Network Architecture and the OSI Reference Model

Computer Networks
Term B10

Architecture Outline

- **The Internet and IP**
- **Network Architecture**
  - **Protocols and Layers**
  - **Encapsulation**
- **The OSI Reference Model**
  - **The Seven OSI Layers**
- **The TCP/IP Internet Stack**
- **Layering Example**
- **Tiered Internet Architecture**

The Internet versus an internet

An internet :: involves the interconnection of multiple networks into a single large networks. [LG&W]

The Internet :: refers to the successor to ARPANET. The modern Internet is multi-tiered and includes industrial participation.

IP (the Internet Protocol) :: provides *connectionless* transfer of packets across an internet.

The Internet

- Provides a *name space* to refer to machines connected to the Internet (e.g. chablis.cs.wpi.edu).

- The name space is hierarchical, but it is only administrative and not used in network routing operations.

- DNS (Domain Name Service) provides automatic translation of names to addresses.

IP

- **Currently IP provides best-effort service.**

  – packets may be lost (i.e., IP is unreliable).

- General IP design philosophy

  – Keep internal operations simple by relegating complex functions to the edge of the subnet.

  – IP can operate over any network.

  – This design allows IP to scale!!!

  – The end-to-end mechanisms are responsible for recovery of packet losses and congestion control.

WPI

IPv4

- Uses 32 bit *hierarchical address space* with location information embedded in the structure.

Network ID	Host ID
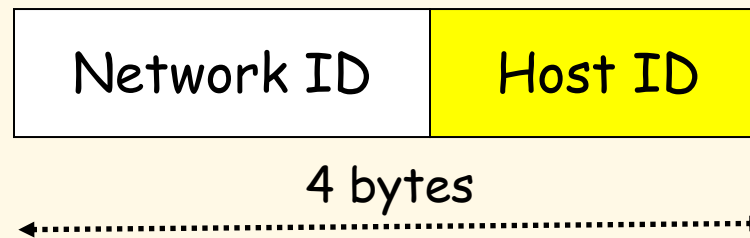
4 bytes
<--->

- IP address is usually expressed in *dotted-decimal notation*

e.g.,

128.100.11.56

IPv6

- IPv6 addresses are 128 bits long.
- 16 bytes of IPv6 address are represented as a group of hexadecimal digits, separated by colons. e.g.

[D&C]

2000:fdb8:0000:0000:0001:00ab:853c:39a1

- Shorthand – leave out groups of zeros and leading zeros.

2000:fdb8::1:ab:853c:39a1

- In the 1970's vendor companies (IBM and DEC) developed proprietary networks with the common feature of grouping communication functions into related and manageable sets called layers.

network architecture :: a set of protocols that specify how every layer is to function and the defined interfaces between the layers. [LG&W]

What's a protocol?

human protocols:

- "what's the time?"
- "I have a question"
- introductions

... specific msgs sent

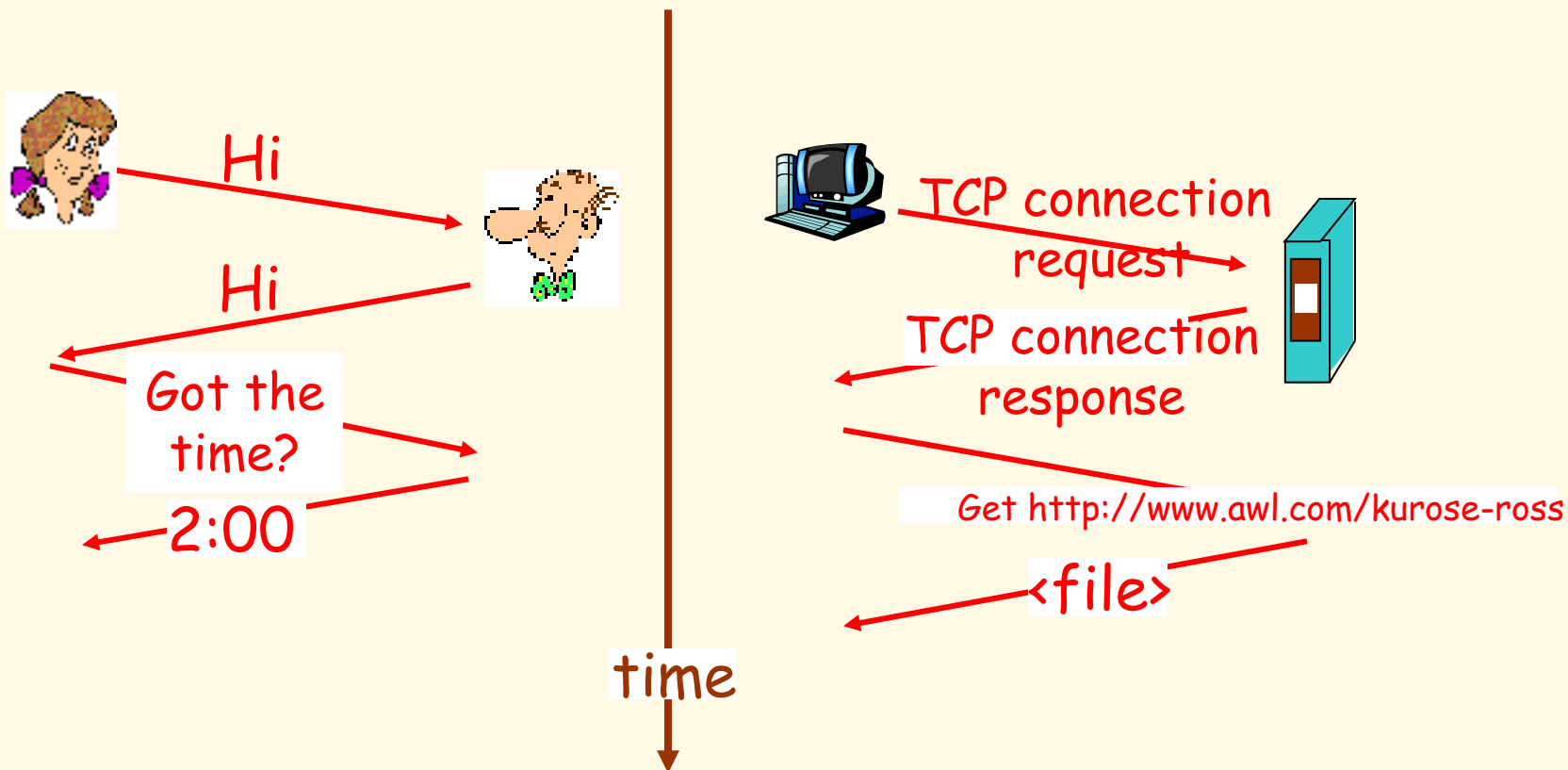... specific actions taken when msgs received, or other events

network protocols:

- machines rather than humans
- all communication activity in Internet governed by protocols

protocols *define format, order of msgs sent and received among network entities, and actions taken on msg transmission, receipt*

What's a protocol?

a human protocol and a computer network protocol:

Hi

Hi

Got the time?

2:00

TCP connection request

TCP connection response

Get http://www.awl.com/kurose-ross

<file>

time

<u>Q</u>: Other human protocols?

International Standards Organization
Open Systems Interconnect (OSI) Reference Model

Network Architecture
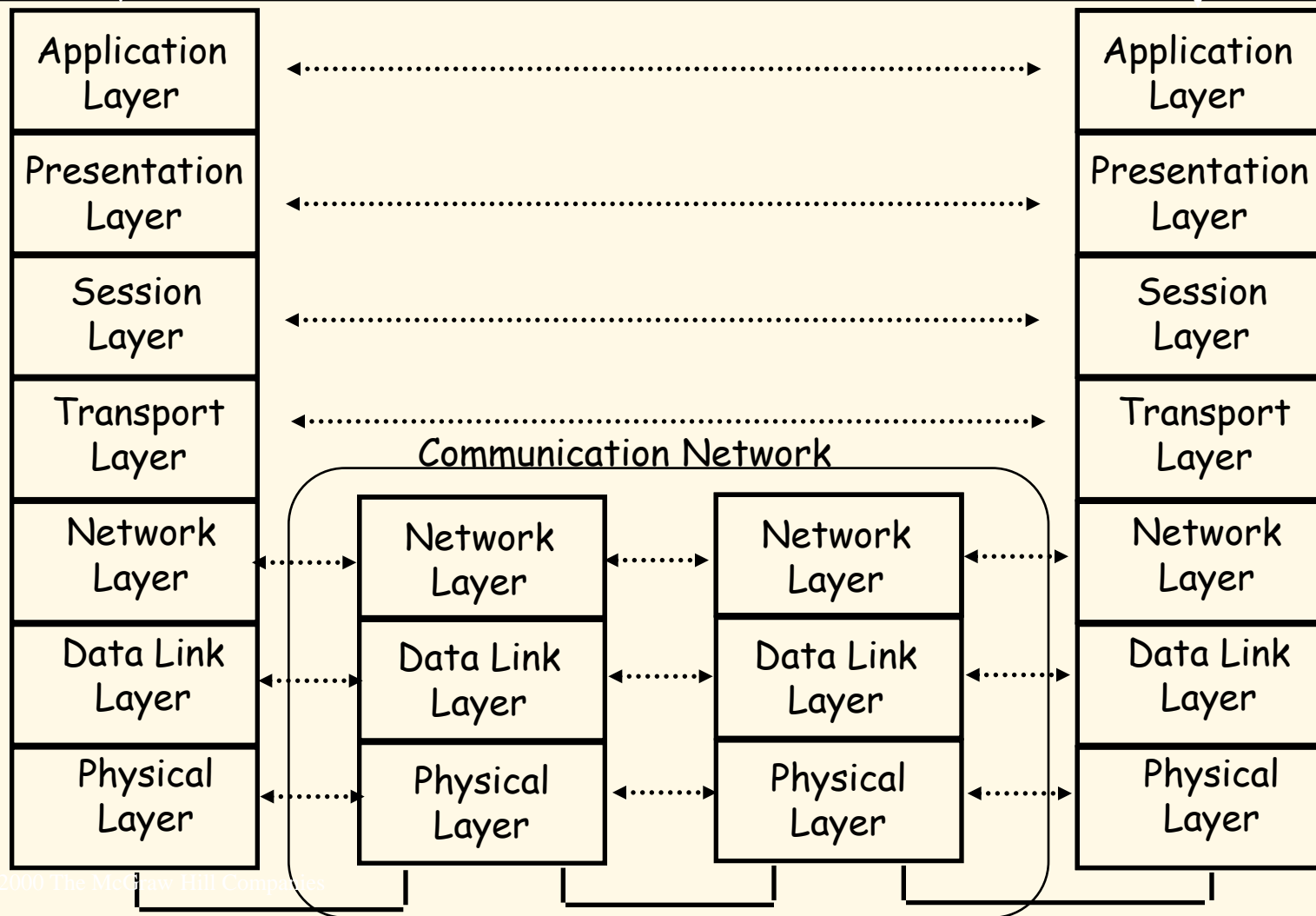
The OSI Model

OSI Layer Encapsulation

Application A
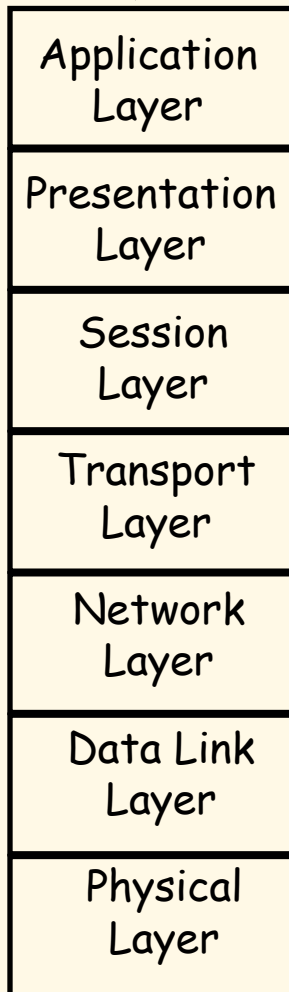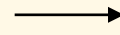
Application B
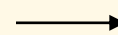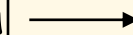
| data | → |

| data | ah | → |

| data | ph | → |

| data | sh | → |

| data | th | → |

| data | nh | → |

| dt | data | dh | → |

| bits | → |
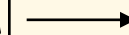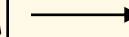
Application A (layers, top to bottom):
- Application Layer
- Presentation Layer
- Session Layer
- Transport Layer
- Network Layer
- Data Link Layer
- Physical Layer

Application B (layers, top to bottom):
- Application Layer
- Presentation Layer
- Session Layer
- Transport Layer
- Network Layer
- Data Link Layer
- Physical Layer
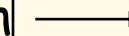
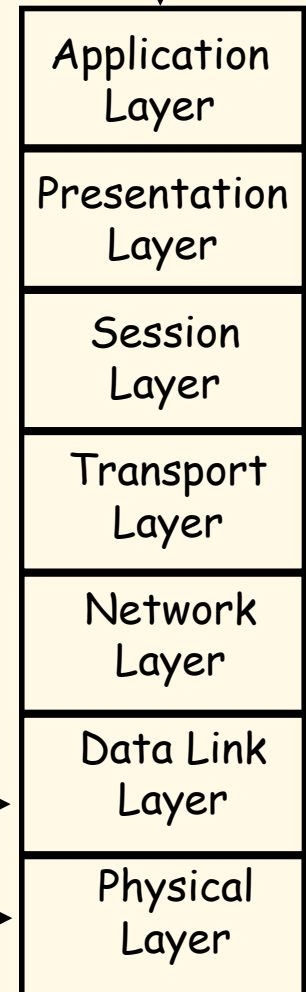**Leon-Garcia & Widjaja: *Communication Networks***

Seven Layer OSI Model

Application Layer
Provides users access to the OSI environment and distributed information services.

Presentation Layer
Provides application processes independence from differences in data representations.

Session Layer
Provides the control structure for communicating between applications.
Establishes, manages and terminates session connections between cooperating applications.

Transport Layer
Provides reliable transparent transfer of data between end points.
Provides end-to-end flow control and error recovery.

Network Layer
Provides independence from the data transmission, routing/switching technologies used
to connect systems. Responsible for establishing, managing and terminating connections.

Data Link Layer
Provides for reliable transfer of information across the physical layer. Sends
and receives frames with the necessary synchronization, flow control and error control.

Physical Layer
Concerned with transmission of unstructured bit stream over a physical medium. Deals with
mechanical, electrical, functional and procedural characteristics to access the physical medium.

ISO/OSI Reference Model

- **Presentation layer:** allow applications to interpret meaning of data, e.g., encryption, compression, machine-specific conventions
- **Session layer:** synchronization, check-pointing, recovery of data exchange
- The TCP/IP Internet stack is "missing" these two layers!
  - these services, *if needed,* must be implemented in an application
  - needed?

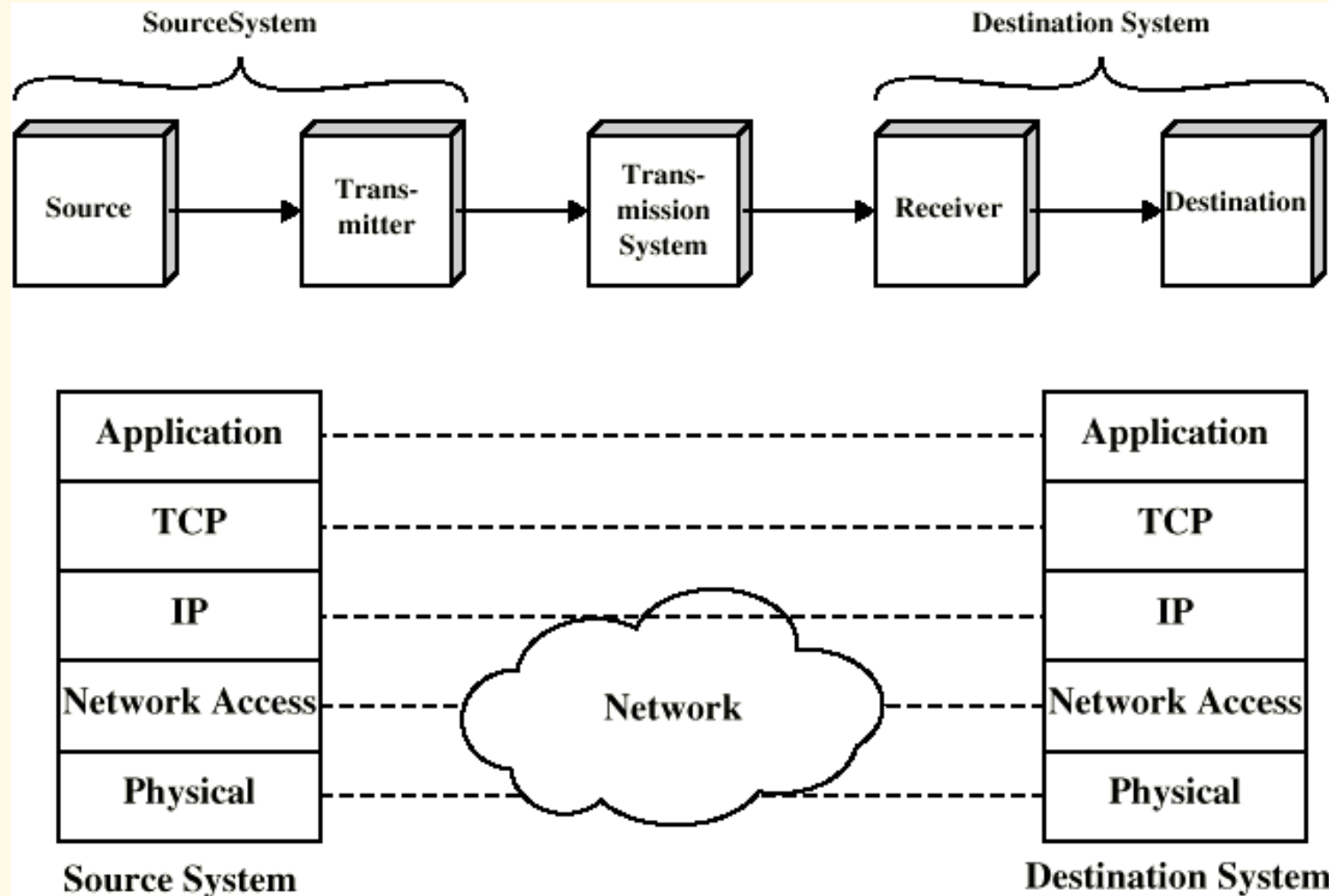| application |
| --- |
| presentation |
| session |
| transport |
| network |
| data link |
| physical |

K & R

WPI
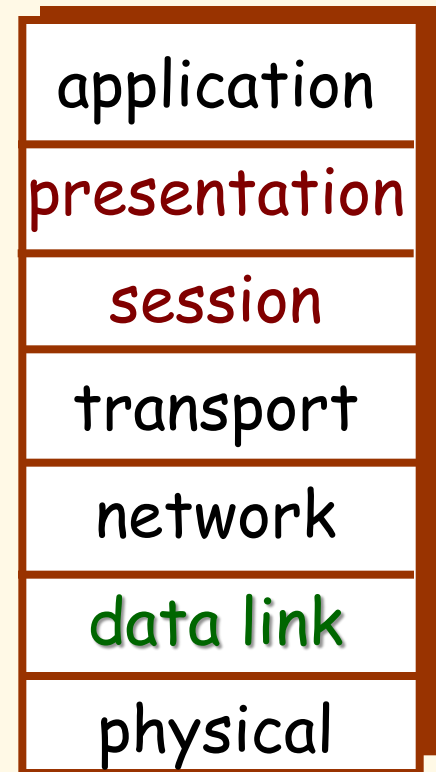
Advantages of Layering Design

- An explicit structure for dealing with a complex system:
  - allows identification and structures the relationship of complex system's pieces.
  - layered **reference model** for discussion.
- Provides an abstraction for functional locality.
- Simplifies the design process.

Advantages of Layering Design

- Modularity of layers eases maintenance and updating of system components:
  - change in implementation of a layer's service is transparent to rest of the system.
  - Led to flexibility in modifying and developing network architectures.
  - Accommodates incremental changes.

TCP/IP Architectural Model

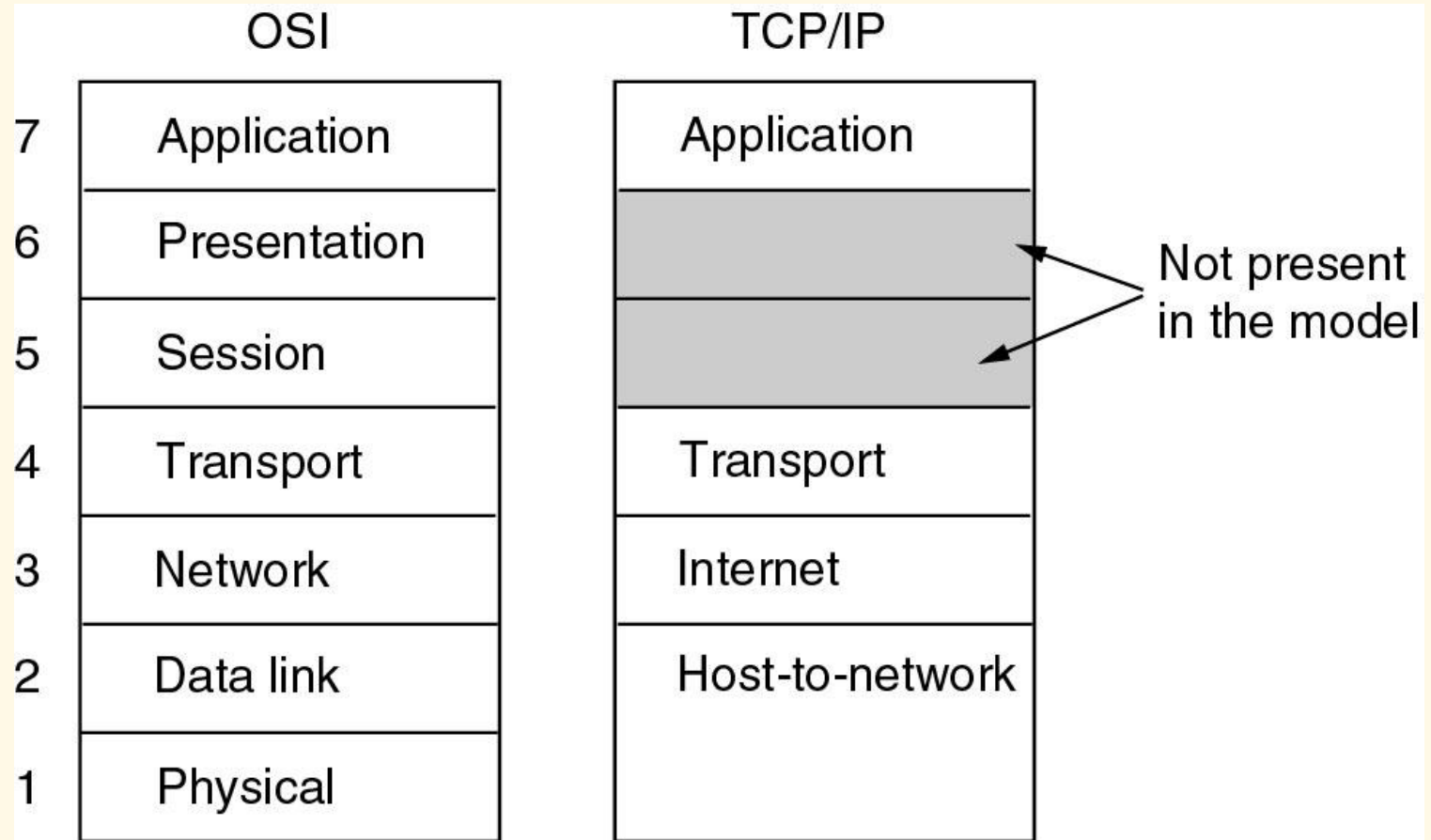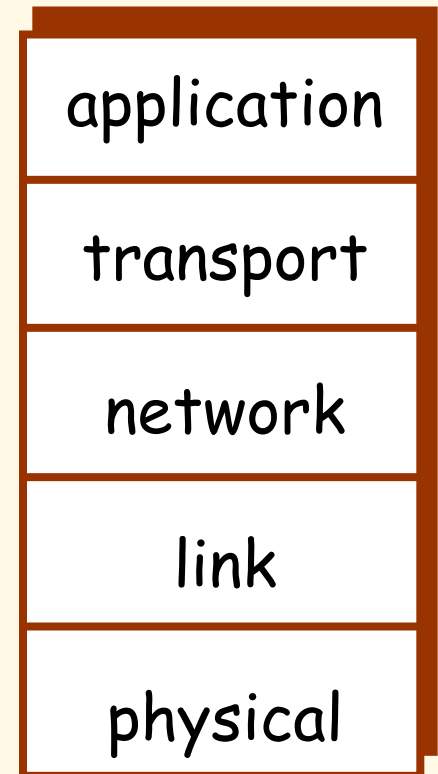DCC 6th Ed., W. Stallings

OSI versus TCP/IP

Figure 1-21. [old] The TCP/IP reference model.

Tanenbaum

Internet Protocol Stack

- **application:** supporting network applications
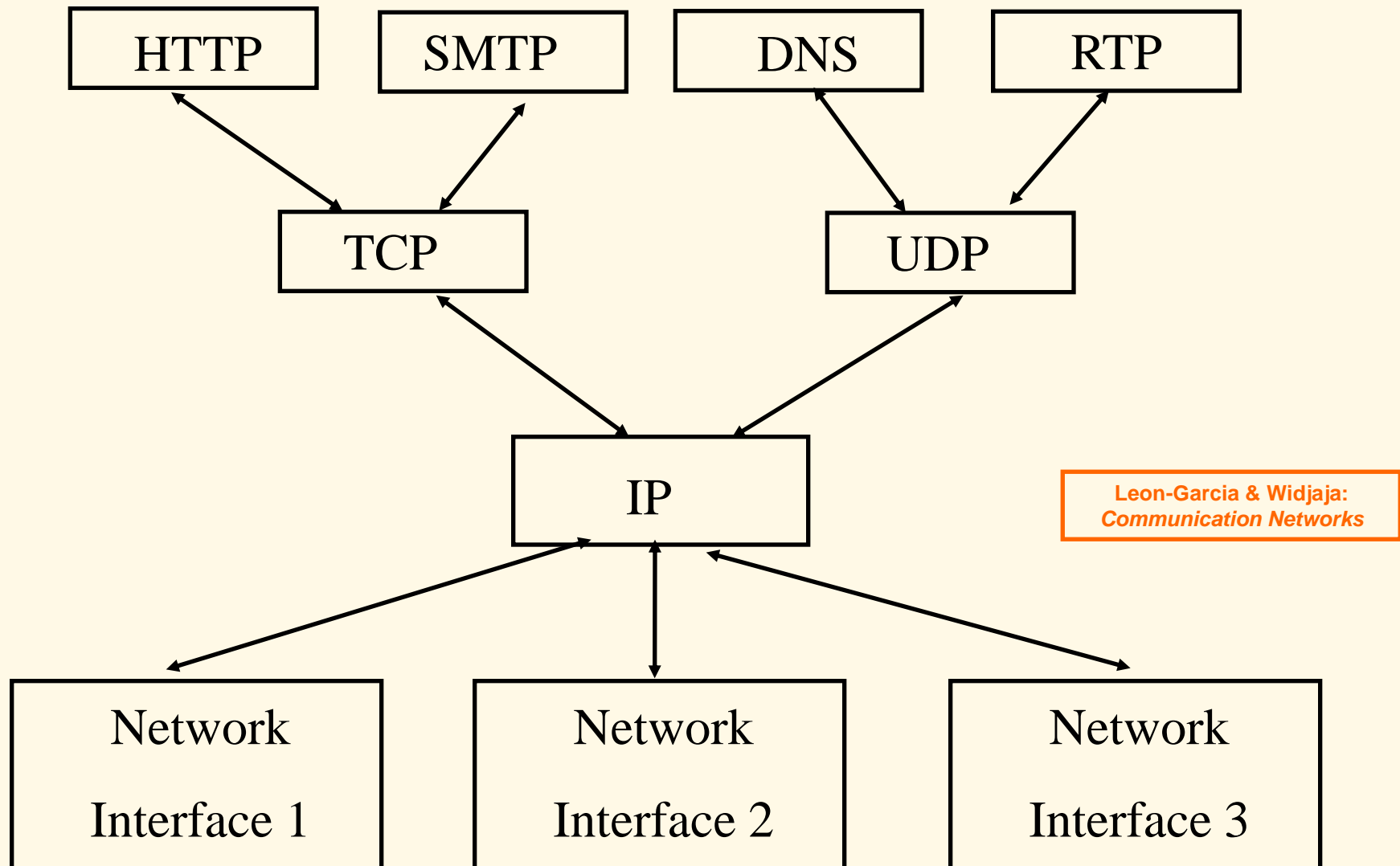 - FTP, SMTP, HTTP
- **transport:** process-process data transfer
  - TCP, UDP
- **network:** routing of datagrams from source to destination
  - IP, routing protocols
- **link:** data transfer between neighboring network elements
  - PPP, Ethernet
- **physical:** bits "on the wire" or in the air

application
transport
network
link
physical

K & R

TCP/IP Protocols

HTTP   SMTP   DNS   RTP

TCP   UDP

IP

Leon-Garcia & Widjaja:
Communication Networks

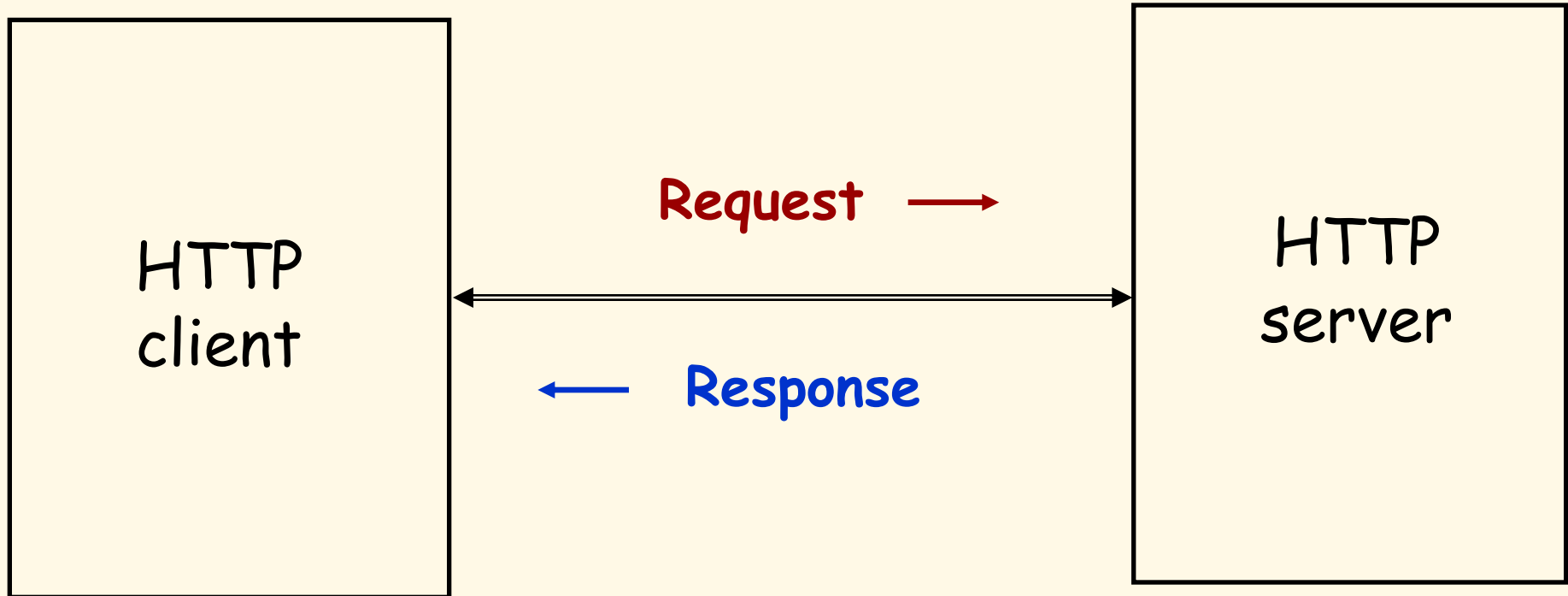Network Interface 1   Network Interface 2   Network Interface 3

Layering Example

Client/server relationship
- Server process waits for incoming requests by listening to a **port**.
- Client process makes requests as required.
- Server process provides responses to these requests.
- The server process usually runs in the background as a daemon (e.g. httpd is the server daemon). for HTTP).

HTTP Example

- **HTTP (HyperText Transfer Protocol)** specifies rules by which the client and the server interact so as to retrieve a document.

- The protocol assumes the client and the server can exchange messages directly.

- The client software needs to set up a two-way connection prior to the HTTP request.
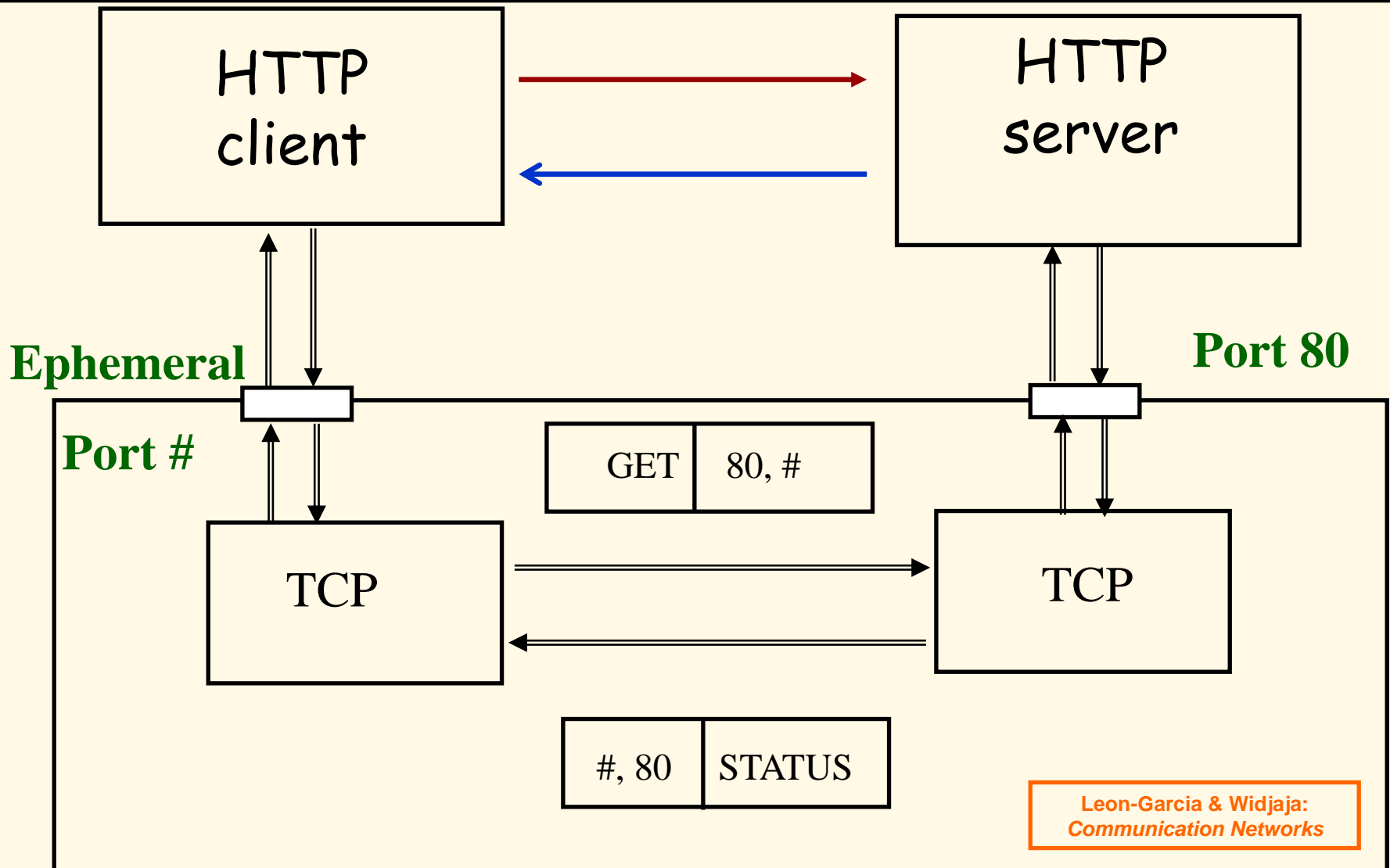
HTTP Client/Server Interaction

HTTP client   **Request** ⟶   HTTP server

⟵ **Response**

Leon-Garcia & Widjaja:
Communication Networks

HTTP/TCP Layering Interface

Leon-Garcia & Widjaja:
Communication Networks

HTTP Encapsulation Example

HTTP Request

Header contains source and destination port numbers

| TCP Header | |

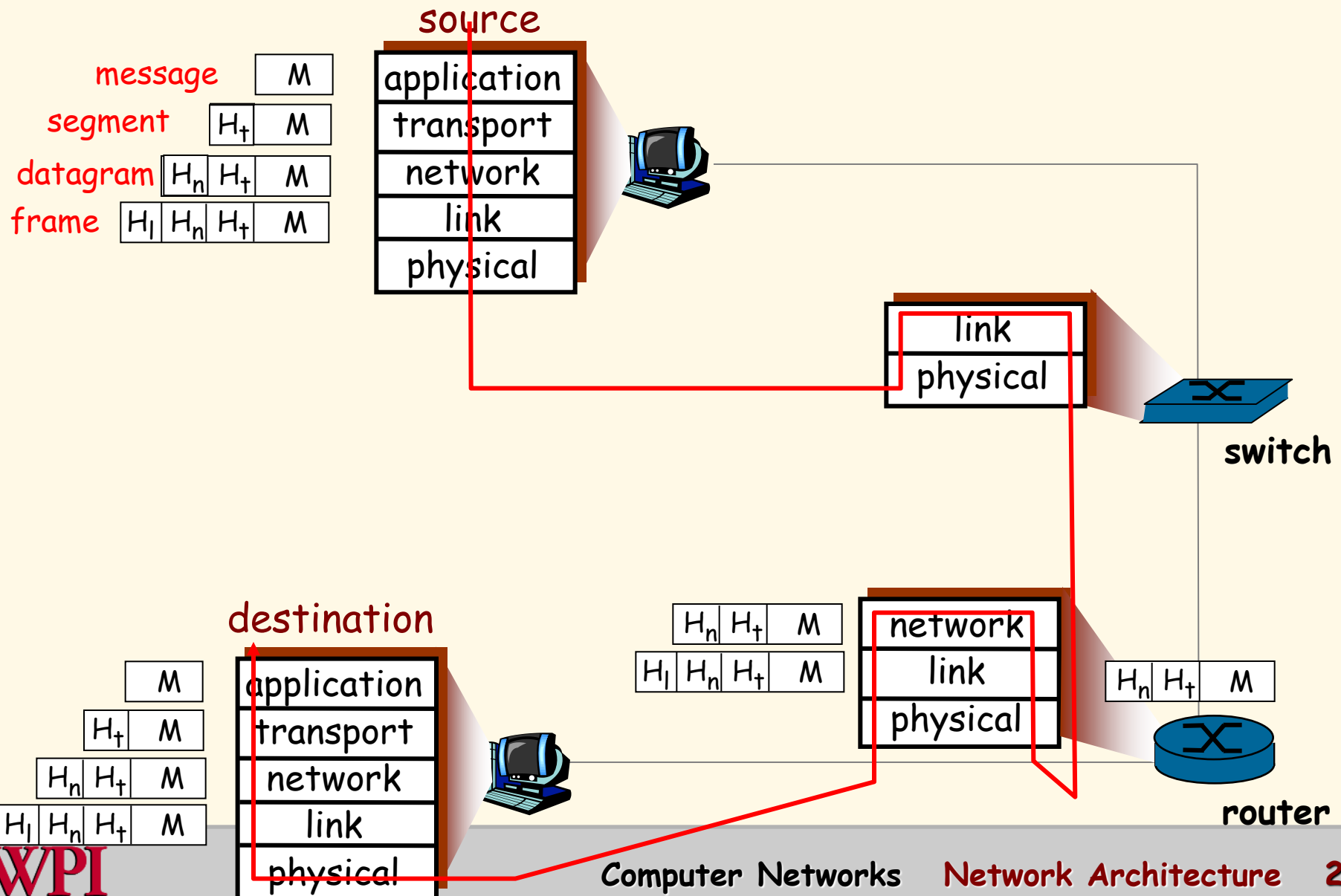Header contains source and destination IP addresses; transport protocol type

| IP Header | |

Header contains source and destination physical addresses; network protocol type

| Ethernet Header | | Frame Check Sequence |

Encapsulation Animation

source

message				M
segment			H_t	M
datagram		H_n	H_t	M
frame	H_l	H_n	H_t	M

source

| application |
| transport |
| network |
| link |
| physical |

| link |
| physical |

switch

destination

M			
H_t	M		
H_n	H_t	M	
H_l	H_n	H_t	M

| application |
| transport |
| network |
| link |
| physical |

| H_n | H_t | M |
| H_l | H_n | H_t | M |

| network |
| link |
| physical |

| H_n | H_t | M |

router
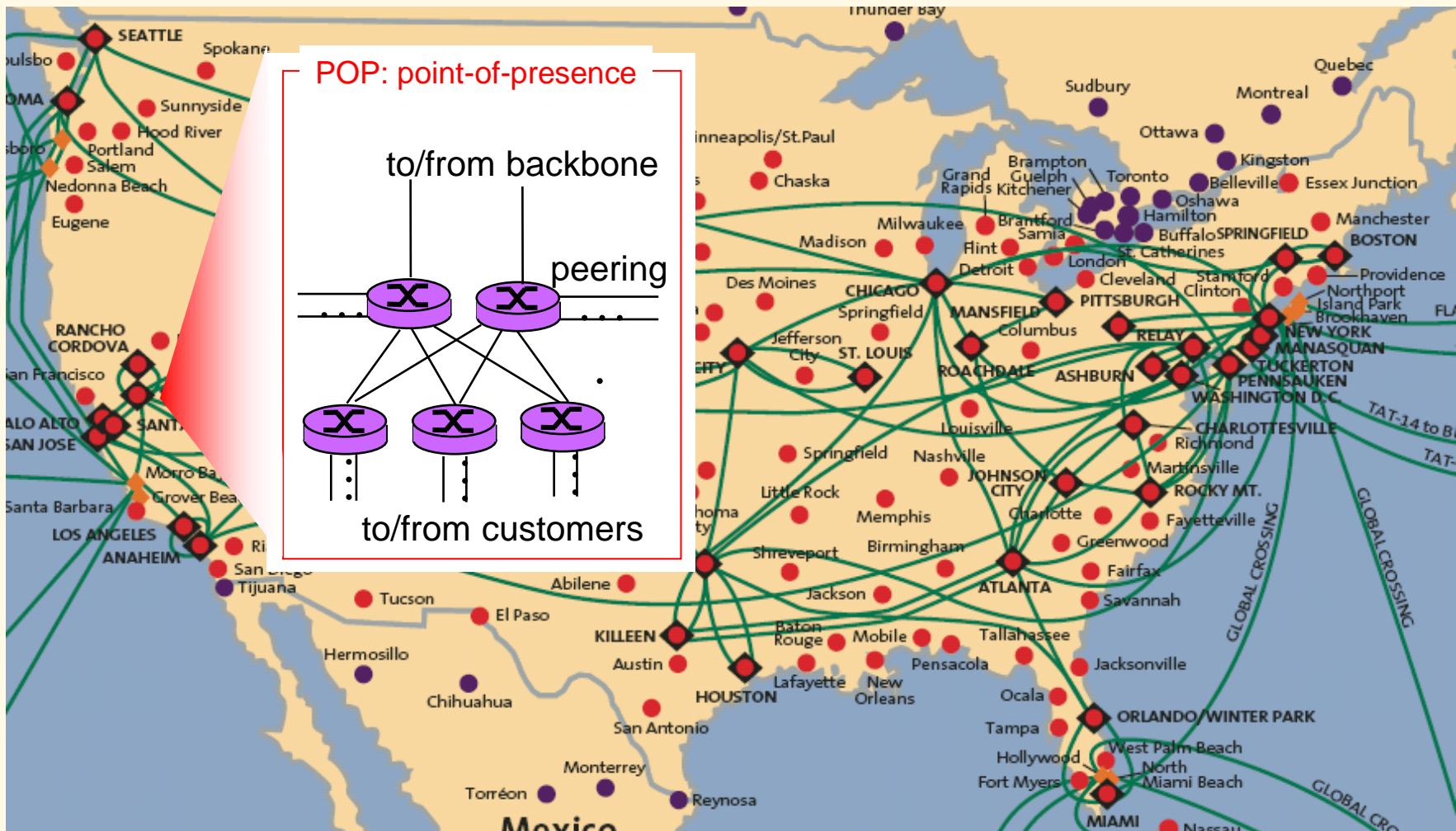
Internet Structure: Network of Networks

- **roughly hierarchical**

- **at center: "tier-1" ISPs** (e.g., Verizon, Sprint, AT&T, Cable and Wireless), national/international coverage
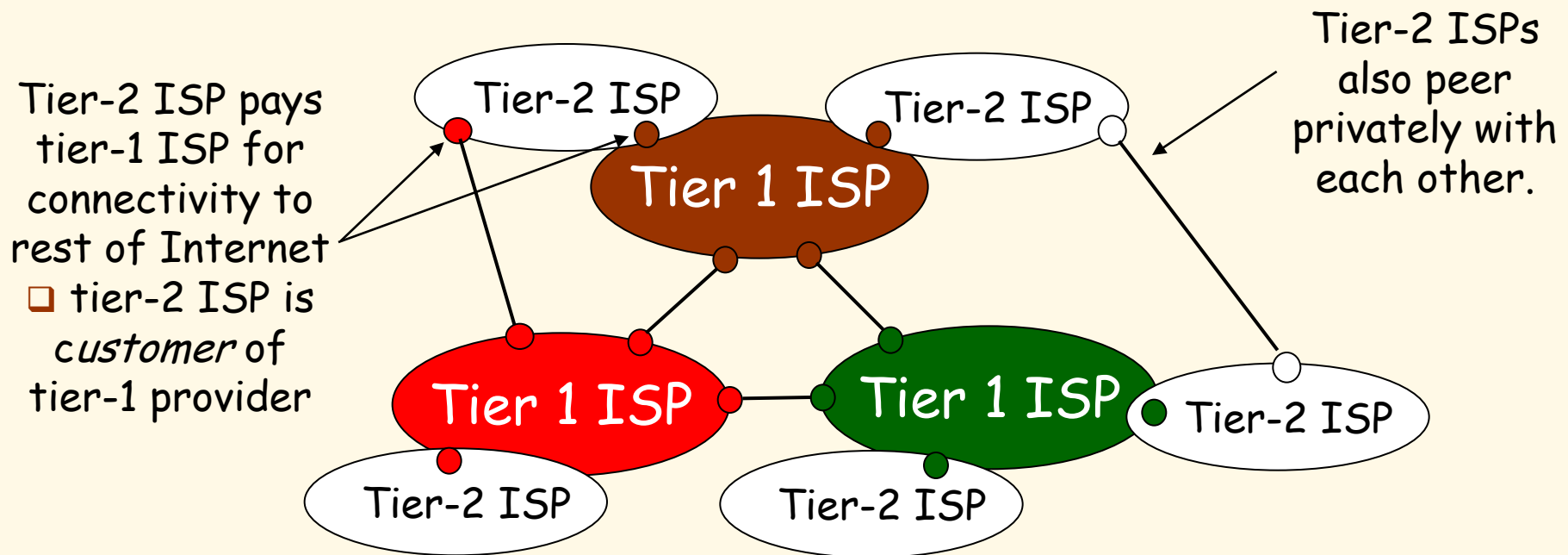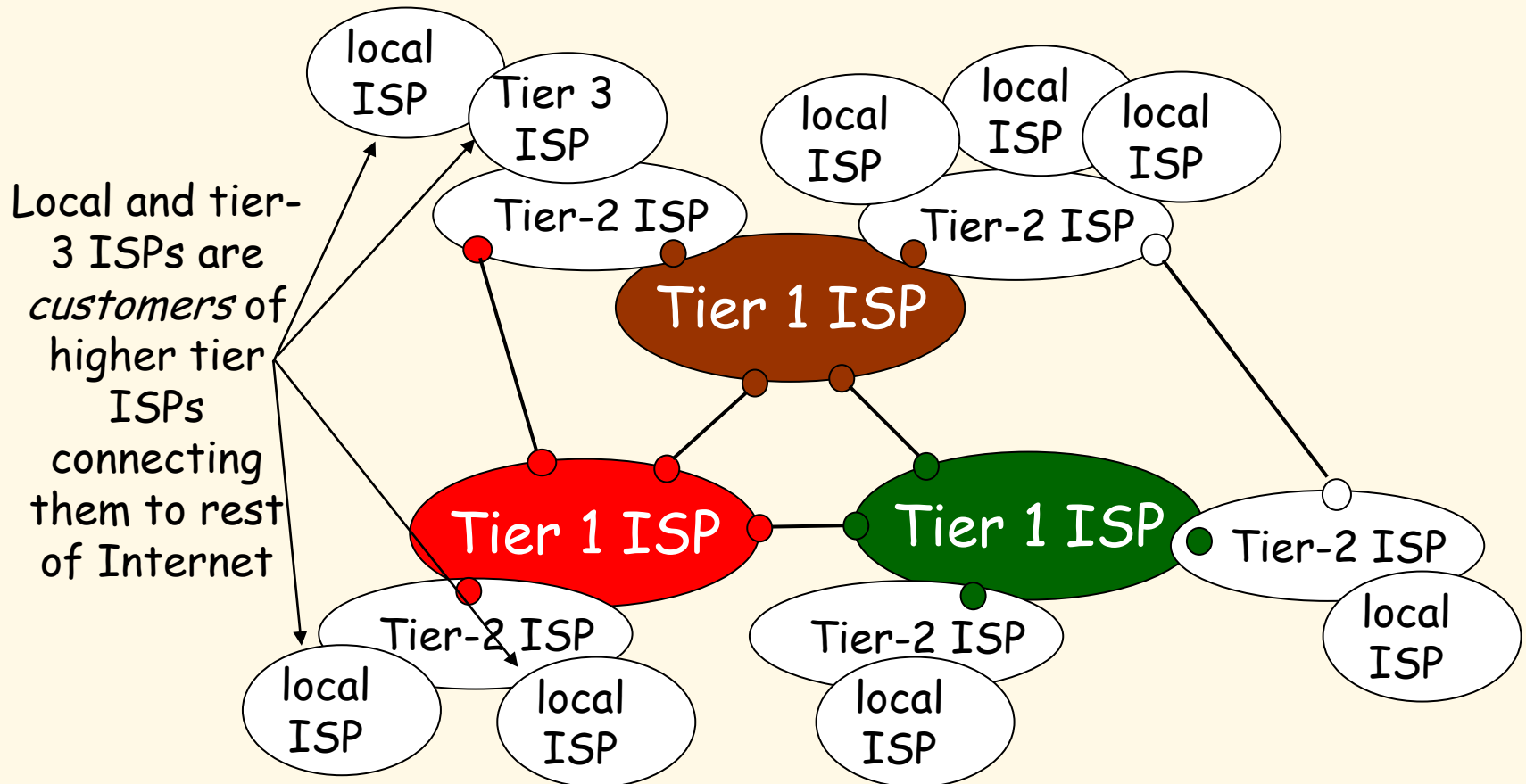
  - **treat each other as equals**

Tier-1 providers interconnect (peer) privately

Tier 1 ISP

Tier 1 ISP

Tier 1 ISP

WPI

POP: point-of-presence

to/from backbone

peering

to/from customers

- **"Tier-2" ISPs: smaller (often regional) ISPs**
  - **Connect to one or more tier-1 ISPs, possibly other tier-2 ISPs**

Tier-2 ISP pays tier-1 ISP for connectivity to rest of Internet

❑ tier-2 ISP is *customer* of tier-1 provider

Tier-2 ISPs also peer privately with each other.

Tier-2 ISP

Tier-2 ISP

Tier 1 ISP

Tier 1 ISP

Tier 1 ISP

Tier-2 ISP

Tier-2 ISP

Tier-2 ISP

- **"Tier-3" ISPs and local ISPs**
  - **last hop ("access") network (closest to end systems)**

Local and tier-3 ISPs are *customers* of higher tier ISPs connecting them to rest of Internet

local ISP

Tier 3 ISP

local ISP

local ISP

local ISP

Tier-2 ISP

Tier-2 ISP

Tier 1 ISP

Tier 1 ISP

Tier 1 ISP

Tier-2 ISP

Tier-2 ISP

local ISP

local ISP

local ISP

local ISP

- a packet passes through many networks!
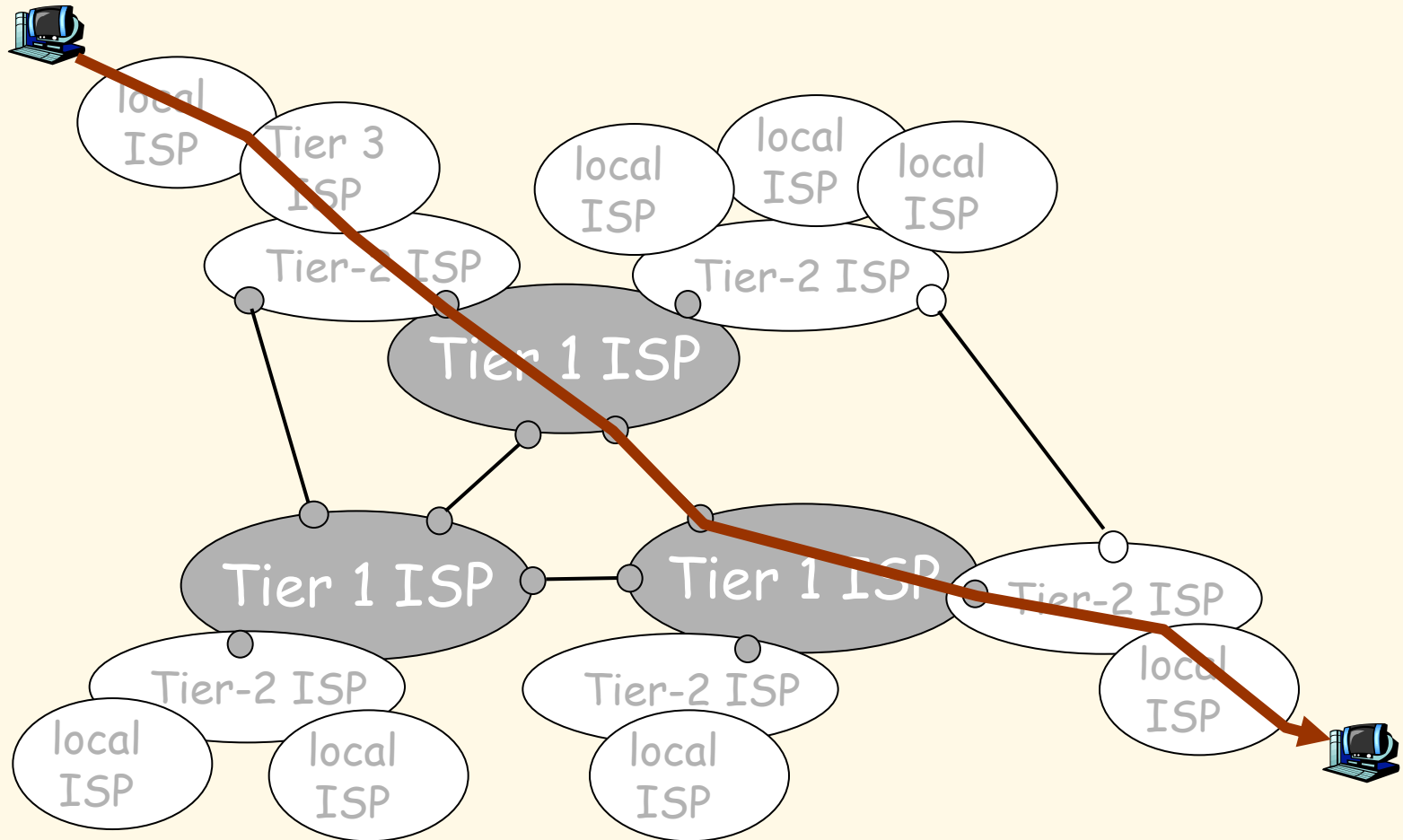
Network Architecture Summary

- **Introduction to IP protocol design**
- **Network architecture**
- **The OSI reference model**
- **The TCP/IP Stack**
- **Layering example**
- **Encapsulation examples**
- **Modern tiered Internet architecture**