Computer Networks

Protocol Architecture and TCP/IP

Amitangshu Pal
Computer Science and Engineering
IIT Kanpur

Protocol “layers” and reference models

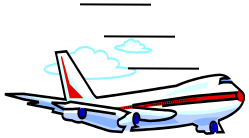
Networks are complex, with many “pieces”:

- hosts
- routers
- links of various media
- applications
- protocols
- hardware, software

Question: is there any hope of *organizing* structure of network?

Solution: ***Layering***

Example: organization of air travel

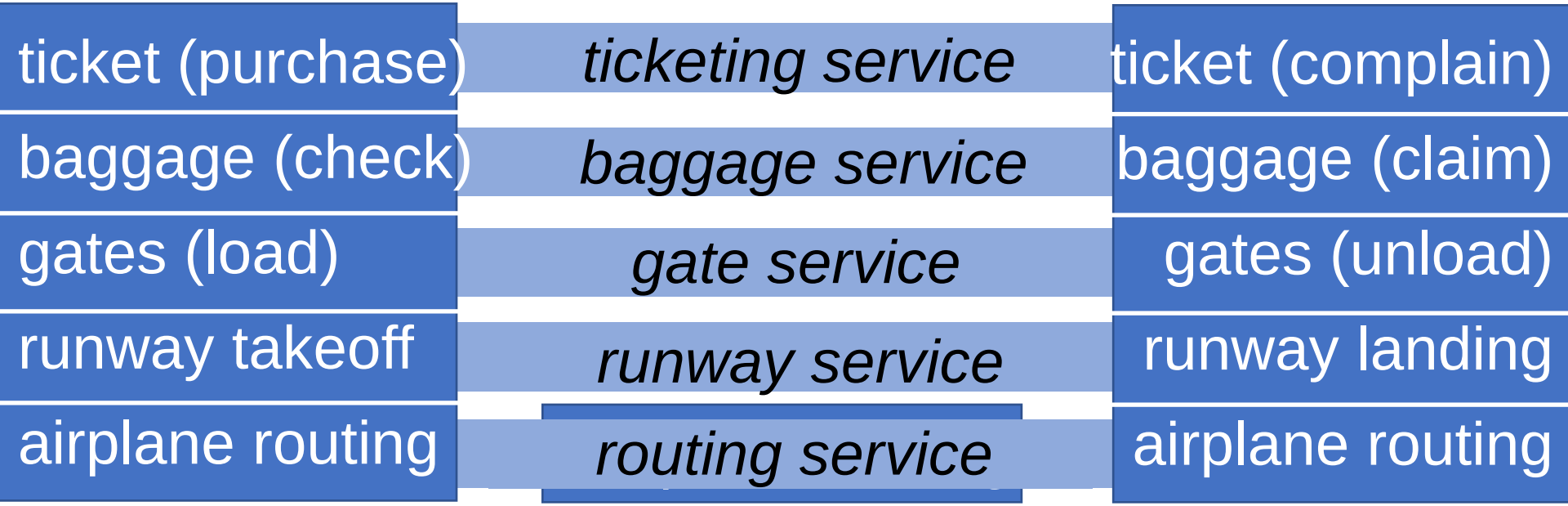


end-to-end transfer of person plus baggage

ticket (purchase)	ticket (complain)
baggage (check)	baggage (claim)
gates (load)	gates (unload)
runway takeoff	runway landing
airplane routing	airplane routing

airplane routing

Example: organization of air travel



layers: each layer implements a service

Why layering?

Approach to designing/discussing complex systems:

- explicit structure allows identification, relationship of system's pieces
- modularization eases maintenance, updating of system
  - change in layer's service *implementation*: transparent to rest of system
  - e.g., change in gate procedure doesn't affect rest of system

Protocol Models

- OSI model: Open System Interconnection
  - Developed by International Standard Organization (ISO)
  - 7 layers

- TCP/IP model:
  - Developed by DARPA for first generation packet switched networks (ARPANET)
  - 5 Layers
  - Used by global Internet

TCP/IP Layers

User Processes


Operating Systems



Network Interface Cards


application

transport

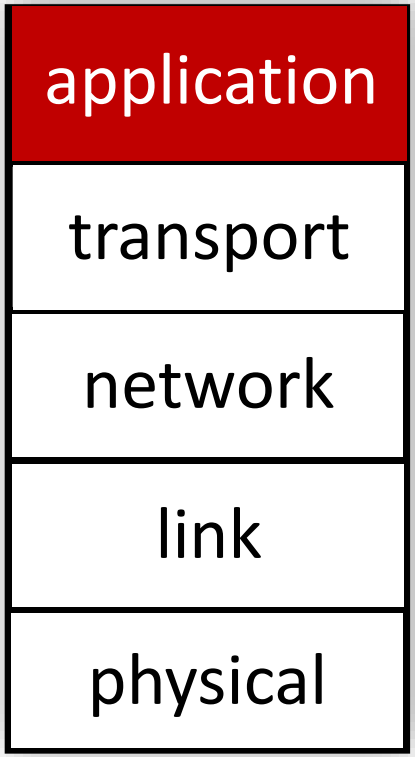
network

link

physical

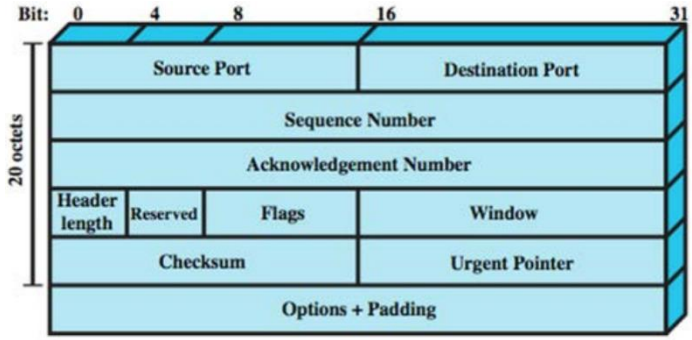
Application Layer

- Application layer:
  - Support user application
    - Web browsing: HTTP
    - File Transfer: FTP
    - Email: SMTP

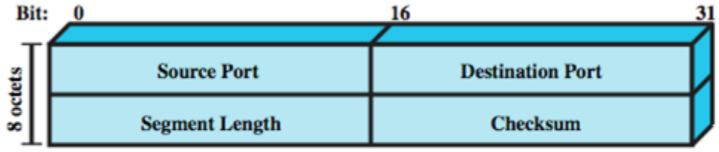


Transport Layer

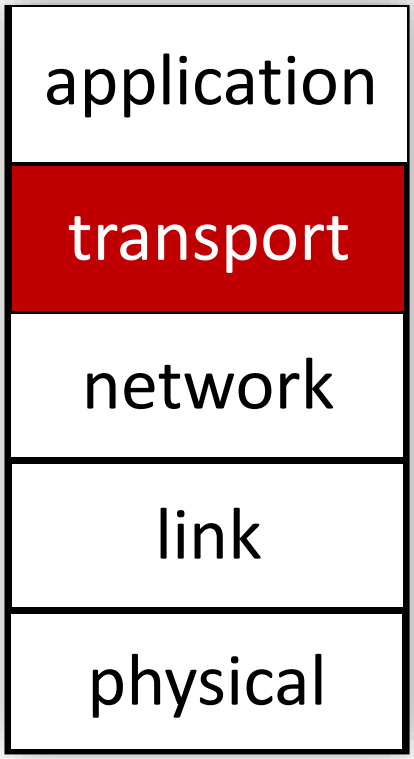
- Transport layer:
  - Provide process-to-process delivery
  - TCP: Connection-oriented, reliable service (No error, In order)
  - UDP: Connectionless, unreliable service



(a) TCP Header

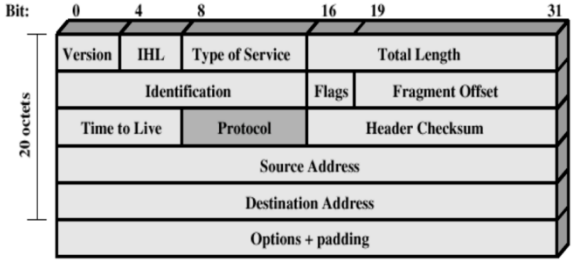


(b) UDP Header

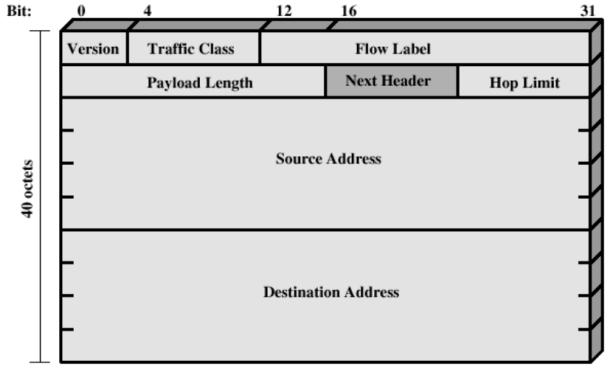


Network Layer

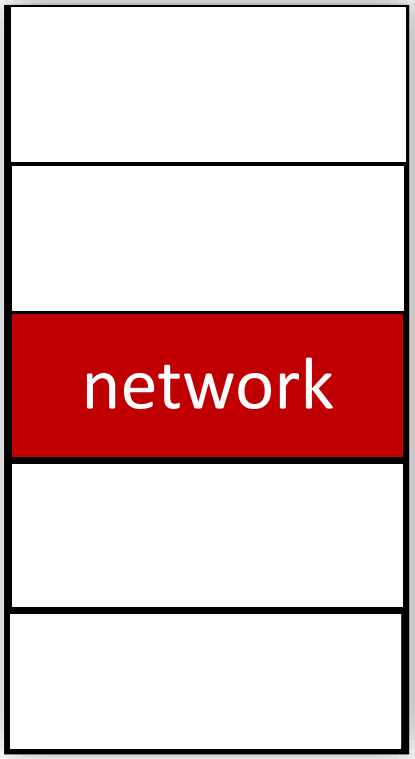
- Network layer:
  - Provide Routing function across multiple networks
  - Uses Internet Protocol (IP) to provide routing functions
  - May provide QoS, congestion control etc
  - Relevant protocols: ICMP, OSPF, RSVP



- IPv4 header:
  - 32 bit source and destination address



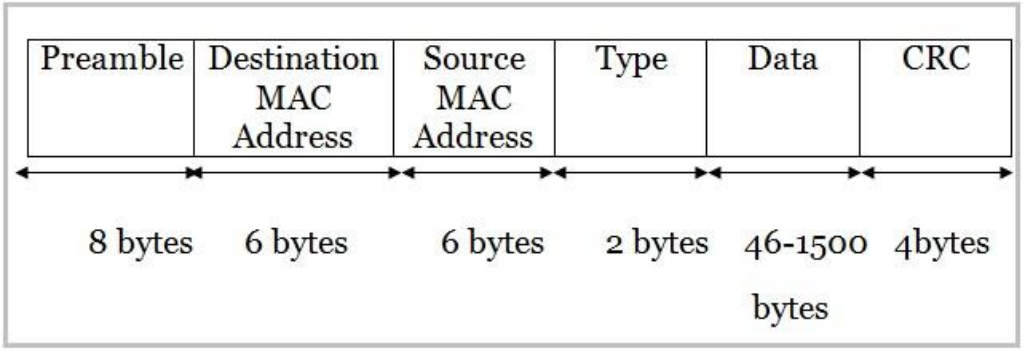
- IPv6 header:
  - 128 bit source and destination address



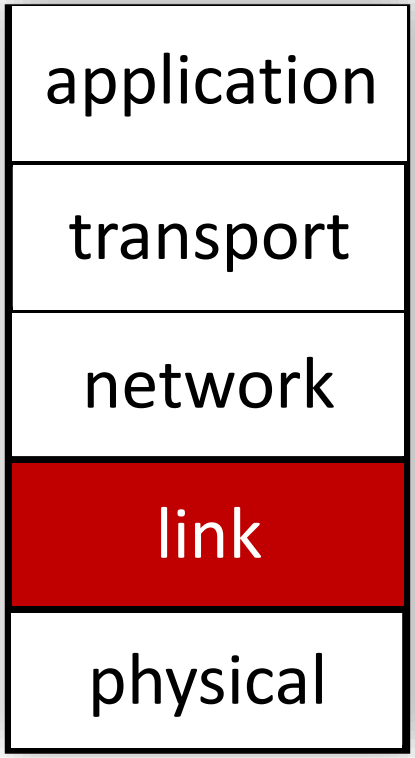
Data Link Layer

❑Data Link layer:

- Transmission of data over the link to which the device is attached
- Provide reliable delivery over a link
- Flow control and error control
- Sometimes called:
  - Network Access Layer
  - MAC Layer
  - Link Layer
  - Hardware Layer
- Relevant protocols: Ethernet, WiFi

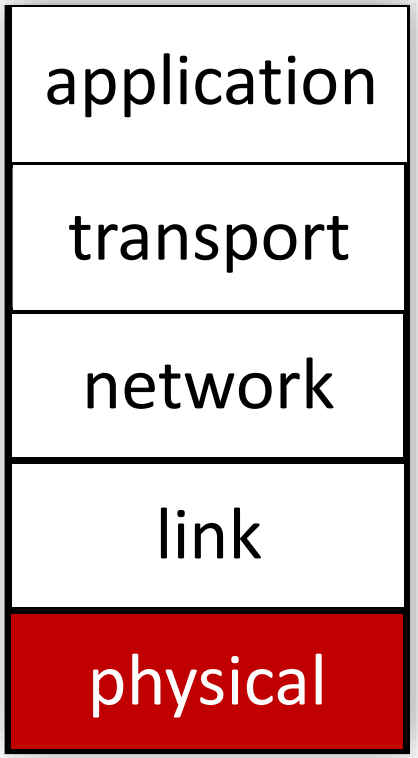


Src: https://www.minitool.com/lib/ethernet-frame.html



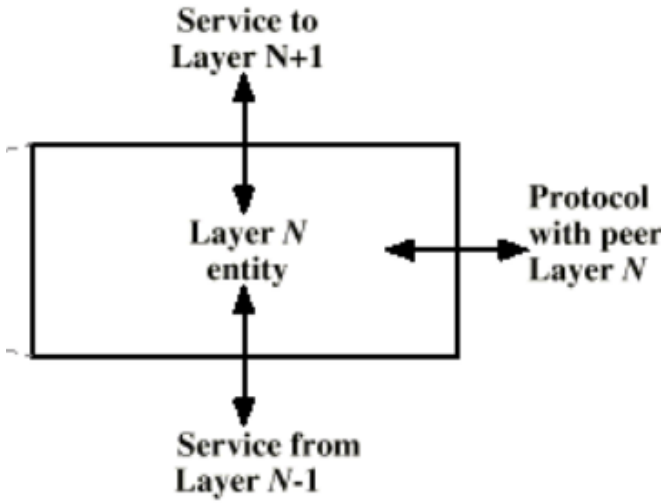
Physical Layer

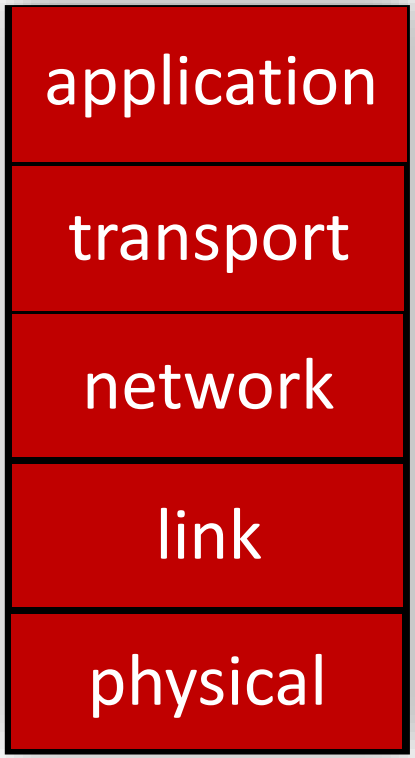
- Physical layer:
  - Transmission of bitstream
  - Apply appropriate encoding, modulation techniques



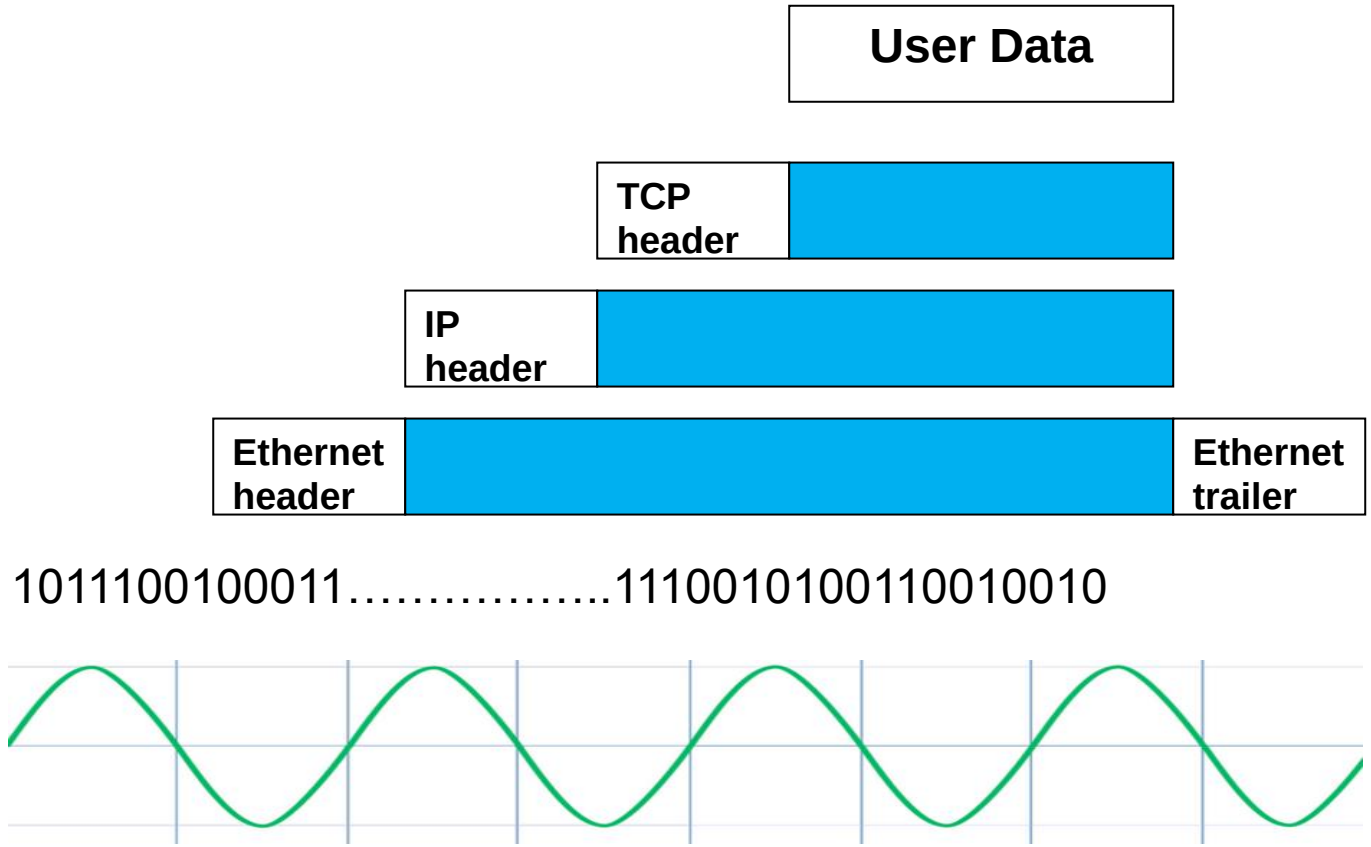
TCP/IP Layers

- Layers are arranged in vertical tasks
  - Layer N uses service of layer N-1
  - Layer N provides service to layer N+1
- Peer layers communicate with a protocol

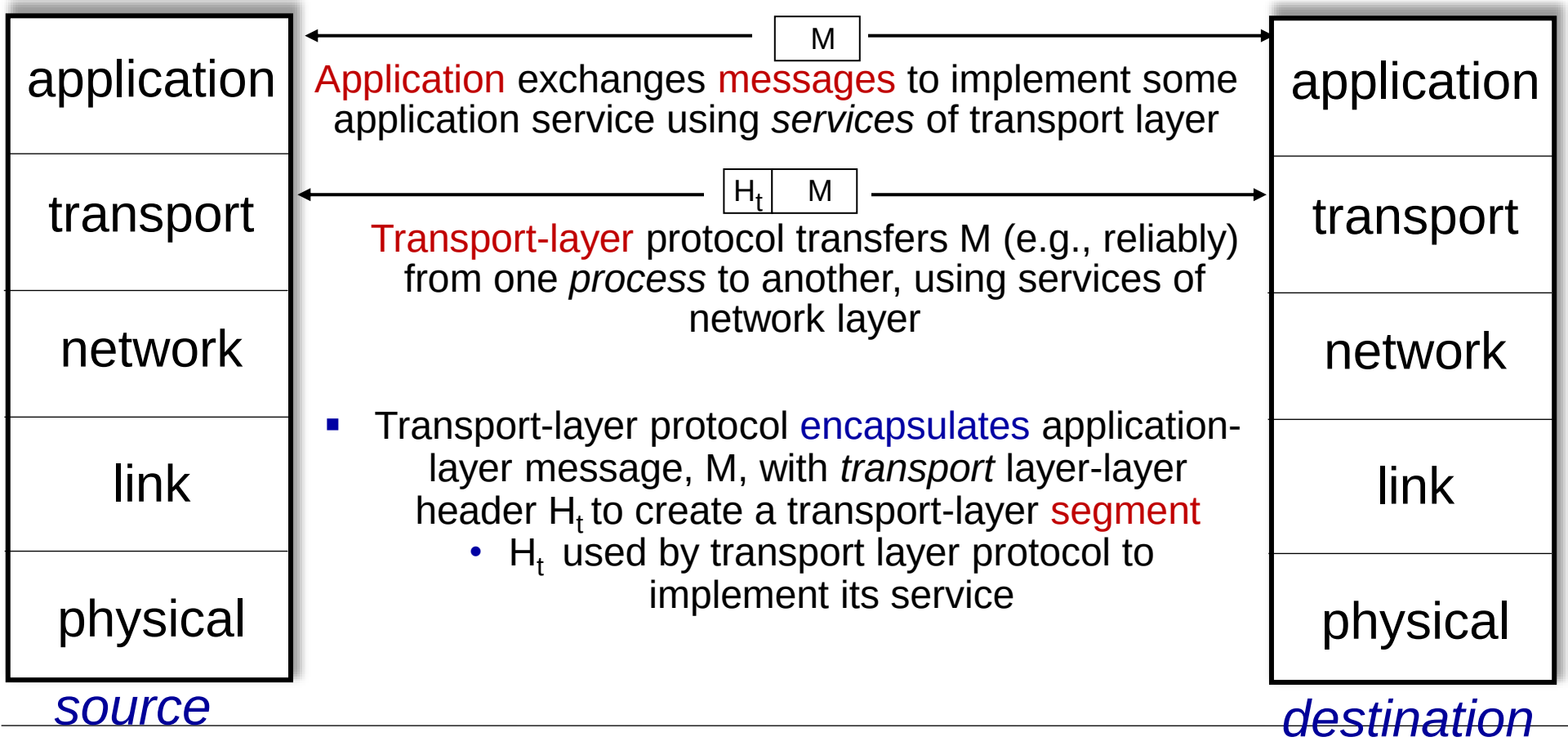




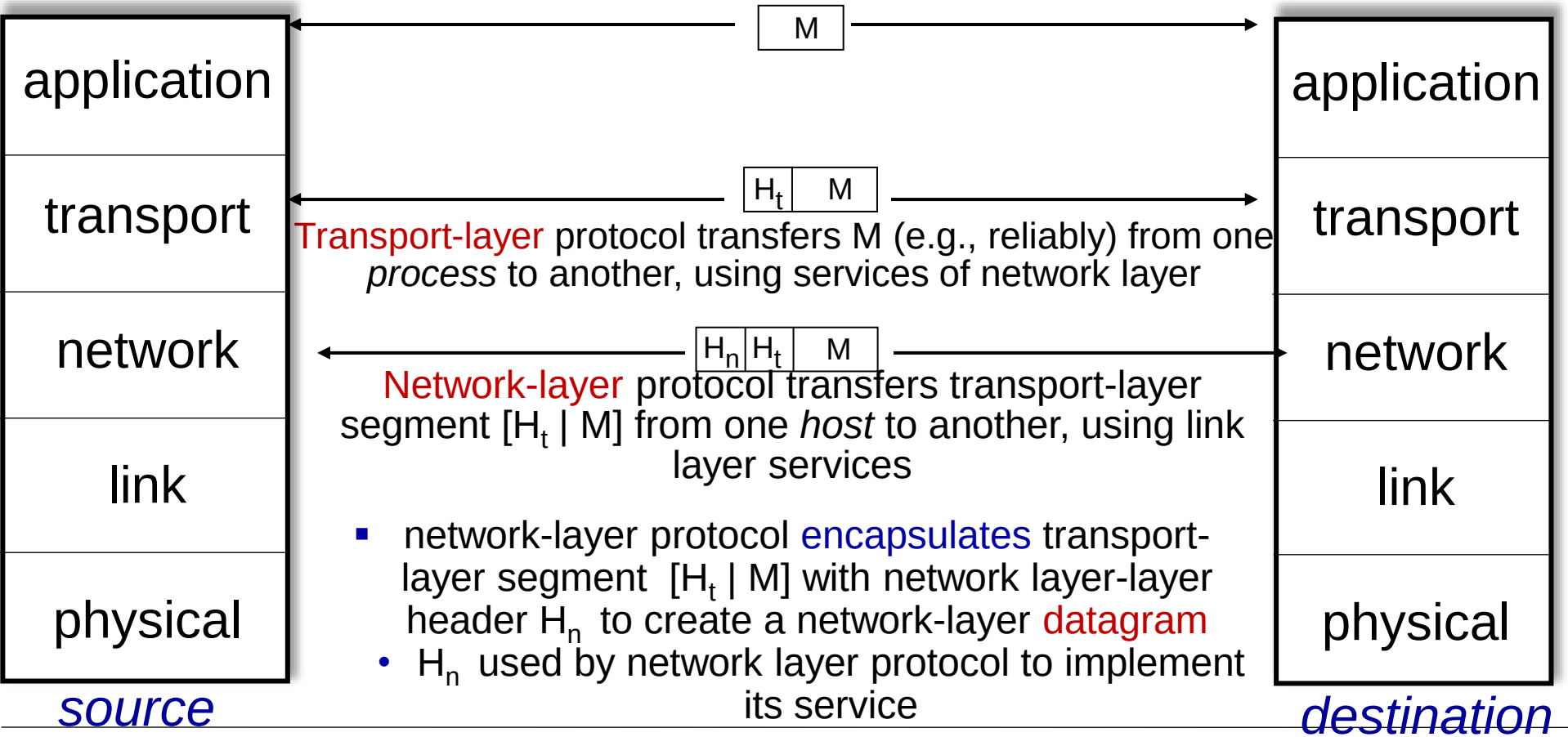
Protocol Data Unit



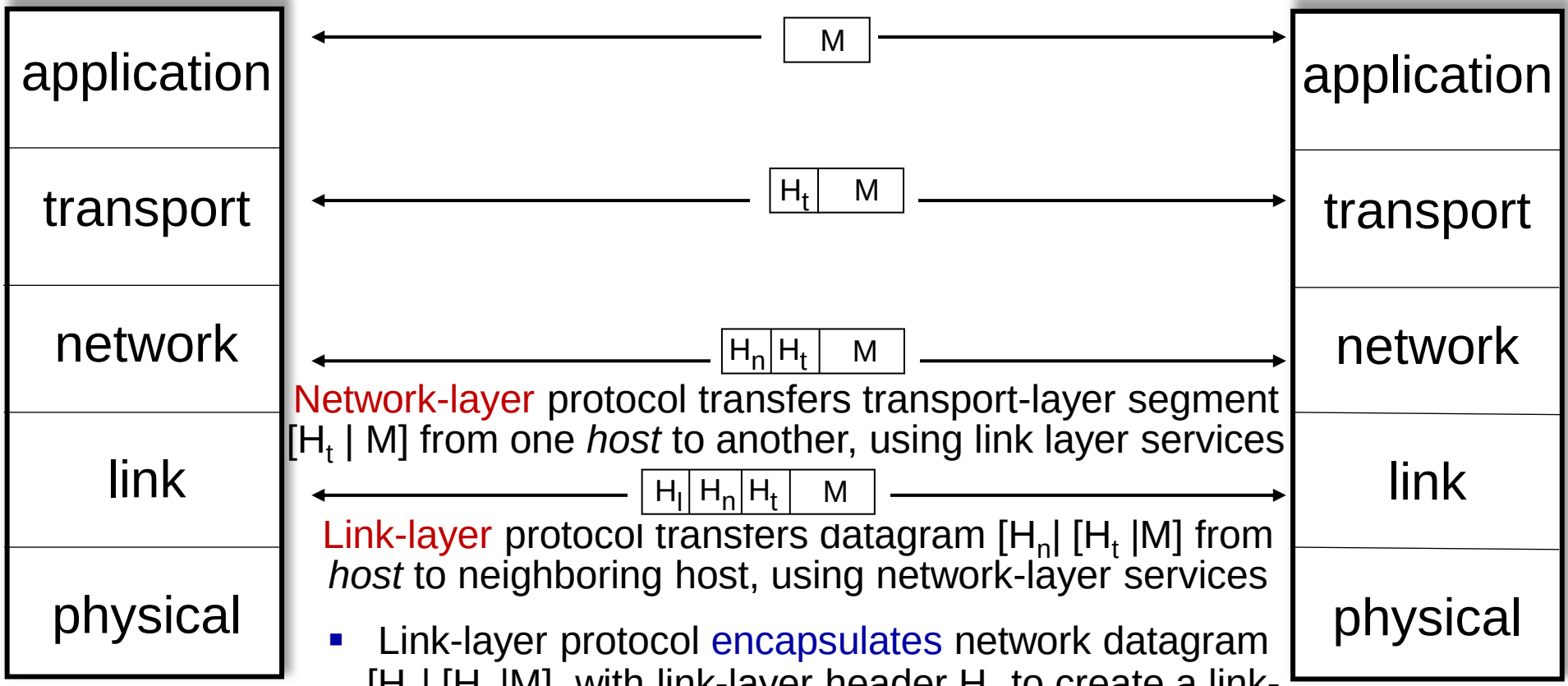
Services, Layering and Encapsulation



Services, Layering and Encapsulation



Services, Layering and Encapsulation



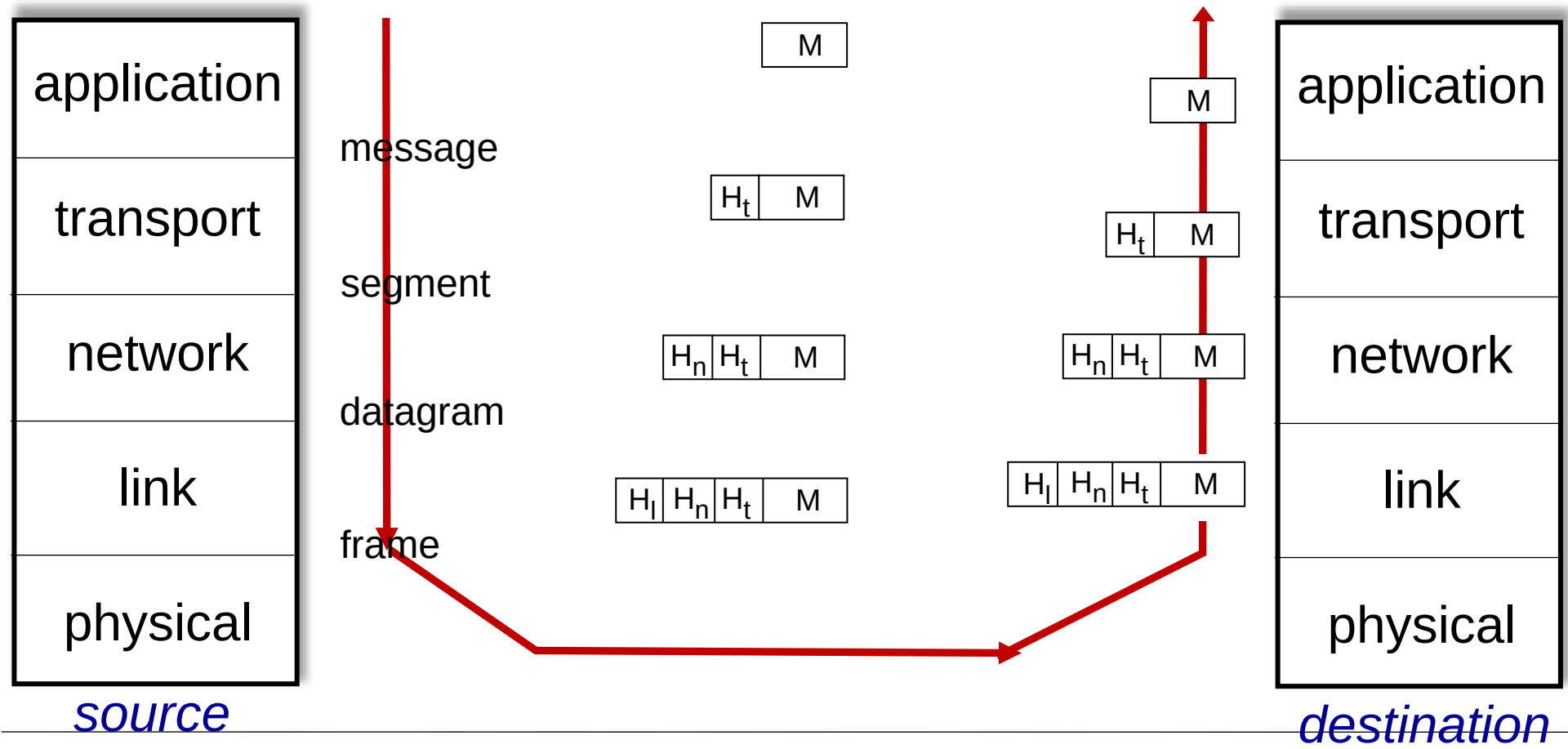
Network-layer protocol transfers transport-layer segment [H_t | M] from one *host* to another, using link layer services

Link-layer protocol transfers datagram [H_n| [H_t |M] from *host* to neighboring host, using network-layer services

- Link-layer protocol encapsulates network datagram [H_n| [H_t |M], with link-layer header H_l to create a link-layer frame

source *destination*

Services, Layering and Encapsulation



Encapsulation: an end-end view