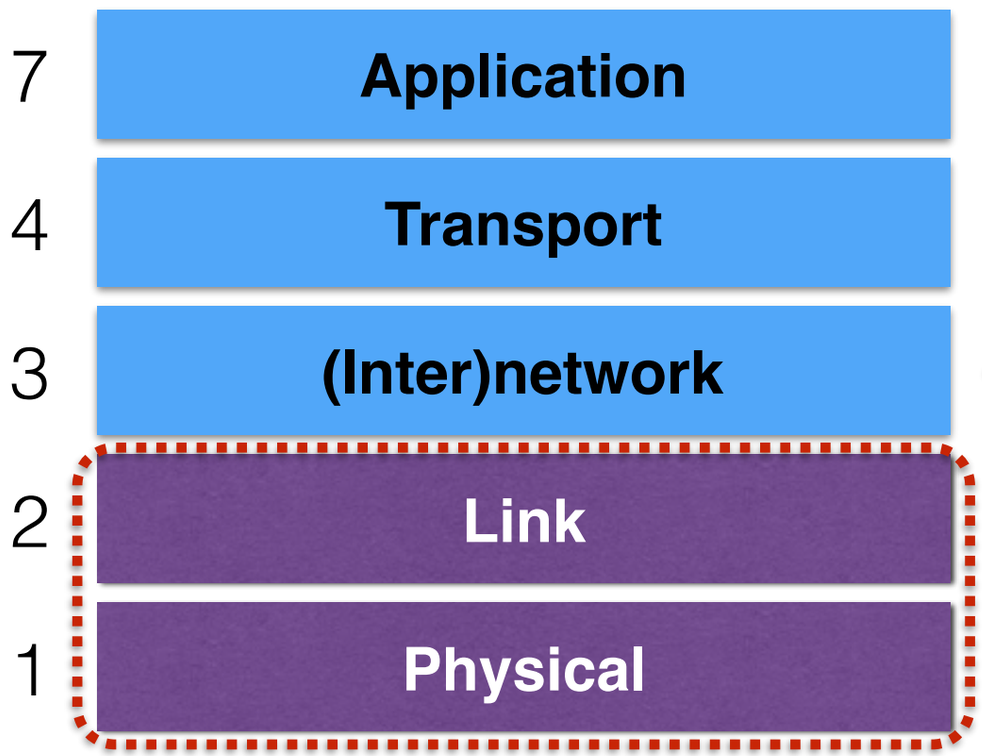
Because you need to be able to join any subnet and be uniquely distinguishable

Layer 3: (Inter)network layer



- Bridges multiple “subnets” to provide *end-to-end* internet connectivity between nodes
- Provides **global** addressing (IP addresses)
- Only provides **best-effort** delivery of data (i.e., no retransmissions, etc.)
- Works across different link technologies

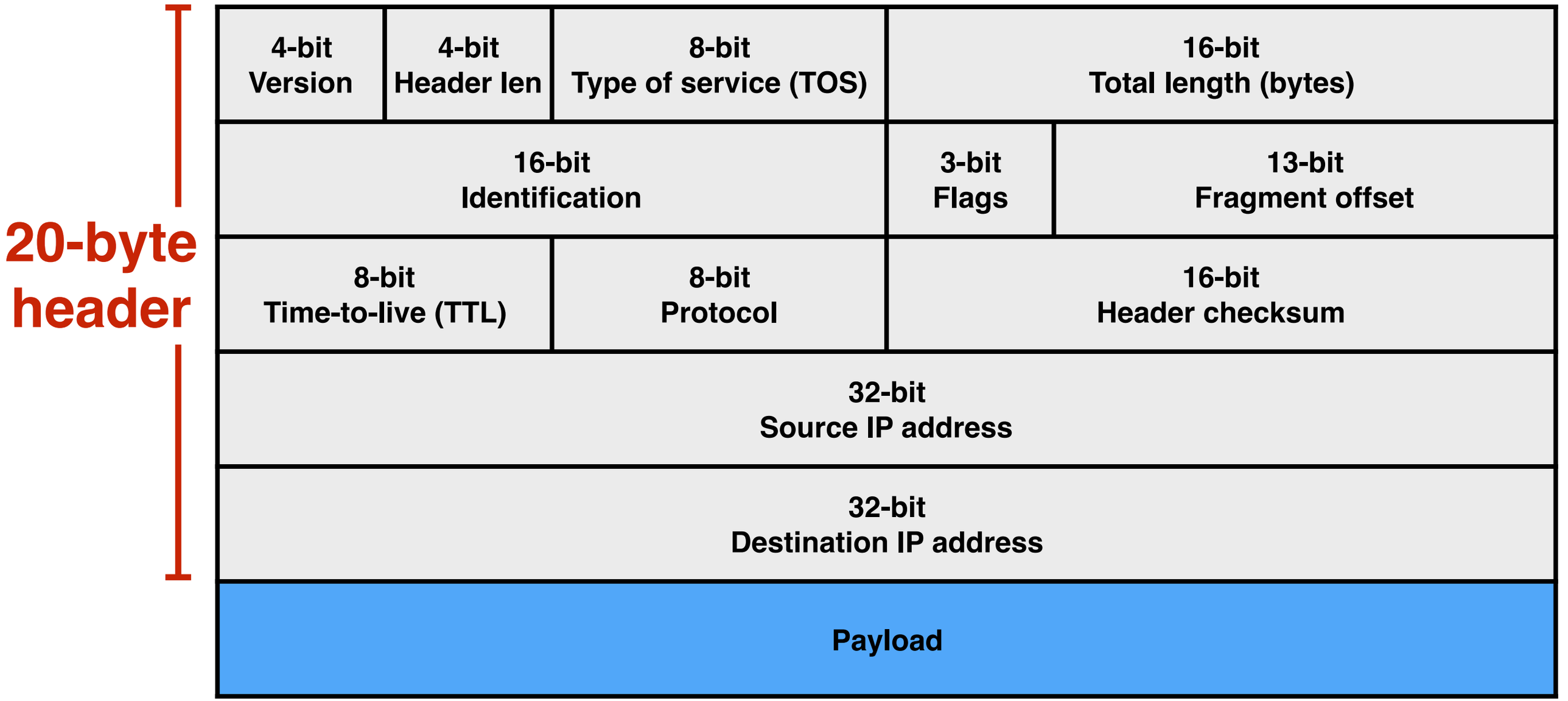
Layer 3: (Inter)network layer



Different for each Internet "hop"

- Bridges multiple "subnets" to provide *end-to-end* internet connectivity between nodes
- Provides **global** addressing (IP addresses)
- Only provides **best-effort** delivery of data (i.e., no retransmissions, etc.)
- Works across different link technologies

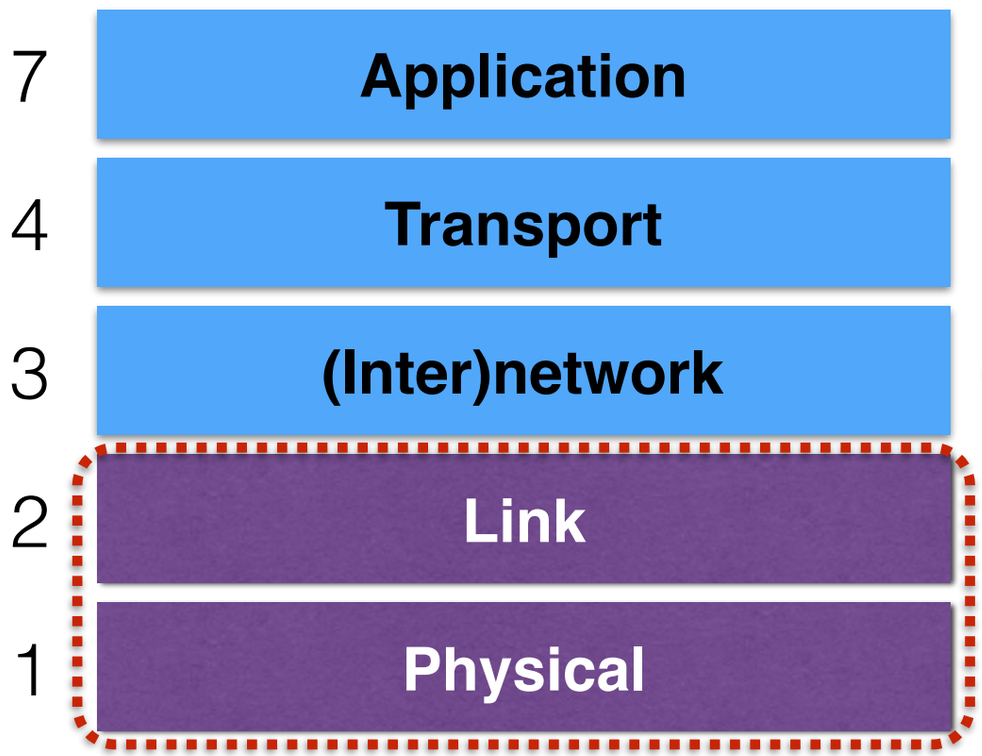
IP packet “header”



The payload is the “data” that IP is delivering:

May contain another protocol’s header & payload, and so on

Layer 3: (Inter)network layer



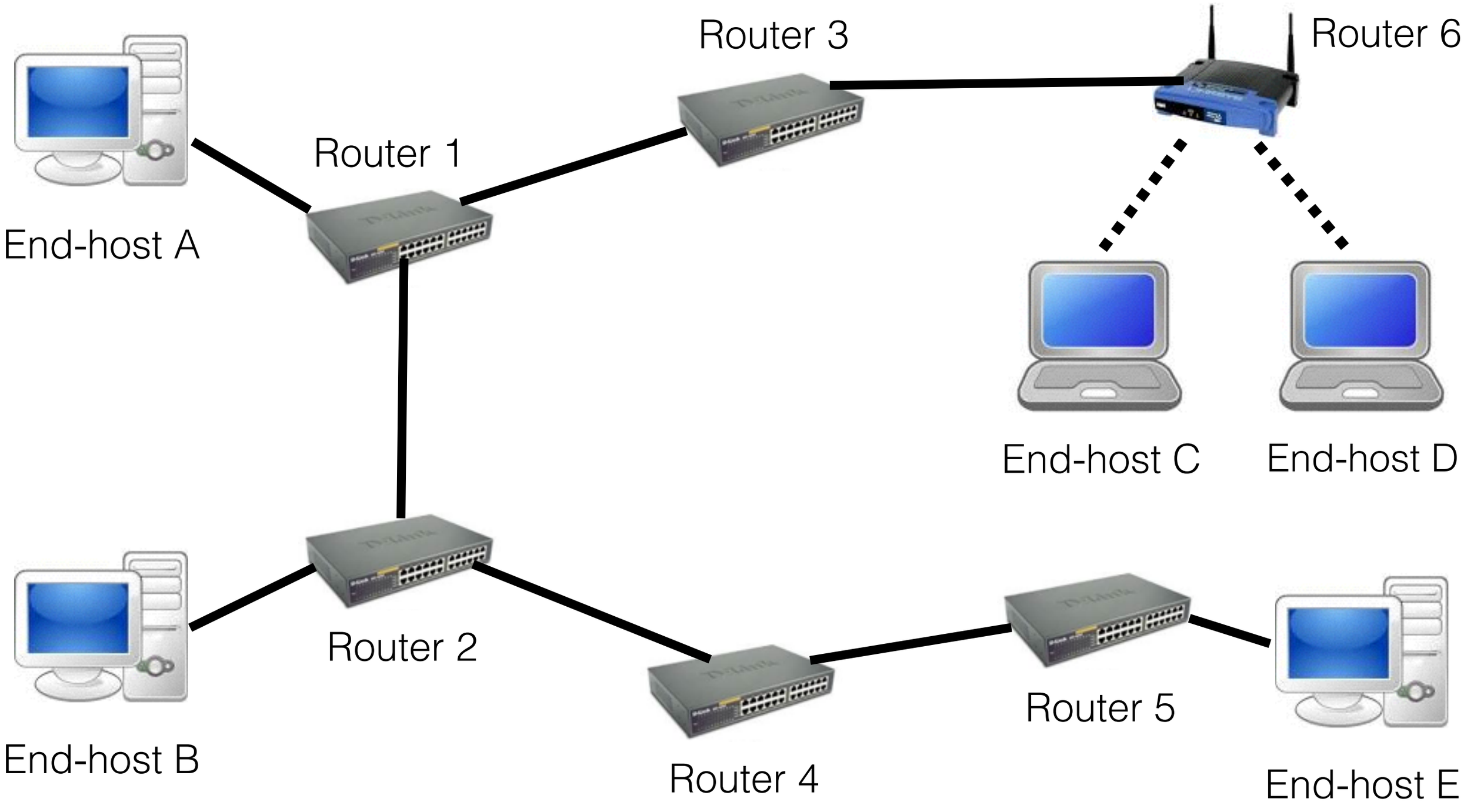
Different for each Internet "hop"

- Bridges multiple "subnets" to provide *end-to-end* internet connectivity between nodes
- Provides **global** addressing (IP addresses)
- Only provides **best-effort** delivery of data (i.e., no retransmissions, etc.)
- Works across different link technologies

Lowercase-i "internet" = network of networks.
Uppercase-i Internet = "*the* Internet"

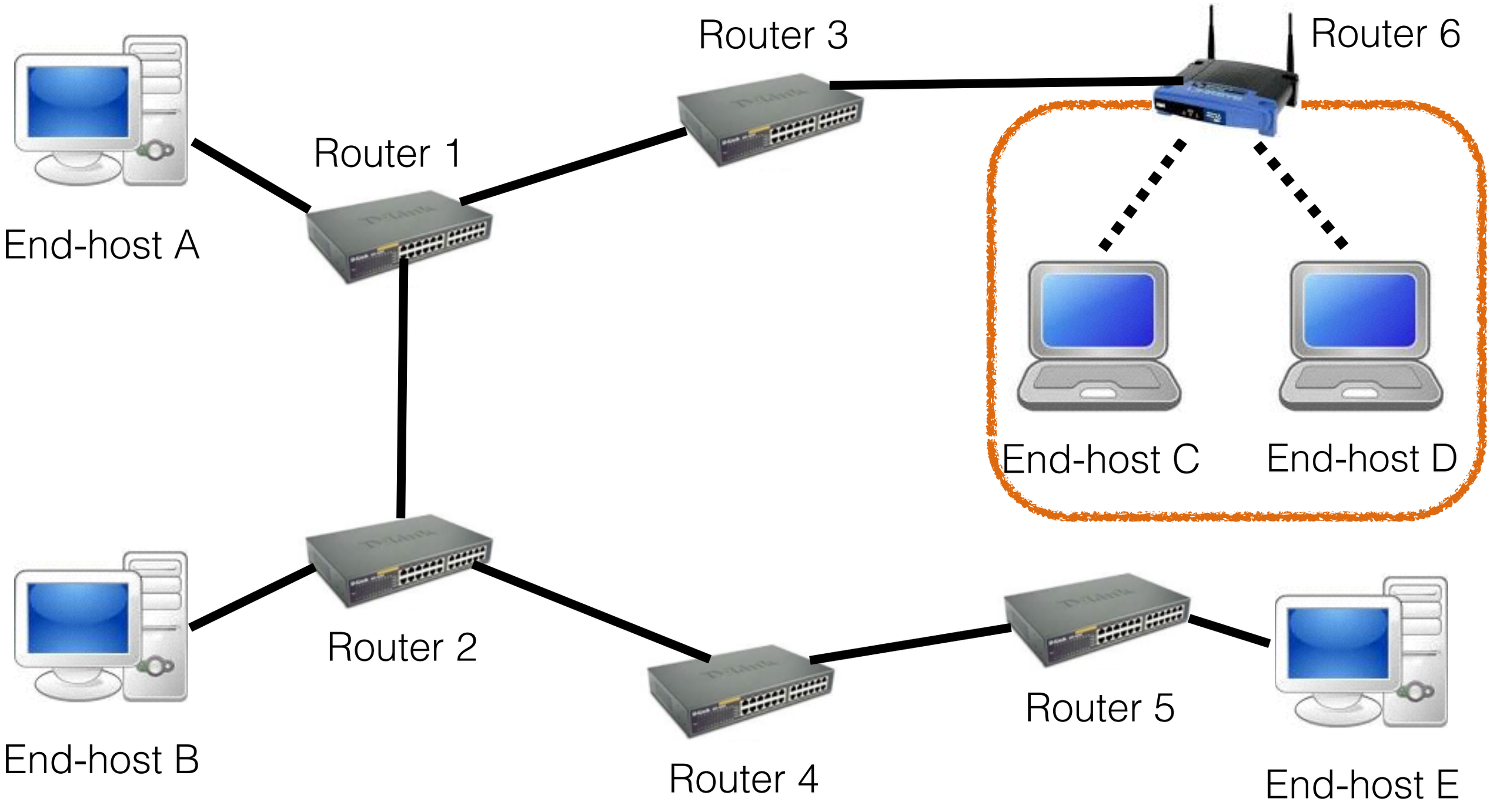
Network layer

- transmitting packets
- within or across subnets
- src/dst identified by locally unique **IP addrs**



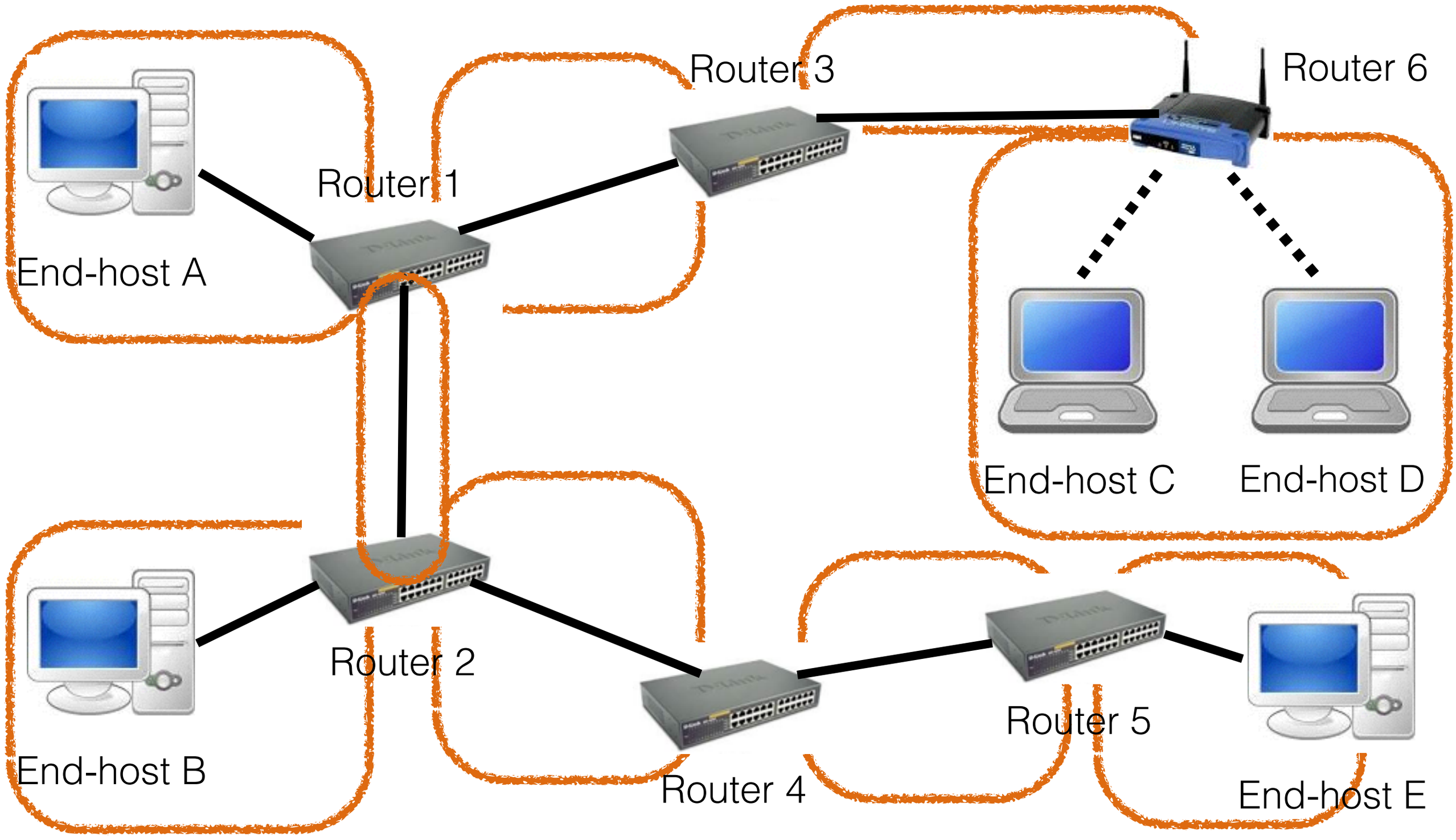
Network layer

- transmitting packets
- within or across subnets
- src/dst identified by locally unique **IP addrs**



Network layer

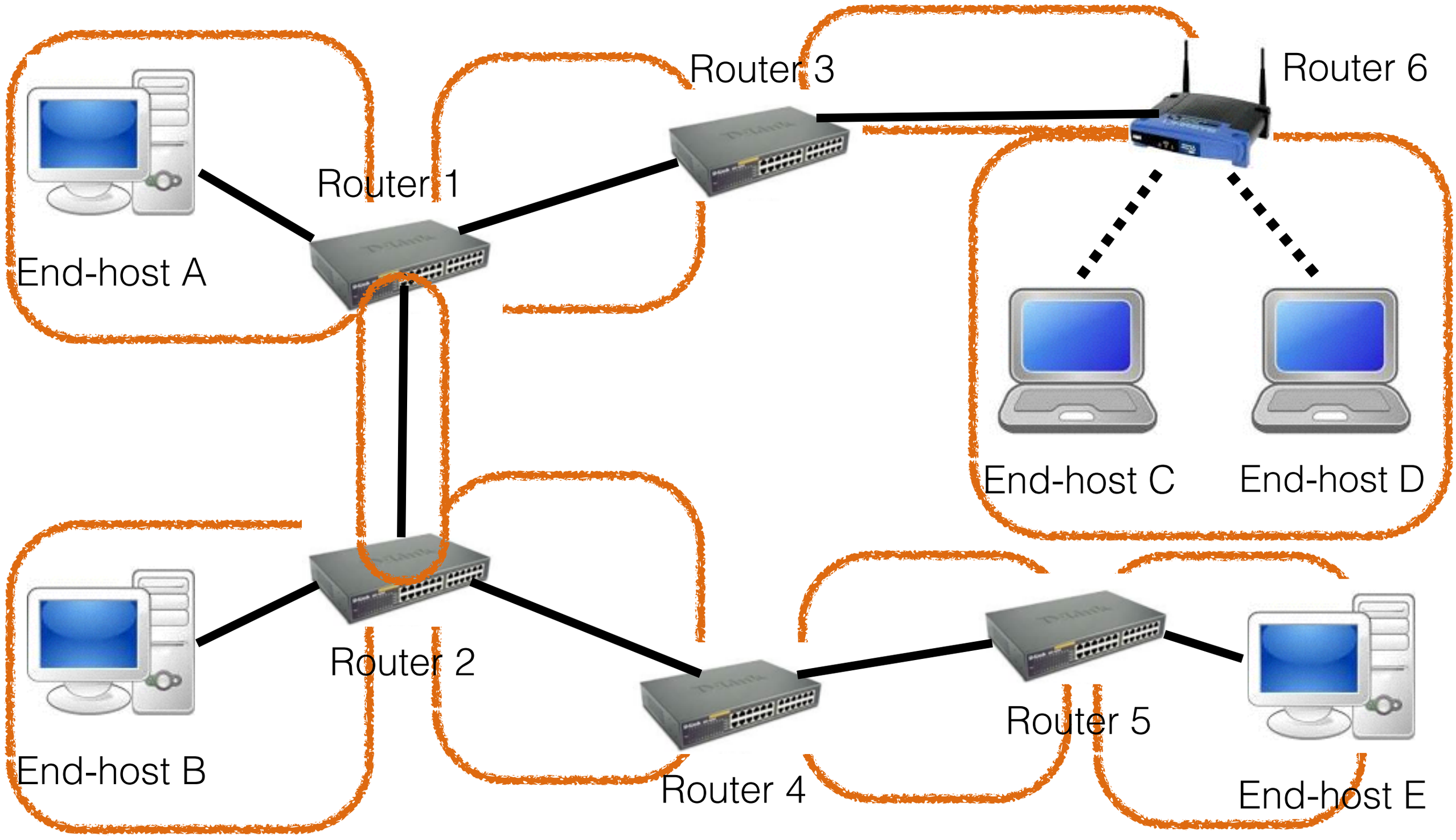
- transmitting packets
- within or across subnets
- src/dst identified by locally unique **IP addrs**



Network layer

- transmitting packets
- within or across subnets
- src/dst identified by locally unique **IP addrs**

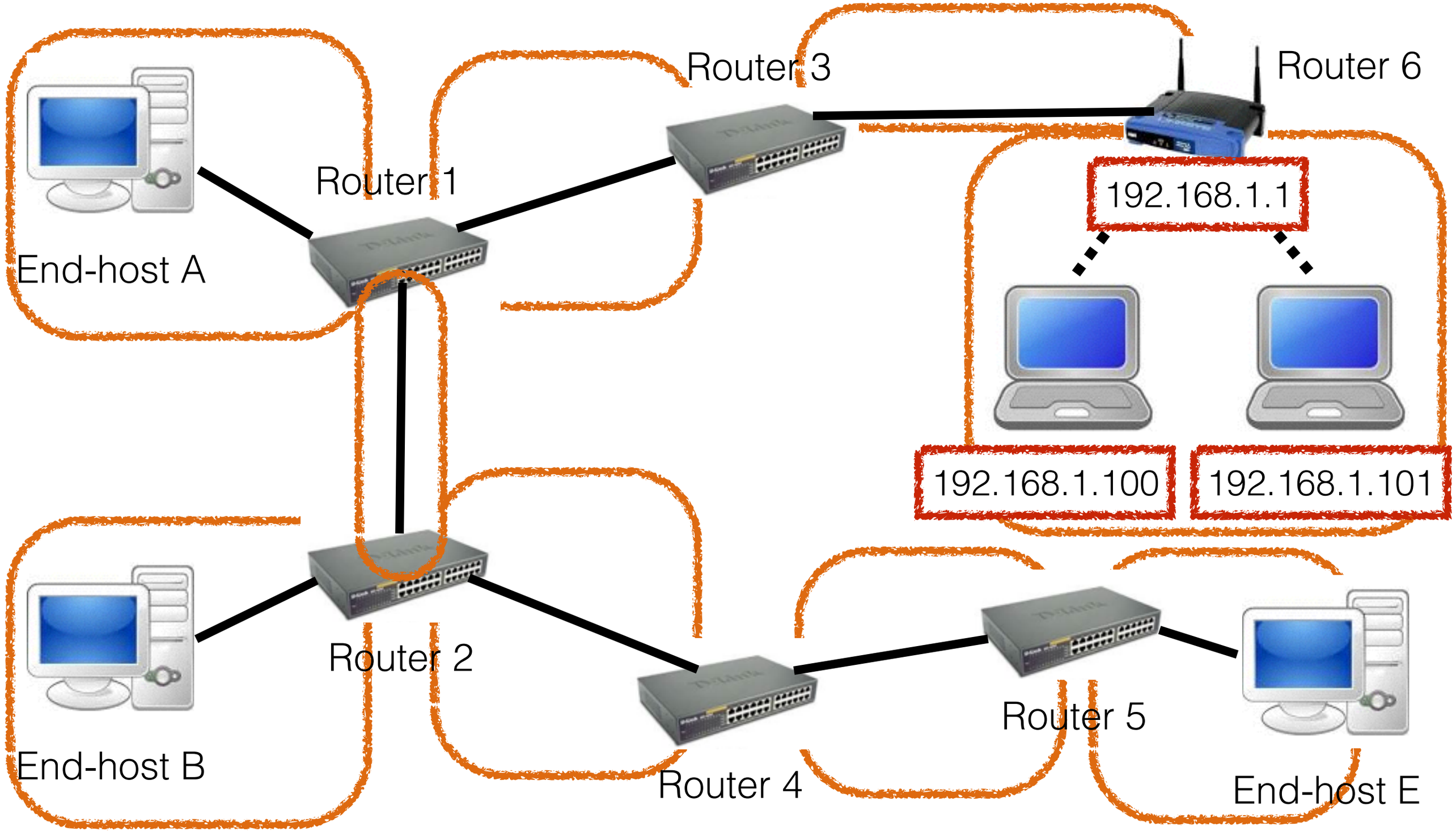
Routers connect multiple subnets



Network layer

- transmitting packets
- within or across subnets
- src/dst identified by locally unique **IP addrs**

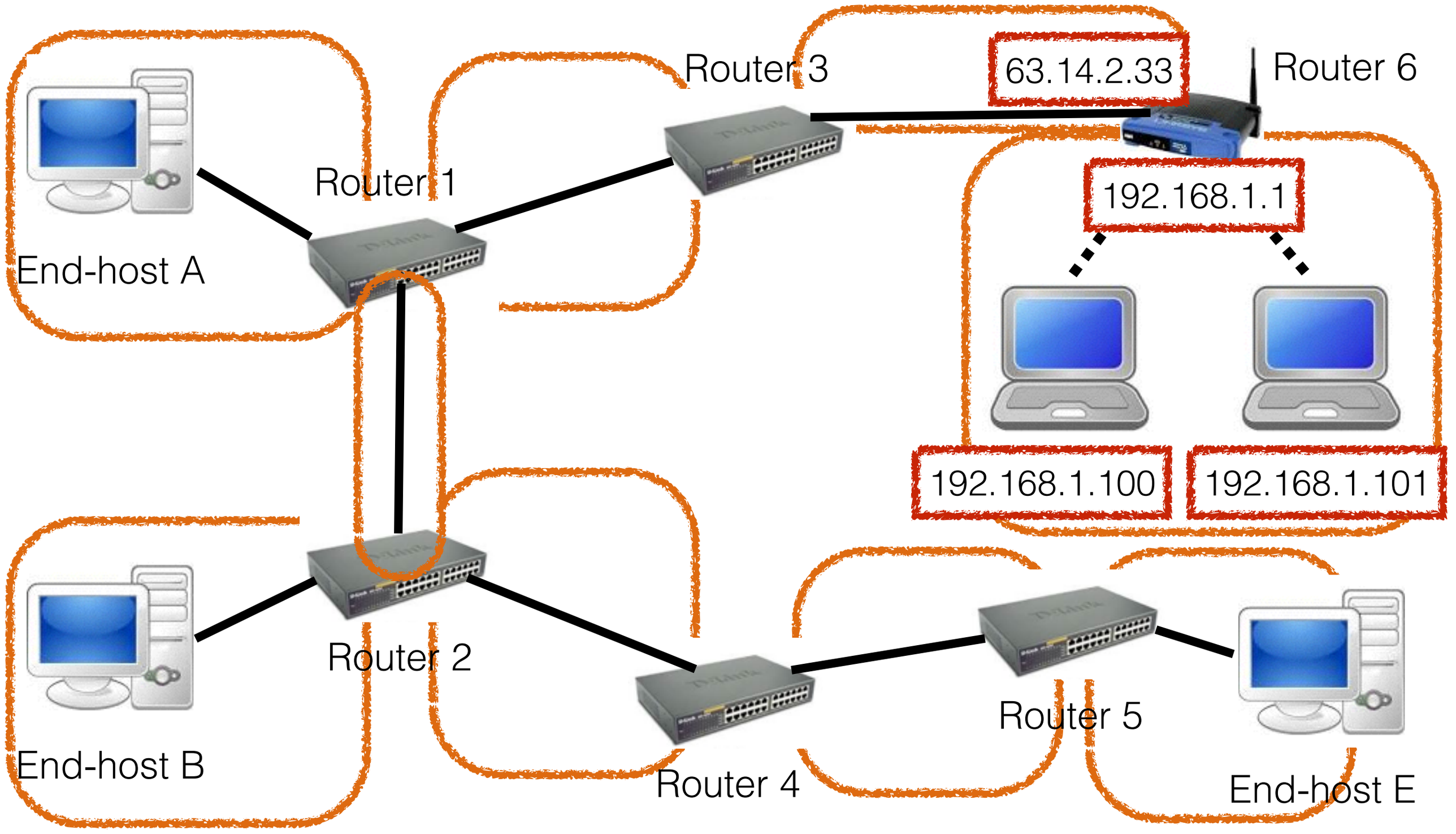
Routers connect multiple subnets



Network layer

- transmitting packets
- within or across subnets
- src/dst identified by locally unique **IP addrs**

Routers connect multiple subnets

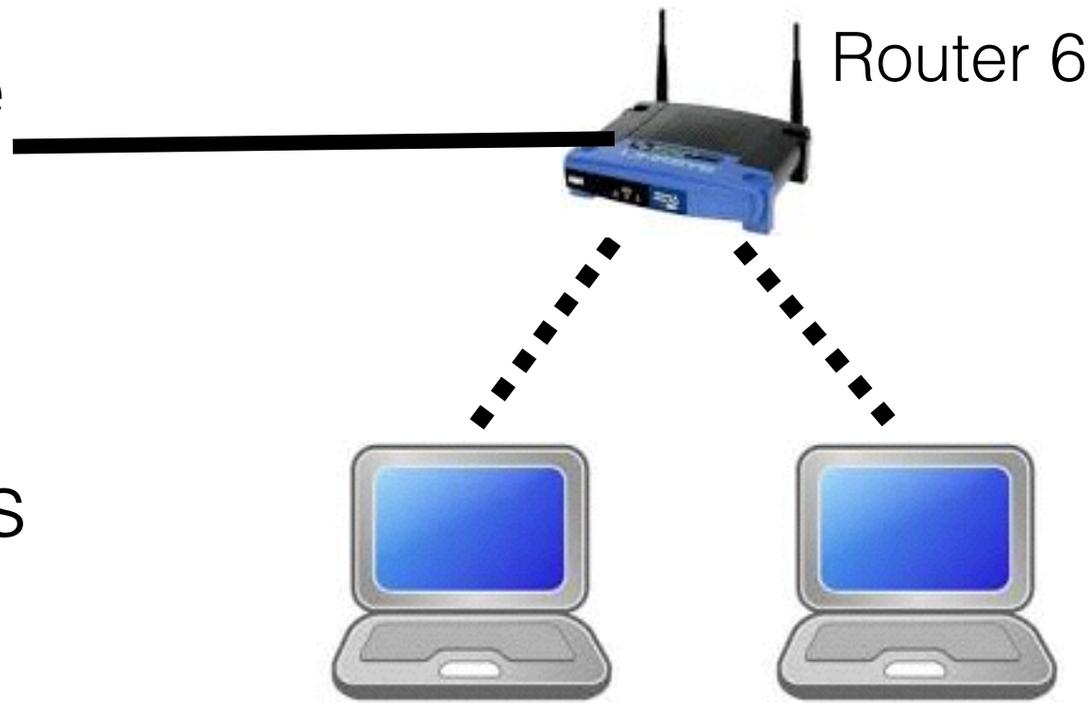


Local uniqueness is often enough

There are only 2^32 IP addrs

Many machines don’t need to be publicly reachable

Some addresses are “private” addresses



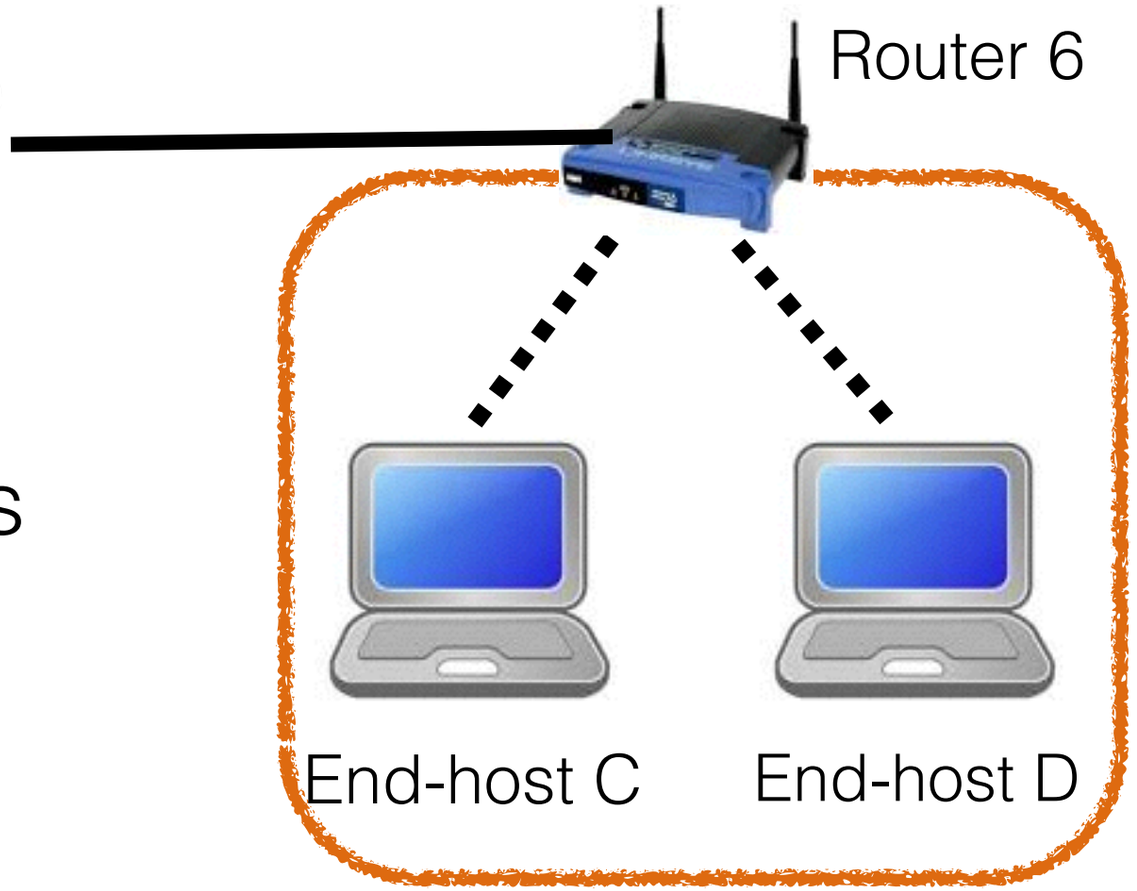
The router performs “**Network Address Translation**”: changes outgoing packets’ src from 192.168.1.100 to 63.14.2.33, and vice versa for incoming packets

Local uniqueness is often enough

There are only 2^32 IP addrs

Many machines don’t need to be publicly reachable

Some addresses are “private” addresses



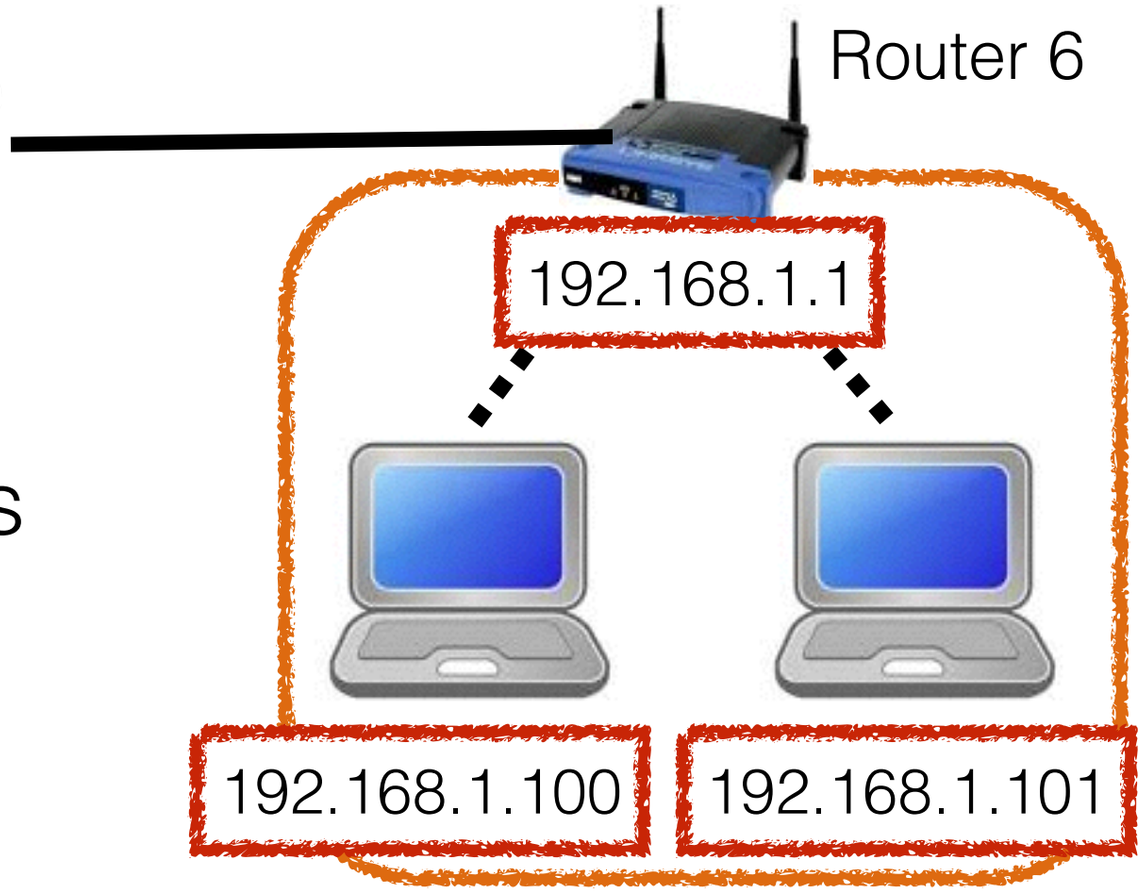
The router performs “**Network Address Translation**”: changes outgoing packets’ src from 192.168.1.100 to 63.14.2.33, and vice versa for incoming packets

Local uniqueness is often enough

There are only 2^32 IP addrs

Many machines don’t need to be publicly reachable

Some addresses are “private” addresses



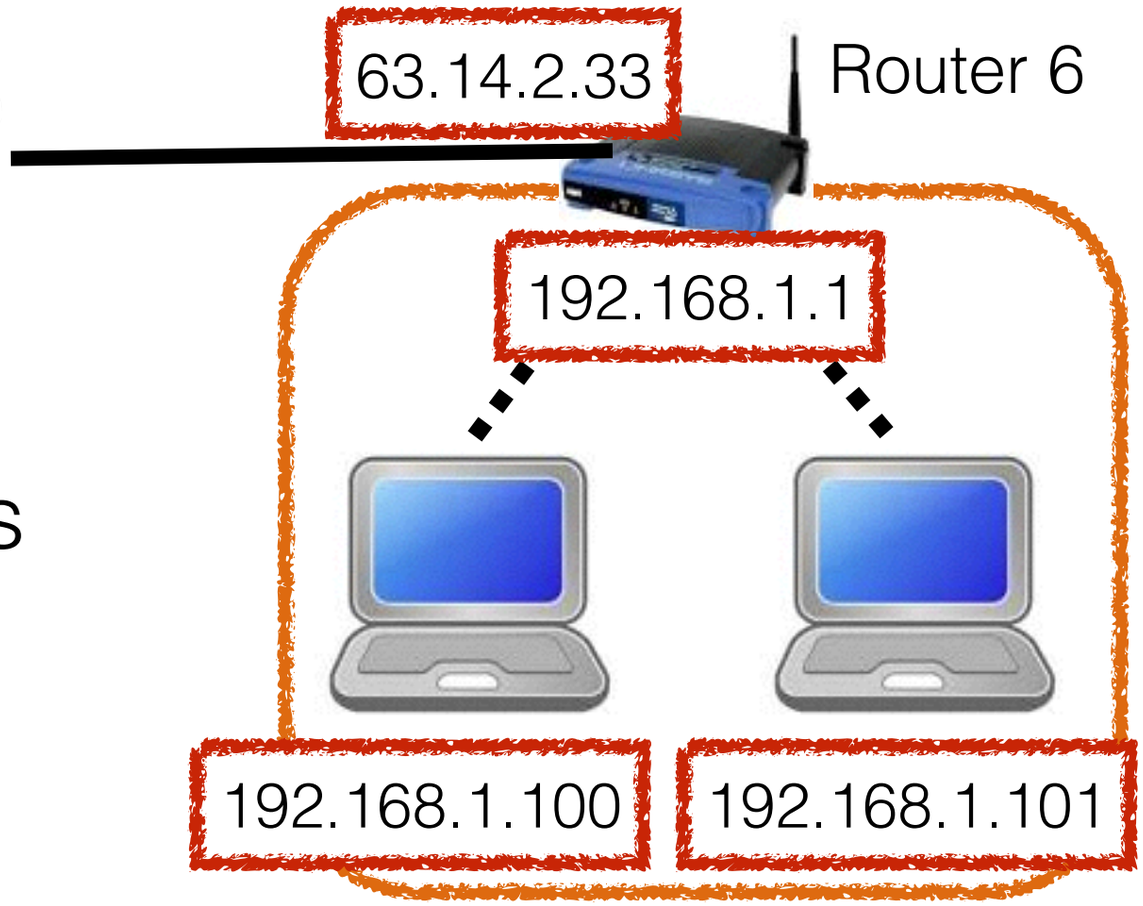
The router performs “**Network Address Translation**”: changes outgoing packets’ src from 192.168.1.100 to 63.14.2.33, and vice versa for incoming packets

Local uniqueness is often enough

There are only 2^32 IP addrs

Many machines don’t need to be publicly reachable

Some addresses are “private” addresses

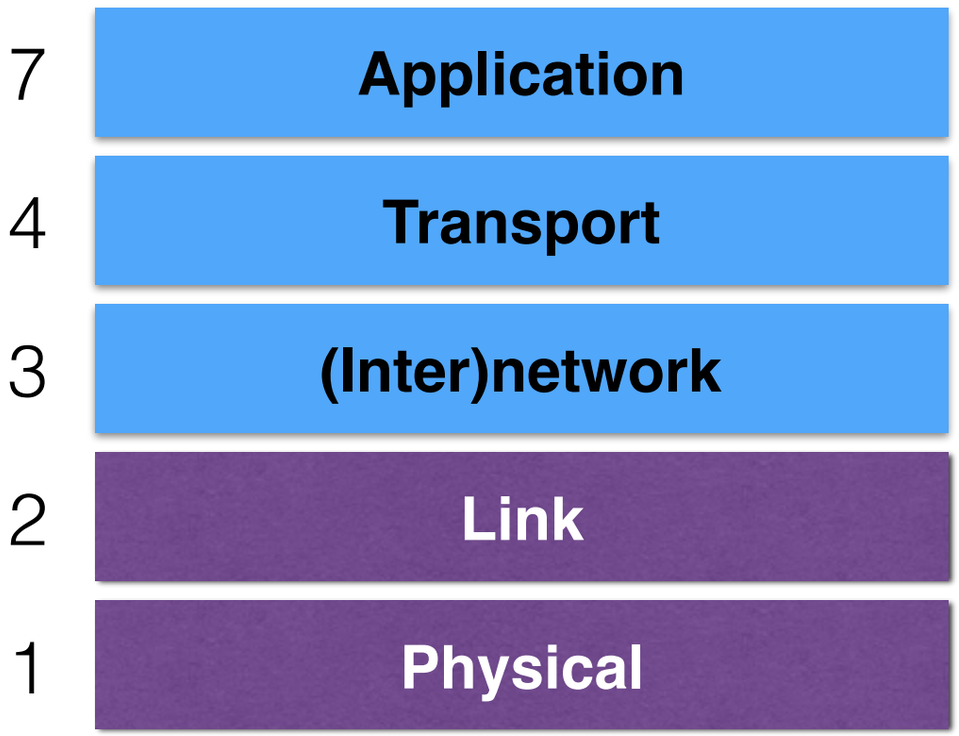


The router performs “**Network Address Translation**”: changes outgoing packets’ src from 192.168.1.100 to 63.14.2.33, and vice versa for incoming packets

Routing vs Forwarding

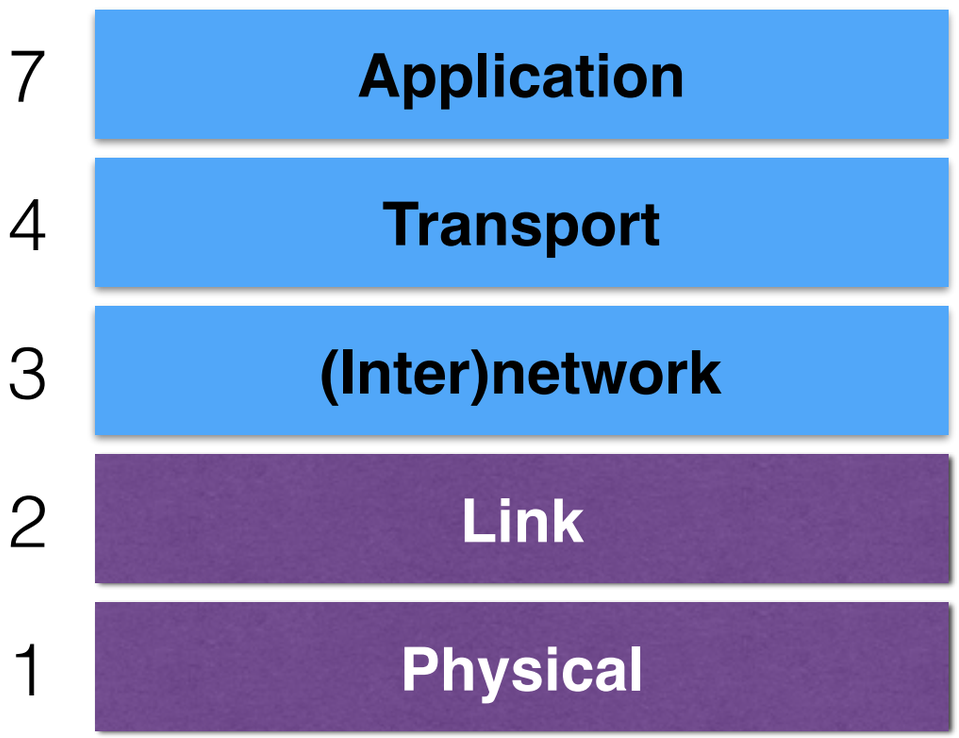
- Routing
  - selecting paths from sources to destinations
  - setting next-hops at each router (based on destination address only)
  - challenge: handling large network size
  - protocols: BGP, Link-state, …
- Forwarding
  - transferring incoming packets to outgoing links
  - challenge: handling high-speed links (pkts/sec)
  - protocol: IPv4, IPv6

Layer 4: Transport layer



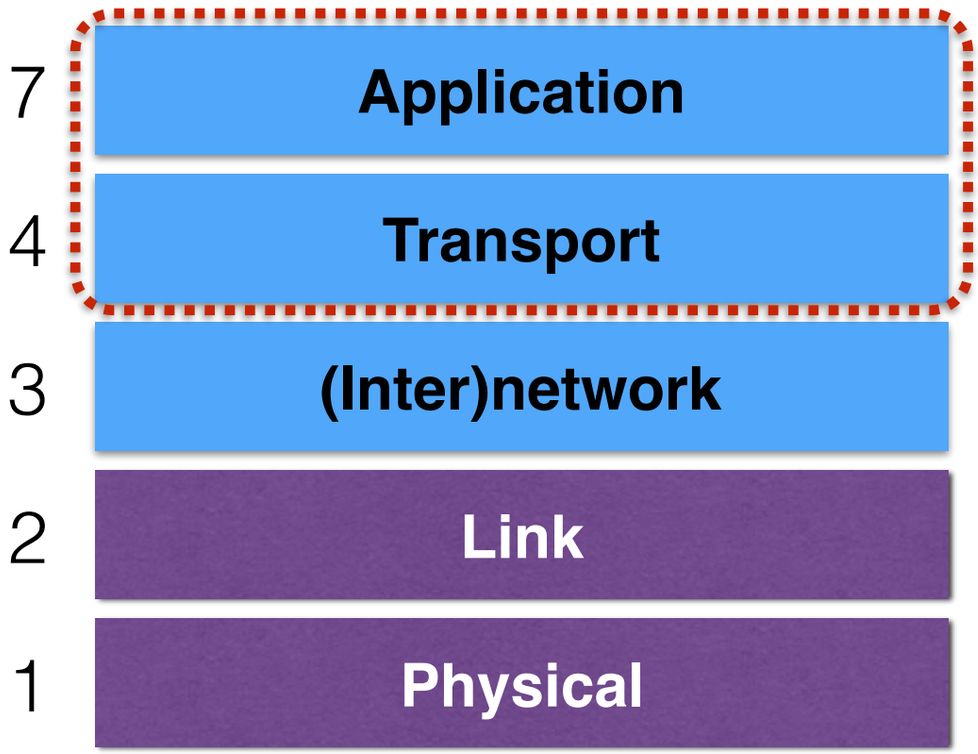
- End-to-end communication between **processes**
- Different types of services provided:
  - UDP: unreliable *datagrams*
  - TCP: *reliable* byte stream
- "Reliable" = keeps track of what data were received properly and retransmits as necessary

Layer 7: Application layer



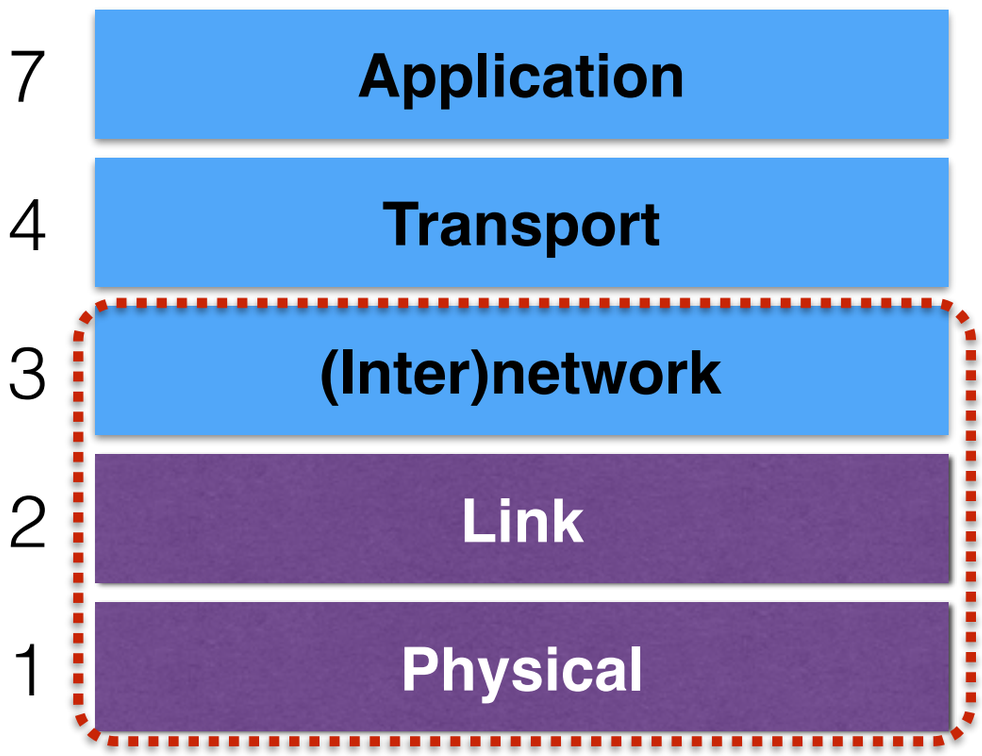
- Communication of whatever you want
- Can use whatever transport(s) is(are) convenient/appropriate
- Freely structured
- Examples:
  - Skype (UDP)
  - SMTP = email (TCP)
  - HTTP = web (TCP)
  - Online games (TCP and/or UDP)

Internet layering = “Protocol stack”



Implemented only at end hosts, not at interior routers (this is our “dumb network”)

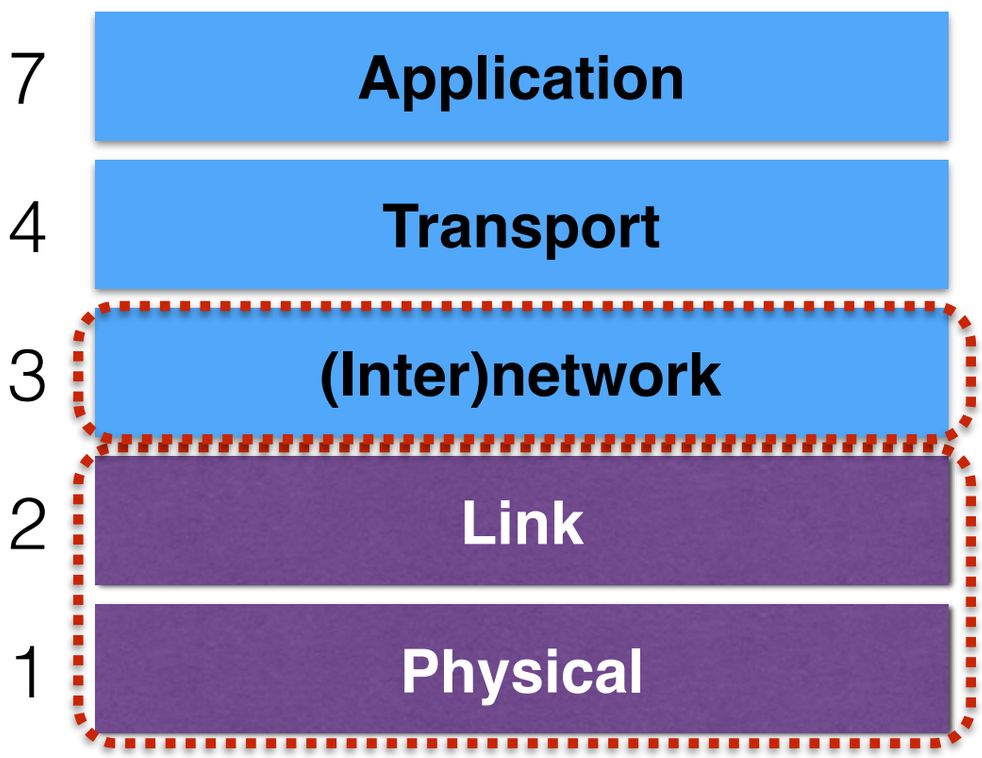
Internet layering = “Protocol stack”



Implemented *everywhere*

The network is “dumb” but it needs to know precisely this much to do its job.

Internet layering = “Protocol stack”

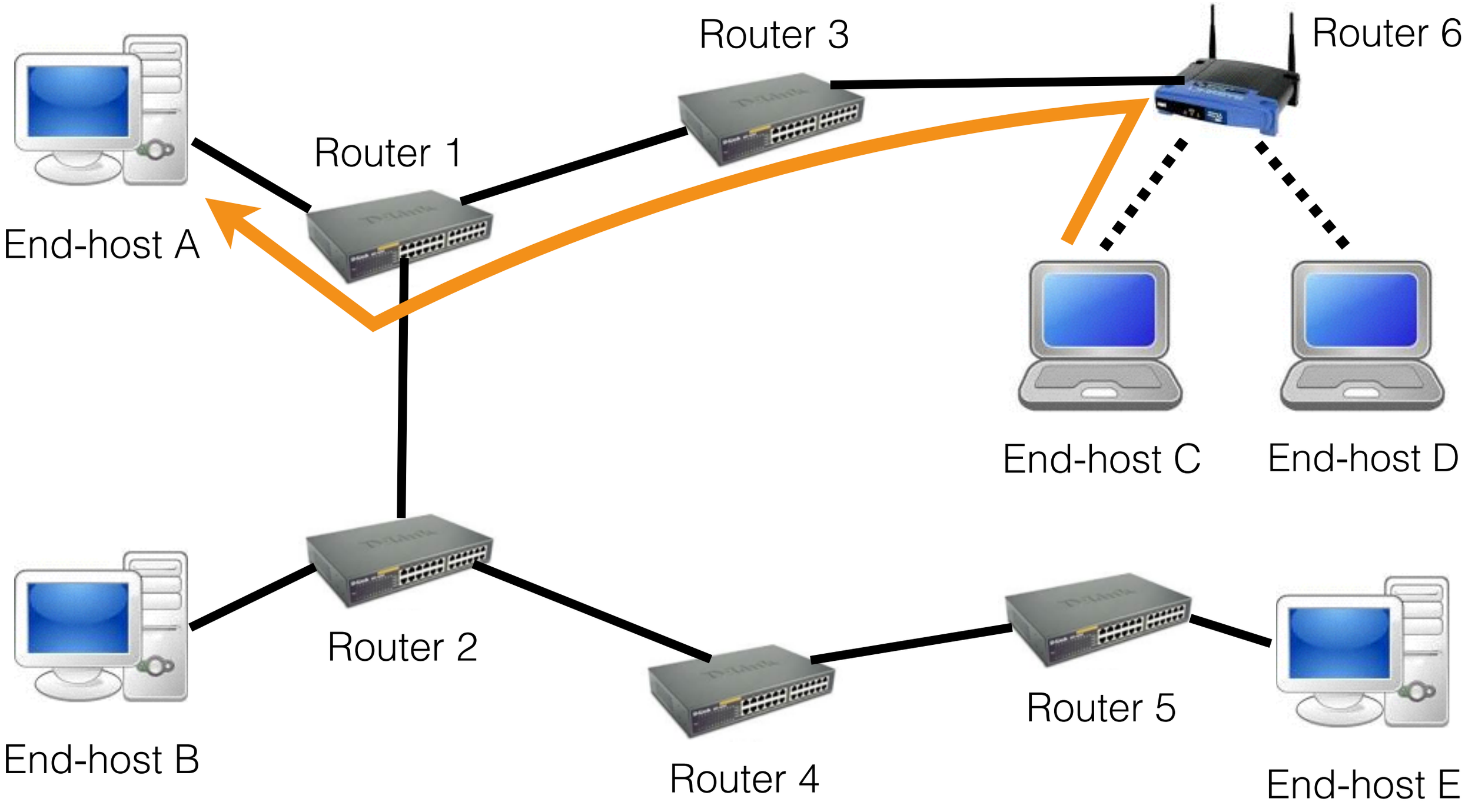


~Same for each Internet “hop”

Can be different for each Internet “hop”

Hop-by-hop vs. end-to-end layers

Host C communicates with host A



Hop-by-hop vs. end-to-end layers