PART X

PROTOCOL LAYERING

Motivation For Layering

- Communication is difficult to understand
- Many subproblems
  - Hardware failure
  - Network congestion
  - Packet delay or loss
  - Data corruption
  - Data duplication or inverted arrivals

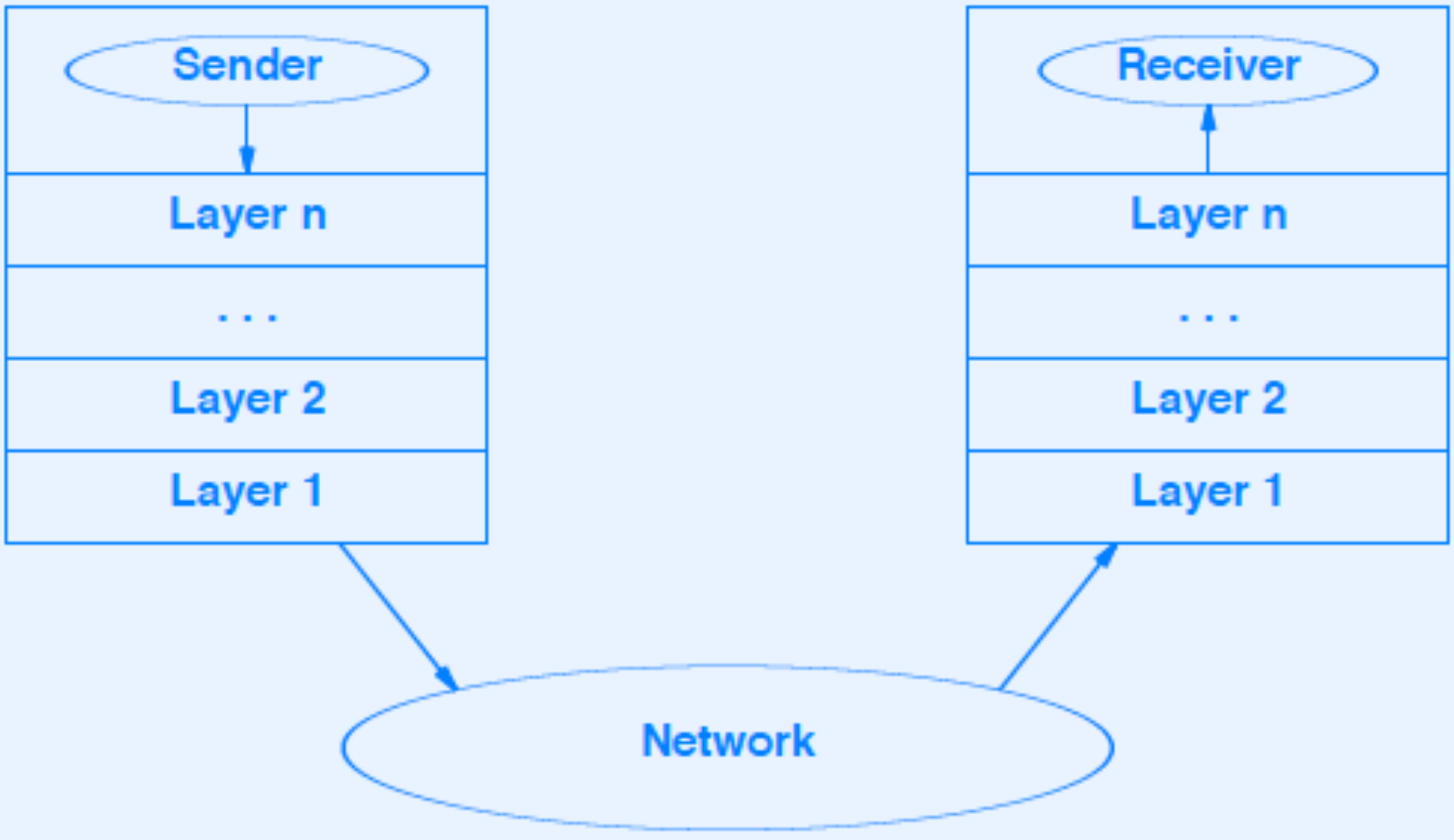
Solving The Problem

- Divide the problem into pieces
- Solve subproblems separately
- Combine into integrated whole
- Result is *layered protocols*

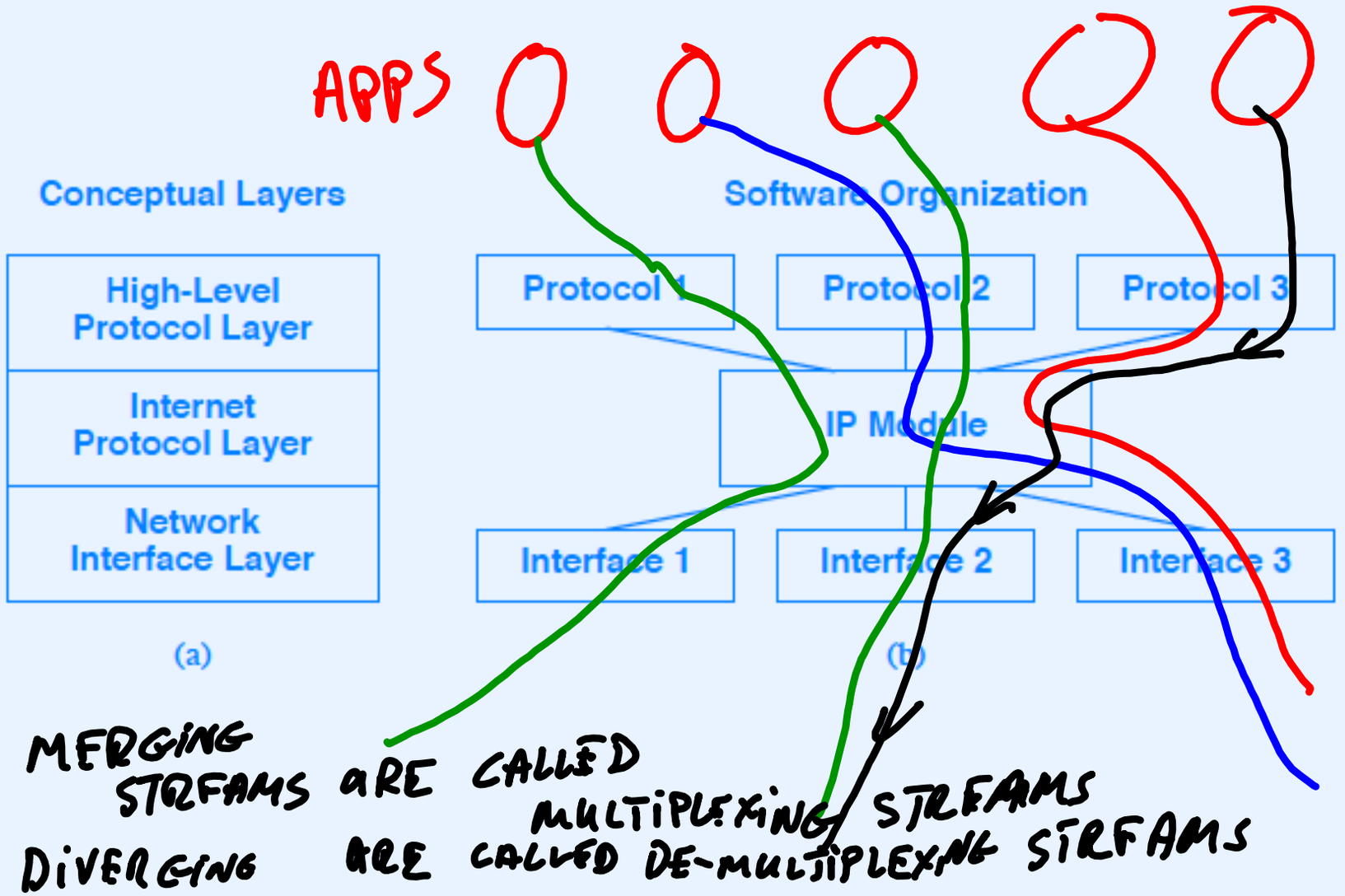
Protocol Layering

- Separates protocol functionality
- Each layer solves one part of the communication problem
- Intended primarily for protocol designers
- Set of layers is called a *protocol stack*

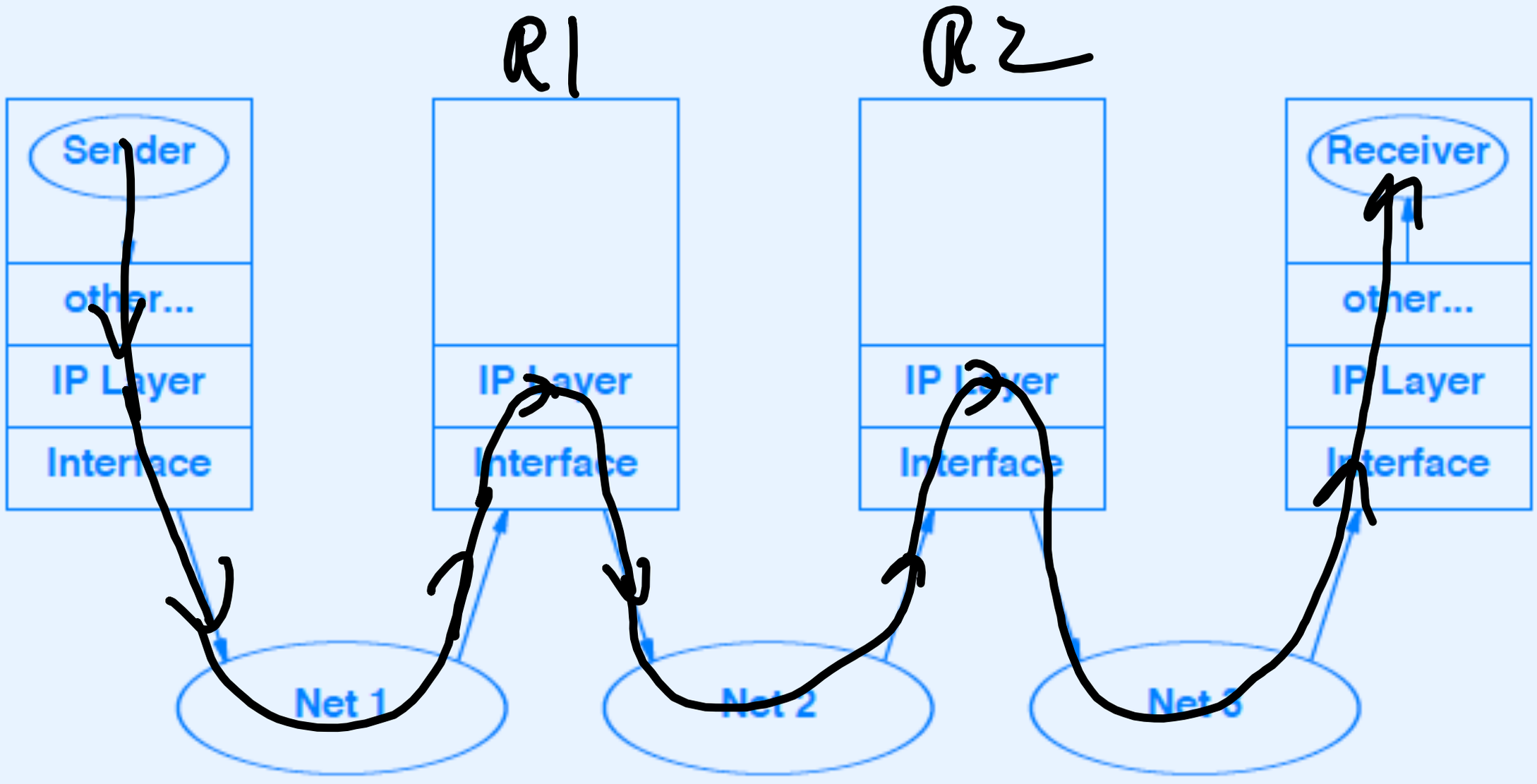
Concept Of Layering



More Realistic Layering



Layering In An Internet



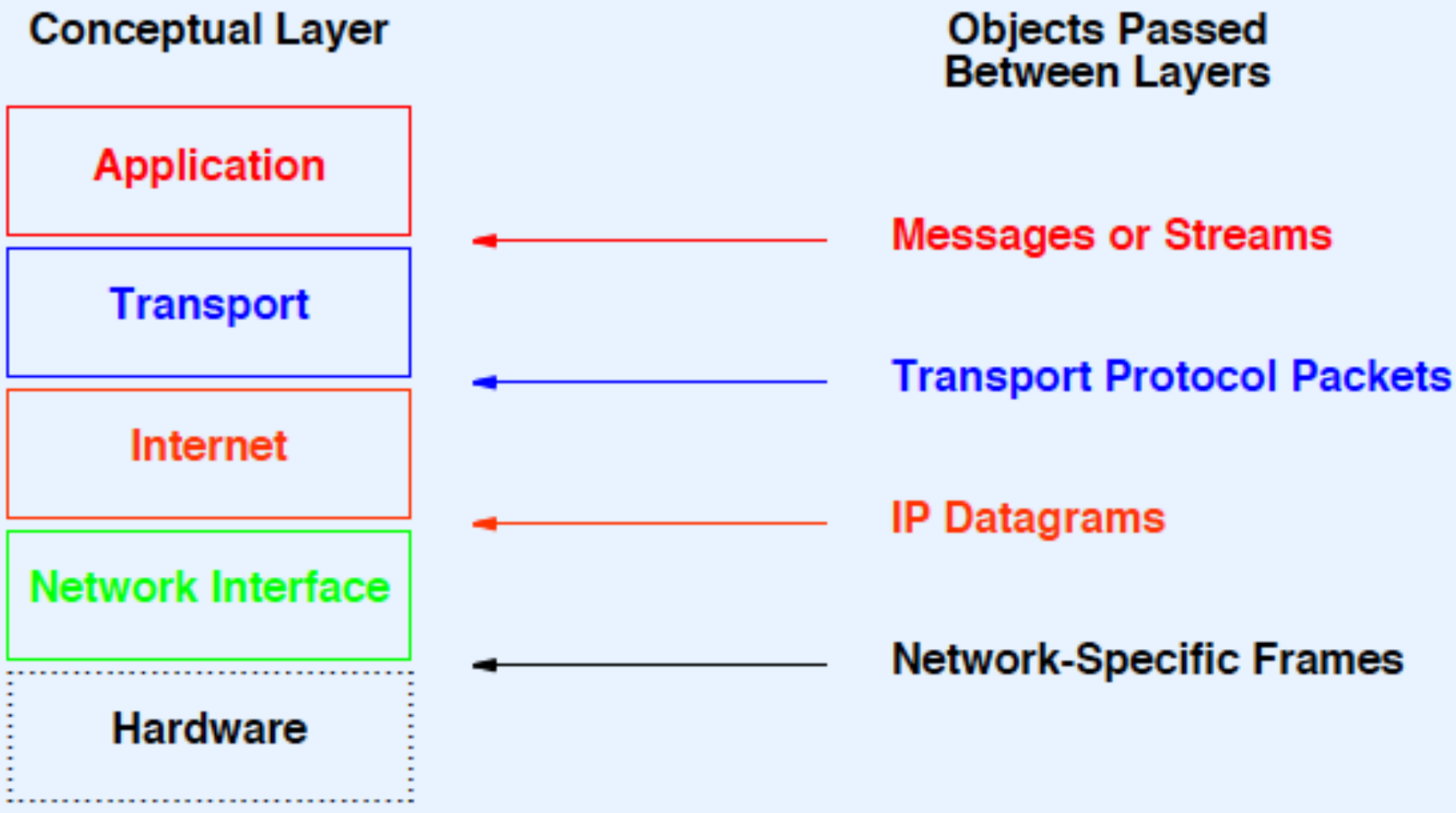
Examples Of Layering

- Two models exist
- ISO 7-layer reference model for *Open System Interconnection* (*OSI*)
  - Predates TCP/IP
  - Does not include an Internet layer
  - Prescriptive (designed before protocols)
- Internet 5-layer reference model
  - Designed for TCP/IP
  - Descriptive (designed along with actual protocols)

ISO 7-Layer Reference Model

Layer	Functionality
7	Application
6	Presentation
5	Session
4	Transport
3	Network
2	Data Link (Hardware Interface)
1	Physical Hardware Connection

TCP/IP 5-Layer Reference Model



- Only four layers above hardware

TCP/IP Layer 1: Physical Hardware

- Defines electrical signals used in communication (e.g., voltages on wires between two computers)
- Uninteresting except to electrical engineers

TCP/IP Layer 2: Network Interface

- Defines communication between computer and network hardware
- Isolates details of hardware (MAC) addressing
- Example protocol: ARP
- Code is usually in the operating system

TCP/IP Layer 3: Internet

- Protocol is IP
- Provides machine to machine communication
- Defines best-effort, connectionless datagram delivery service for the Internet
- Code is usually in the operating system

TCP/IP Layer 4: Transport

- Provides end-to-end connection from application program to application program
- Often handles reliability, flow control
- Protocols are TCP and UDP
- Code is usually in the operating system

TCP/IP Layer 5: Application

- Implemented by application programs
- Many application-specific protocols in the Internet
- Built on top of transport layer

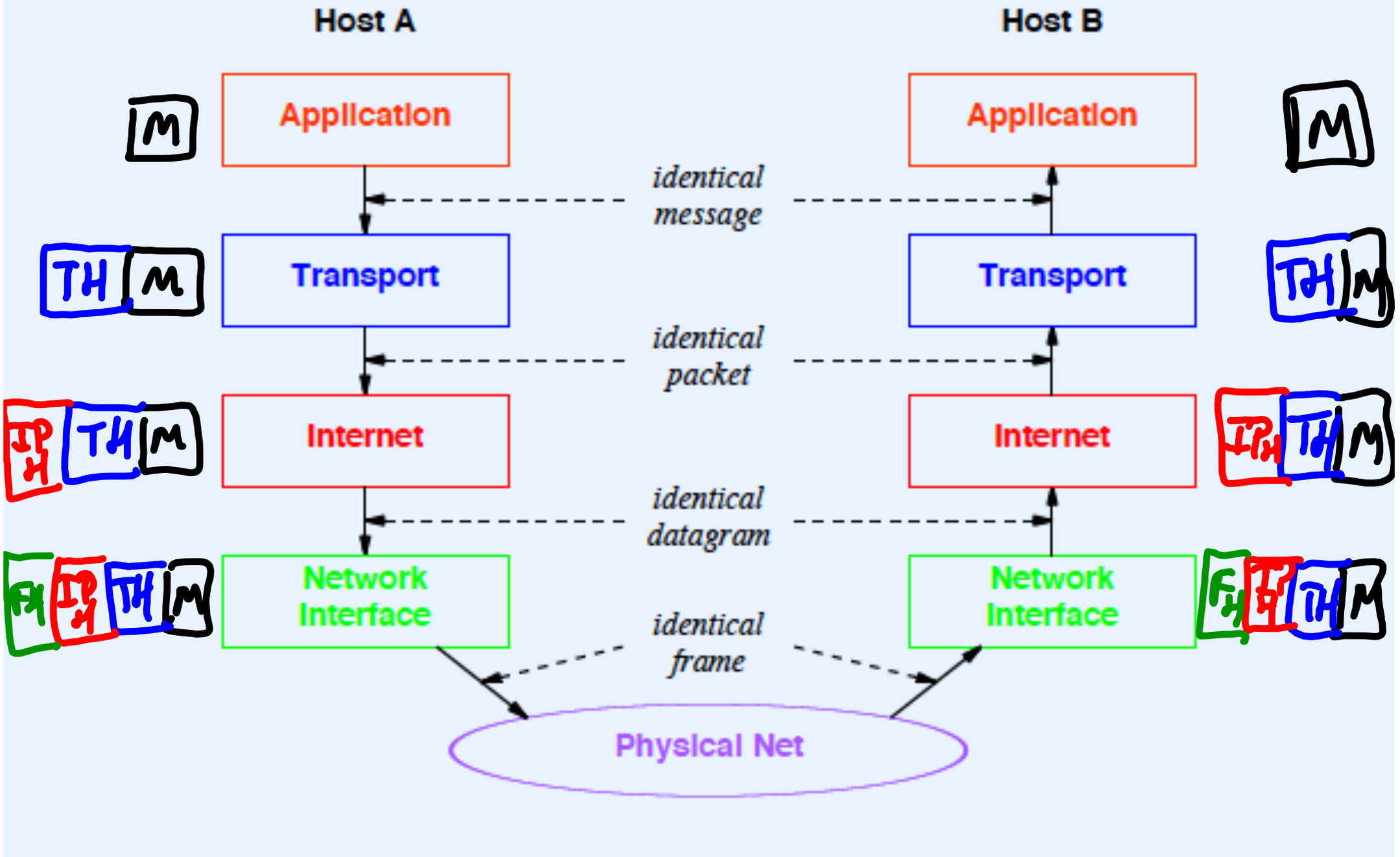
Two Differences Between TCP/IP And Other Layered Protocols

- TCP/IP uses end-to-end reliability instead of link-level reliability
- TCP/IP places the locus of intelligence and decision making at the edge of the network instead of the core

The Layering Principle

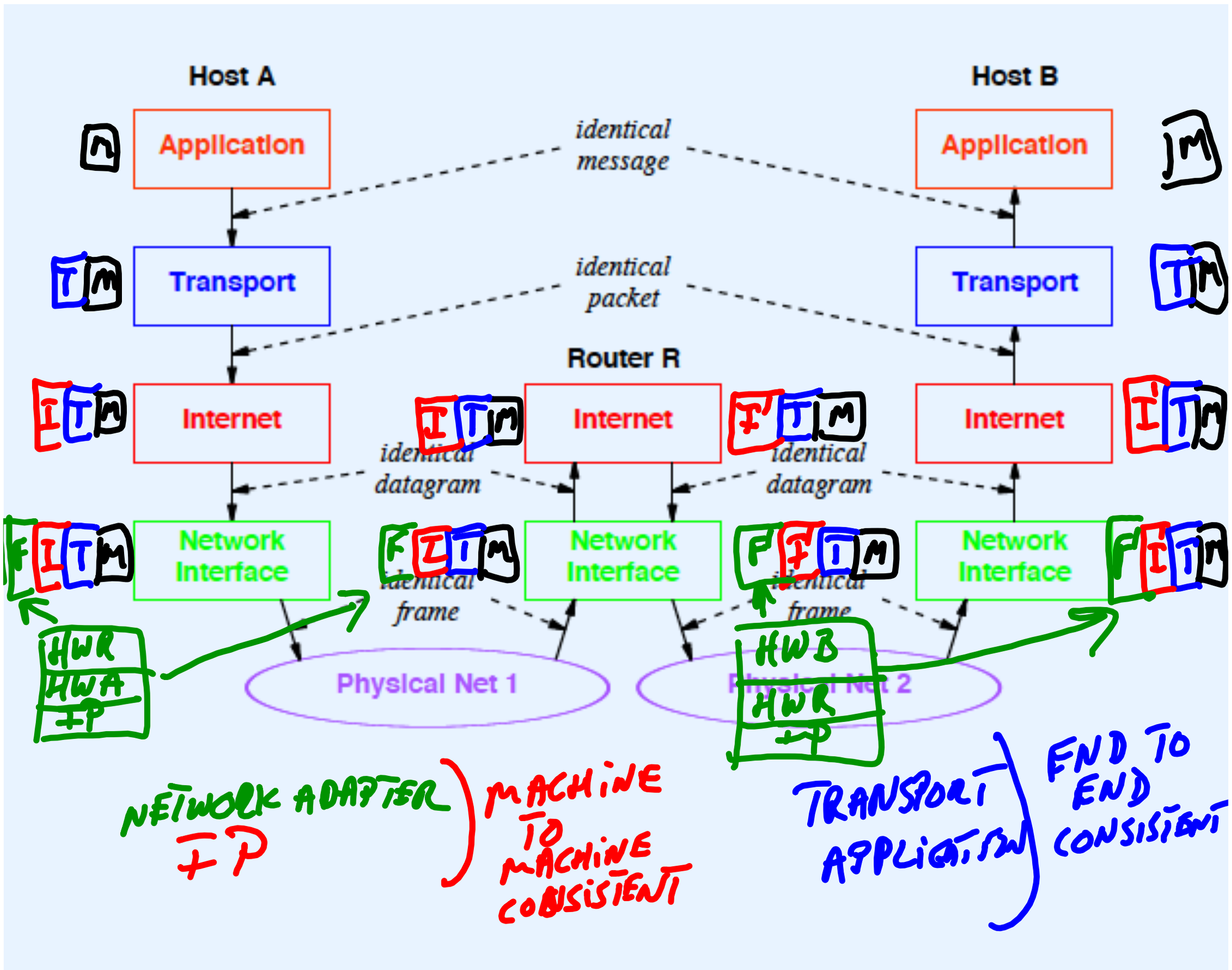
Software implementing layer n at the destination receives exactly the message sent by software implementing layer n at the source.

Illustration Of Layering Principle



When A Datagram Traverses The Internet

- All layers involved at
  - Original source
  - Ultimate destination
- Only up through IP layer involved at
  - Intermediate routers


Host A
Host B
Application
Application
identical message
Transport
Transport
identical packet
Router R
Internet
Internet
Internet
identical datagram
identical datagram
Network Interface
Network Interface
Network Interface
identical frame
identical frame
HWR
HWA
IP
HWB
HWR
IP
Physical Net 1
Physical Net 2
NETWORK ADAPTER
IP
MACHINE TO MACHINE CONSISTENT
TRANSPORT
APPLICATION
END TO END CONSISTENT

A Key Definition

- A protocol is classified as *end-to-end* if the layering principle applies from one end of the Internet to the other
- Examples
  - IP is *machine-to-machine* because layering principle only applies across one hop
  - TCP is *end-to-end* because layering principle from original source to ultimate destination

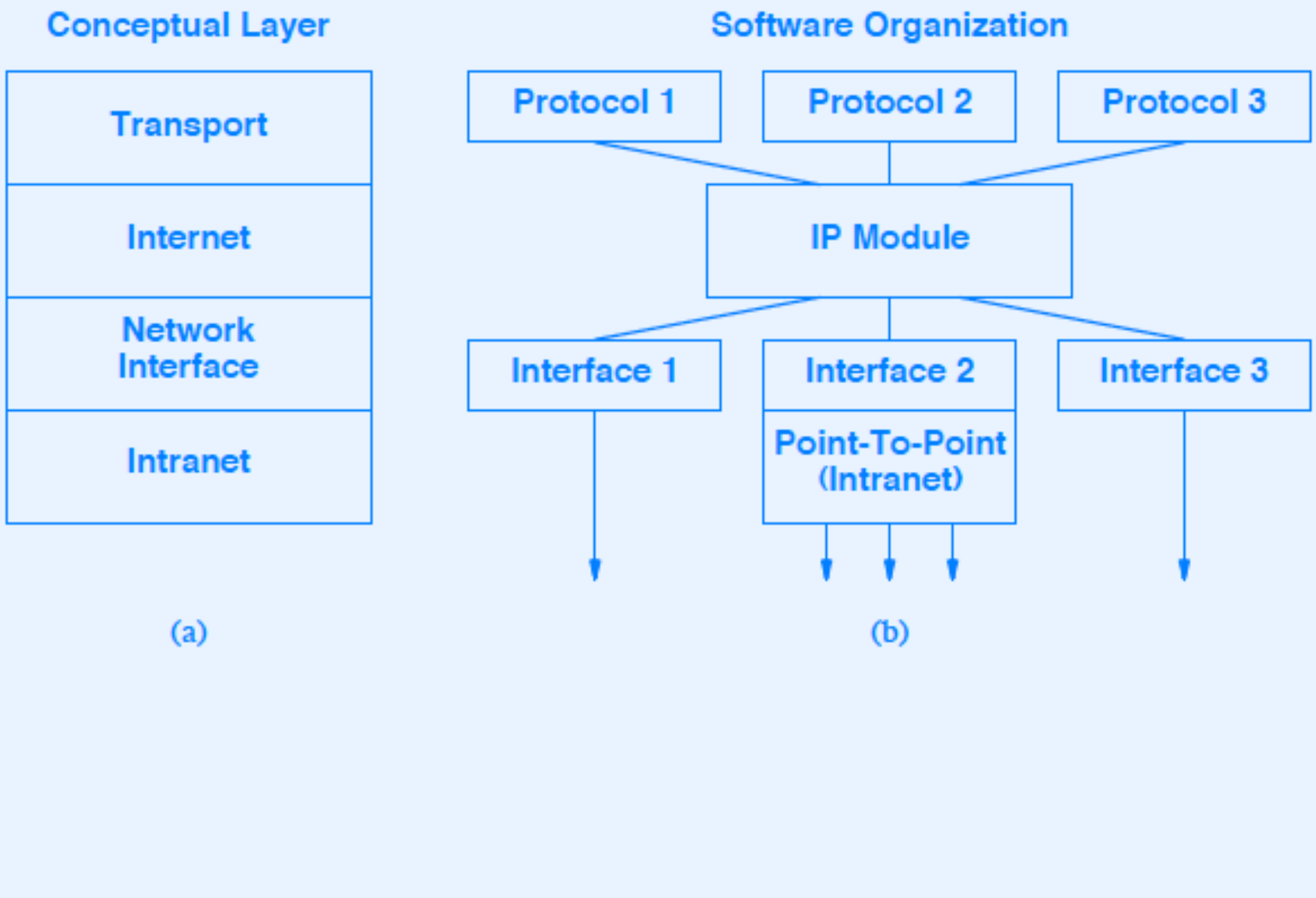
Practical Aspect Of Layering

- Multiple protocols at each layer
- One protocol used at each layer for given datagram

Example Of Two Protocols At Network Interface Layer: SLIP And PPP

- Both used to send IP across
  - Serial data circuit
  - Dialup connection
- Each defines standards for
  - Framing (encapsulation)
  - Addressing
- Incompatible

Notion Of Multiple Interfaces And Layering



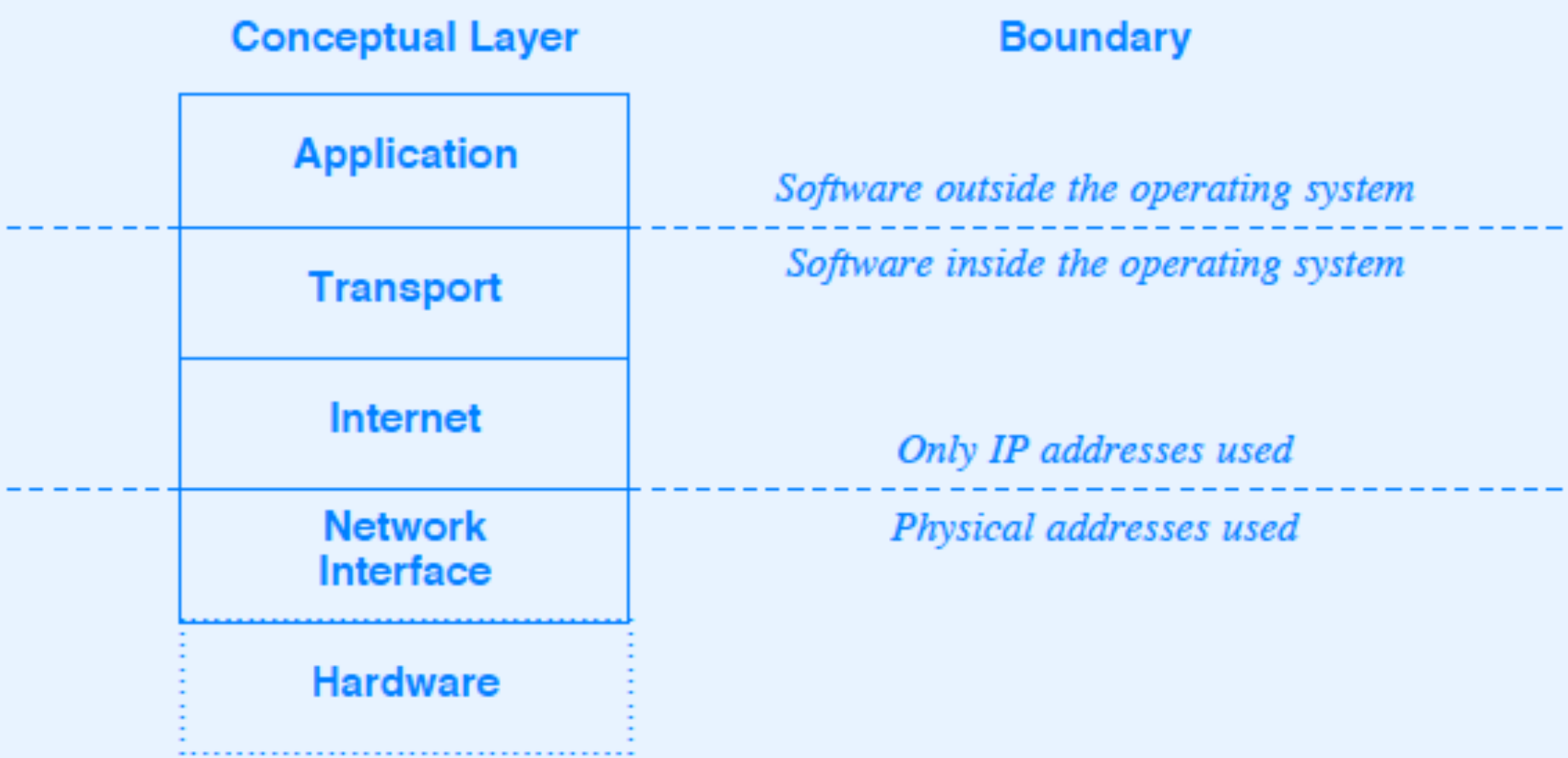
Boundaries In The TCP/IP Layering Model

- High-level protocol address boundary
  - Division between software that uses hardware addresses and software that uses IP addresses
- Operating system boundary
  - Division between application program running outside the operating system and protocol software running inside the operating system

The Consequence Of An Address Boundary

Application programs as well as all protocol software from the Internet layer upward use only IP addresses; the network interface layer handles physical addresses.

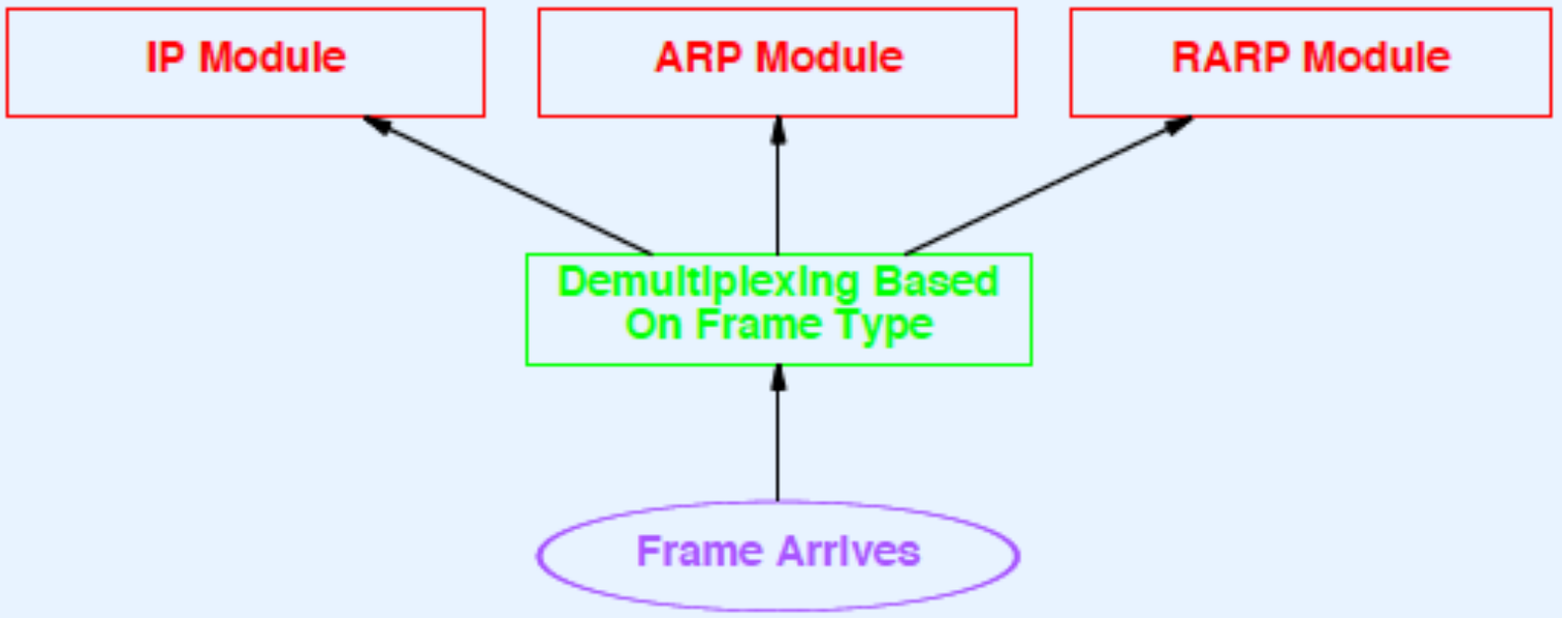
Illustration Of The Two Boundaries



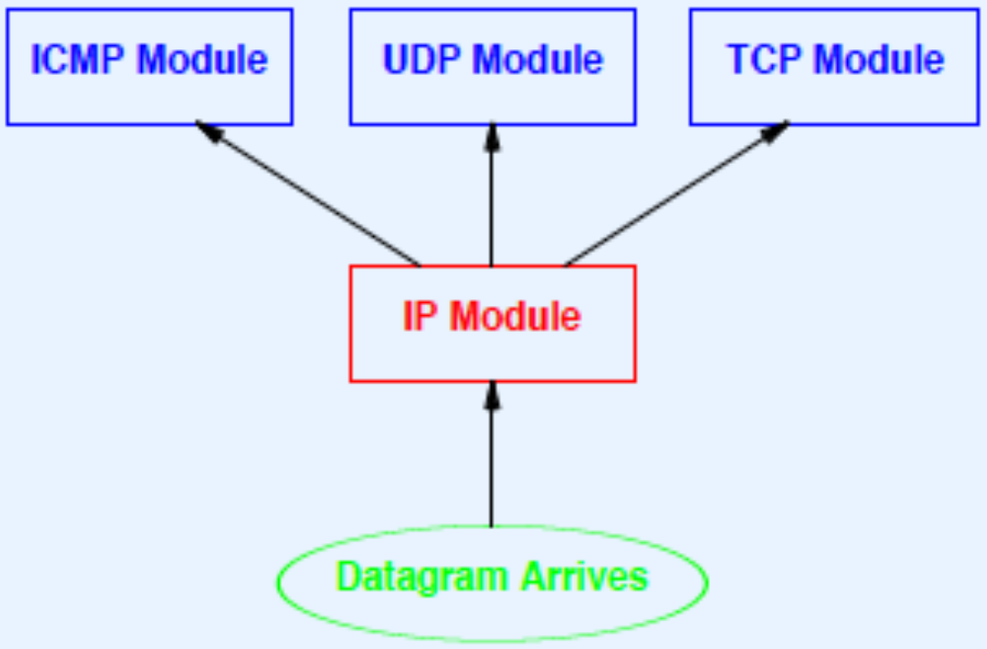
Handling Multiple Protocols Per Layer

- Sender places field in header to say which protocol used at each layer
- Receiver uses field to determine which protocol at next layer receives the packet
- Known as *multiplexing* and *demultiplexing*

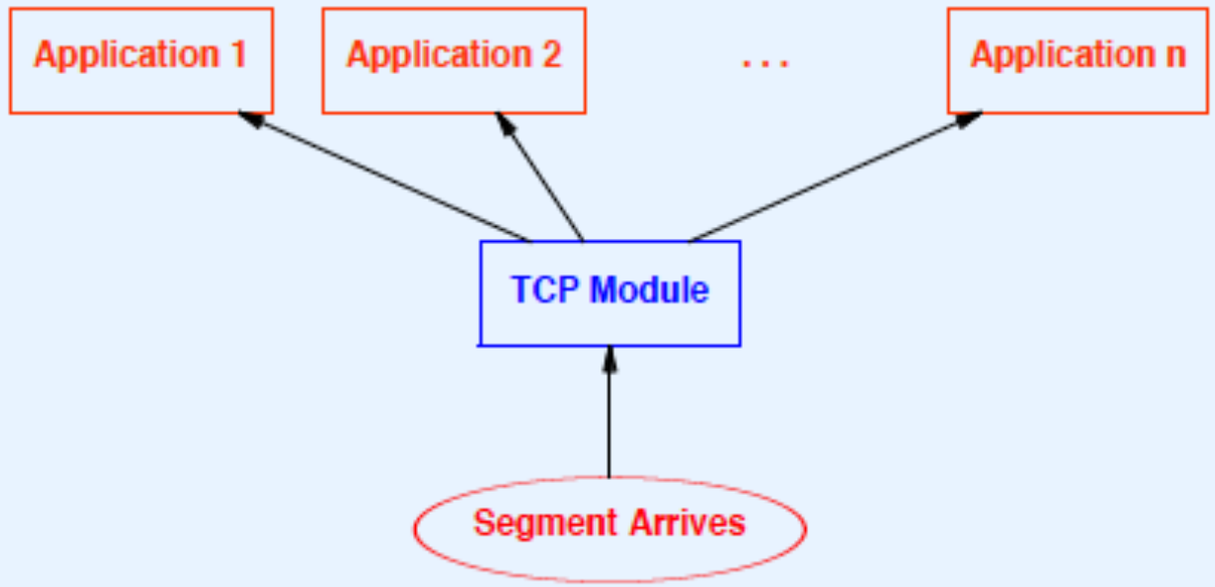
Example Of Demultiplexing An Incoming Frame



Example Of Demultiplexing Performed By IP



Example Of Demultiplexing Performed By TCP



- TCP is part of operating system
- Transfer to application program must cross operating system boundary

Discussion

- What are the key advantages and disadvantages of multiplexing / demultiplexing?
- Can you think of an alternative?

Summary

- Layering
  - Intended for designers
  - Helps control complexity in protocol design
- TCP/IP uses 5-layer reference model
- Conceptually, a router only needs layers 2 and 3, and a host needs all layers
- IP is machine-to-machine protocol
- TCP is end-to-end protocol
- Demultiplexing used to handle multiple protocols at each layer