Chapter 1: Conceptual Basis

Section 1

V 2.5

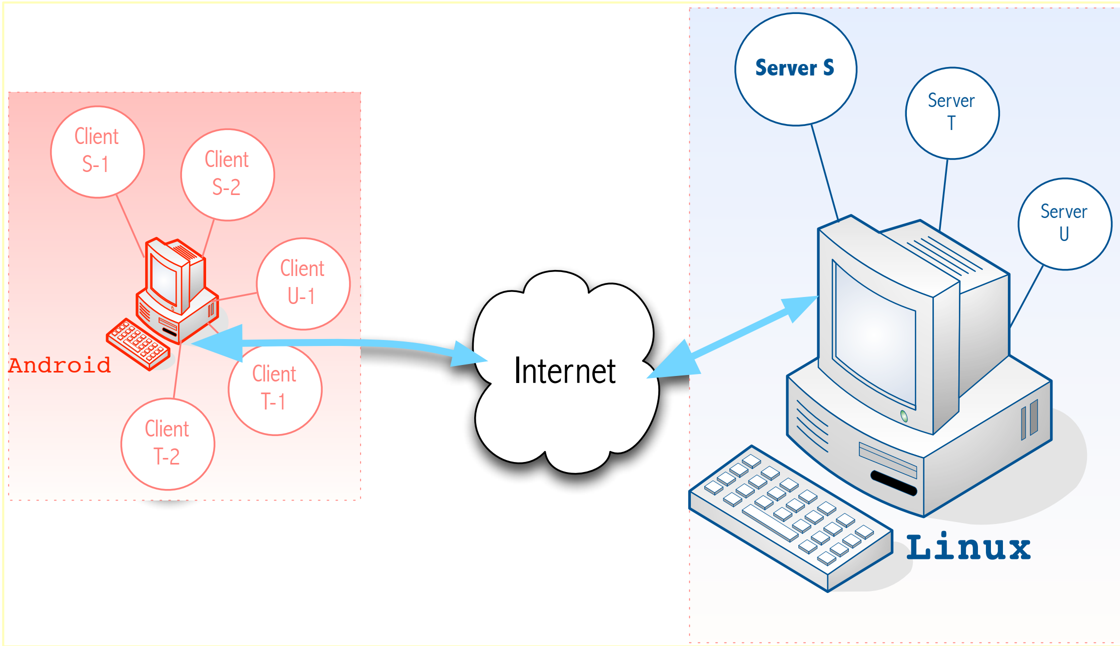
Based on textbook *Conceptual Computer Networks* by:

Leading questions

- What are the principles behind the communication between two parties?
- When can communication be considered fast and efficient?
- What is a network architecture?
- What are the landmarks in the development of Internet?
- Why is networking essential for the progress of humankind?

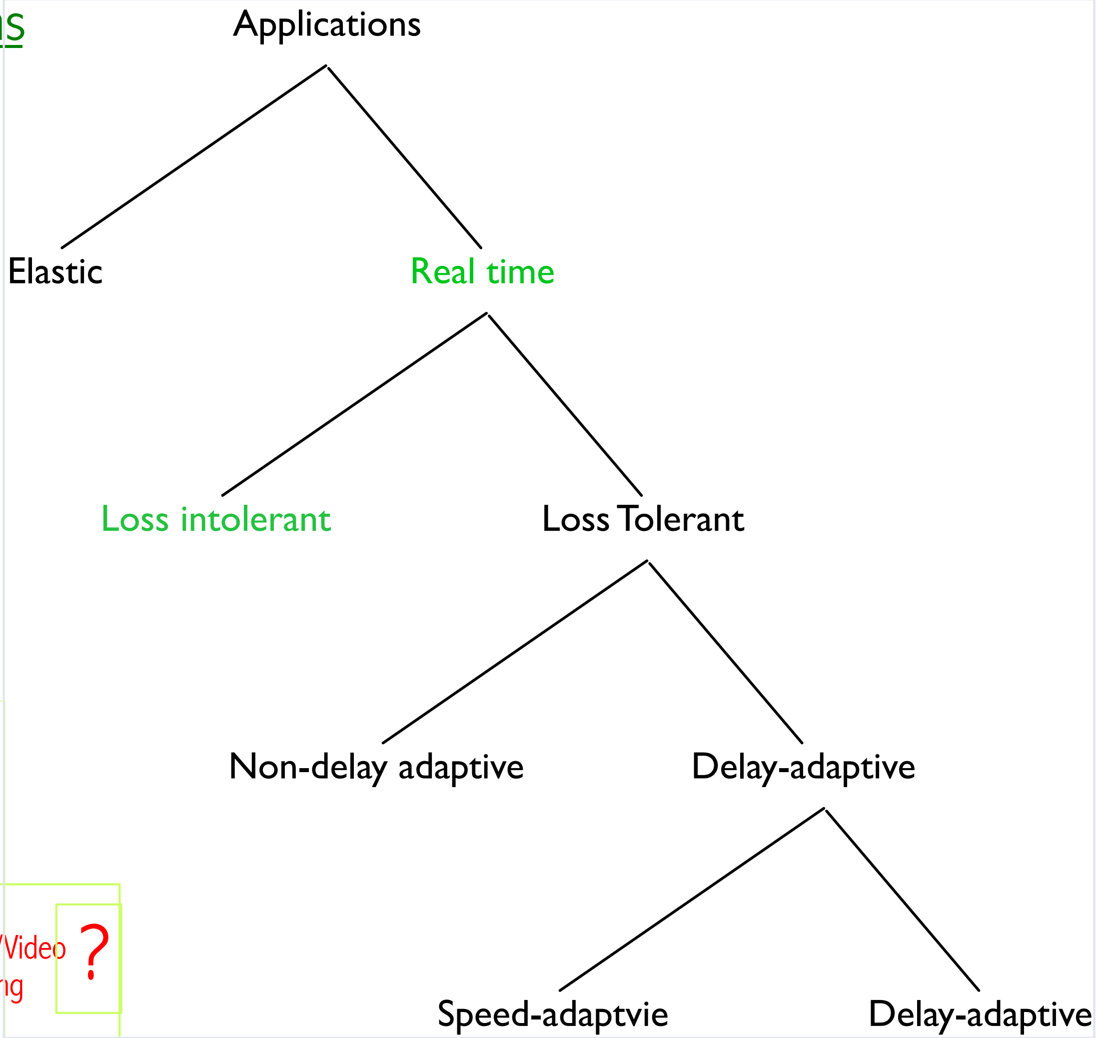
Hosts execute processes that can mutually communicate

- Internet Host
  - Computer System
  - OS
    - PS
    - FS
    - NS (Inet Protocol Stack)
      - TCP, UDP, ICMP, IP, Subnet…
  - NIC
- Collaboration among hosts made possible



Host applications collaborate

- Applications are computer programs
- Communicate over the Internet
- At work, at home and mobile
- Vastly differing **requirements:**
  - Omission faults (Loss of packets)
  - Bandwidth, speed (bps; pps, etc.)
  - Delay (s)
  - Variance of delay (Jitter; s)



- e-mail: smtp, pop3, imap
- File Sharing, file transfer: ftp, rcp, scp
- Printer sharing
- Virtual terminal: telnet, ssh (Secure Shell)
- e-commerce
- Geolocation
- **World Wide Web (www)**
- Social Networks
- Instant Messaging (Whatsup, . . .)

- Network storage
- Streaming Audio/Video
- Video conferencing
- IP Telephony

?

© Summary diagram by Larry Peterson and Bruce Davie (From Clark, Schenker and Zangh)

Example of an Internet application: World Wide Web

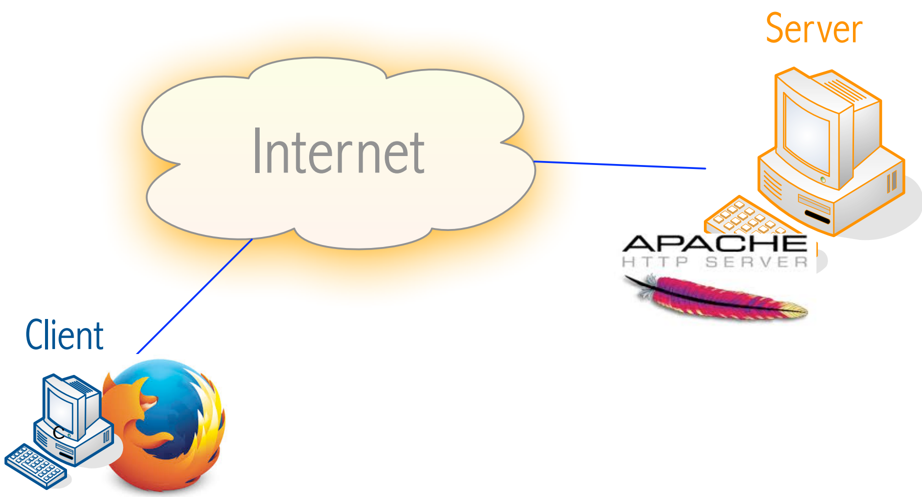
- Web pages are downloaded by the **client** from the **server**
  - Client runs in an Inet host
  - Server also runs in an Inet host
  - Thus, Client and server can collaborate
  - Client and server *speak* the http protocol
  - http = Hyper Text Transfer Protocol

- WWW
  - Server program (e.g., Apache)
  - Client program (e.g., Firefox)

- URL
  - Uniform Resource Locator
  - Example of a URL:

  http://paloalto.unileon.es/cn/index.html

- HTTP, in turn uses the TCP protocol for reliability
  - TCP = Transmission Control Protocol
  - TCP provides reliability
    - In case of packet loss, duplication, errors, etc



Units and multipliers

- Bandwidth
  - Directly related to the acceptable speed of bit transmission over some medium
  - Number of bits transmitted in one second:
    - Bps (Bits Per Second = Bits/Sec)
  - Since bandwidth is a rate, the multipliers take on the following values:
    - K (Kilo = 10^3)
    - M (Mega = 10^6)
    - G (Giga = 10^9)
    - T (Tera = 10^{12})
- Throughput
  - **Number of packets that can reliably come across some section of an internetwork in every second**
    - **Packets/sec**
    - **Same multipliers as above**
- Delay
  - Seconds
  - How much time it takes to transport one bit from a source to a destination directly connected
  - Propagation delay
- Jitter
  - The variance of the delay

Units and multipliers, standard table

System of Units (SI)			Binary Numeral			
Factor	Name	Symbol	Factor	Name	Symbol	# of Bytes
10^3	kilobyte	KB	2^{10}	kibibyte	KiB	1,024
10^6	megabyte	MB	2^{20}	mebibyte	MiB	1,048,576
10^9	gigabyte	GB	2^{30}	gibibyte	GiB	1,073,741,824
10^{12}	terabyte	TB	2^{40}	tebibyte	TiB	1,099,511,627,776
10^{15}	petabyte	PB	2^{50}	pebibyte	PiB	1,125,899,906,842,624
10^{18}	exabyte	EB	2^{60}	exbibyte	EiB	1,152,921,504,606,846,976
10^{21}	zettabyte	ZB	2^{70}	zebibyte	ZiB	1,180,591,620,717,411,303,424
10^{24}	yottabyte	YB	2^{80}	yobibyte	YiB	1,208,925,819,614,629,174,706,176

Examples

- Transmission speed: bps, SI

- Information: bits, powers of 2

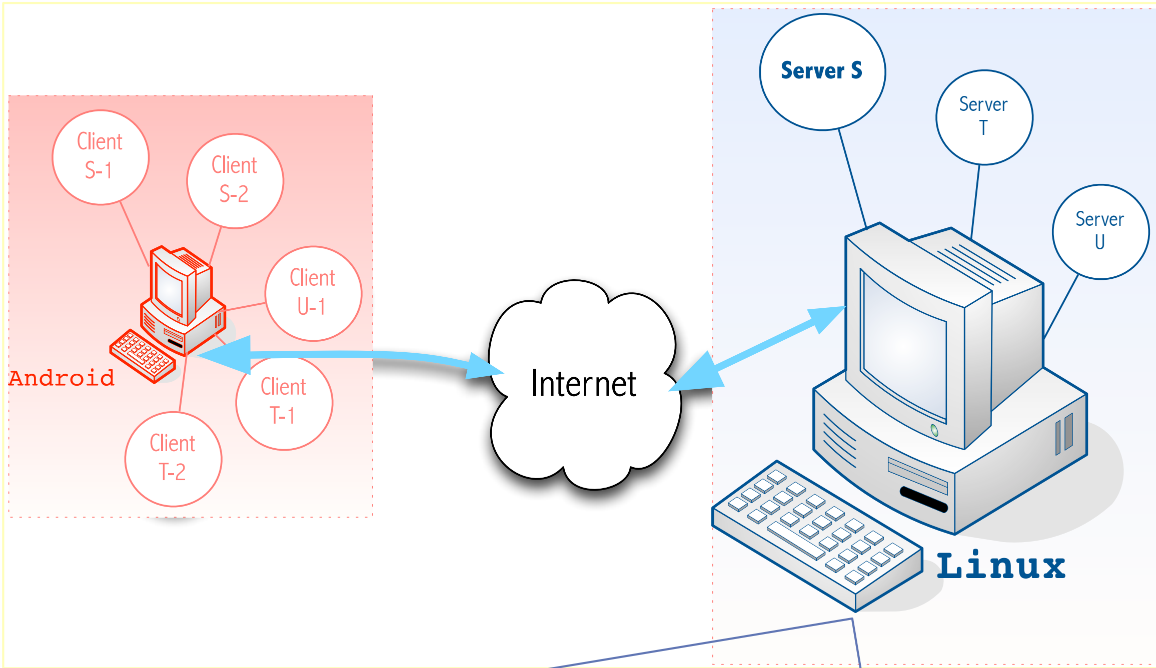
9

Network Architecture

Manage the complexity of networks

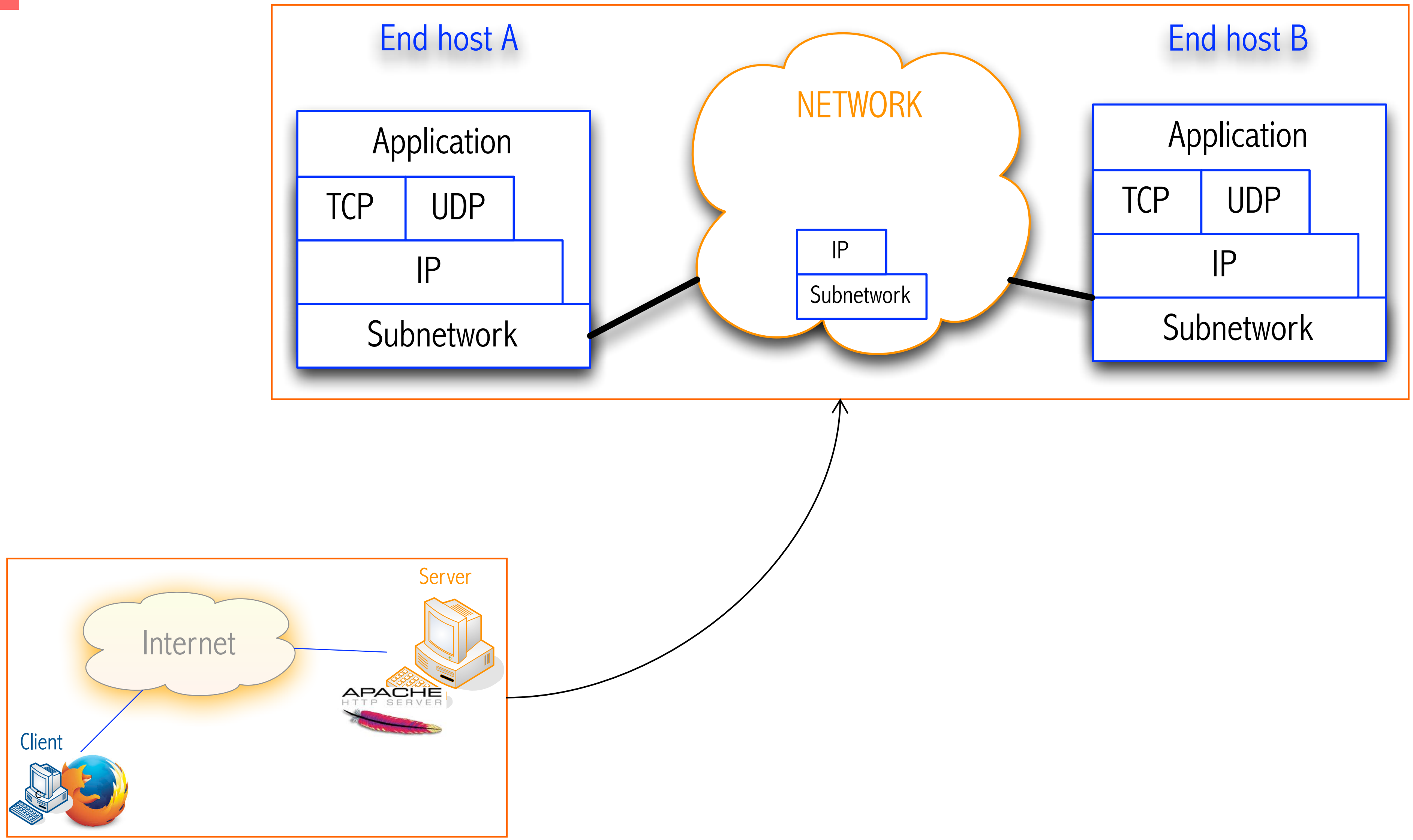
Logical channels

- Applications communicate over the Internet
- The channel between two communicating applications is logical
- Each channel:
  - *Connects* two applications
  - Hosts must be identified:
    - IP address
  - Applications must be identified:
    - Port numbers



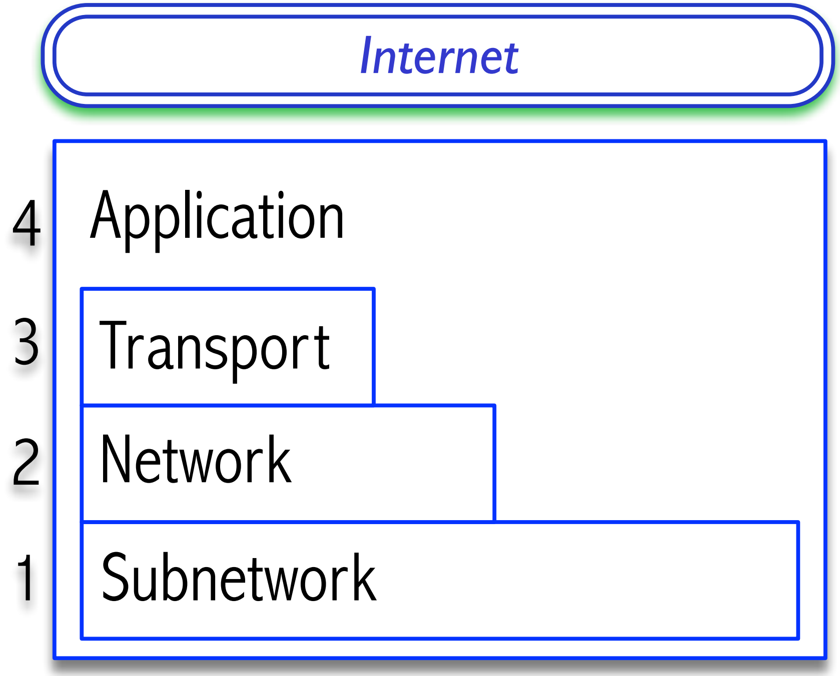
How to **hide** network **complexity** from application **programmers?**

Layering in hosts and network



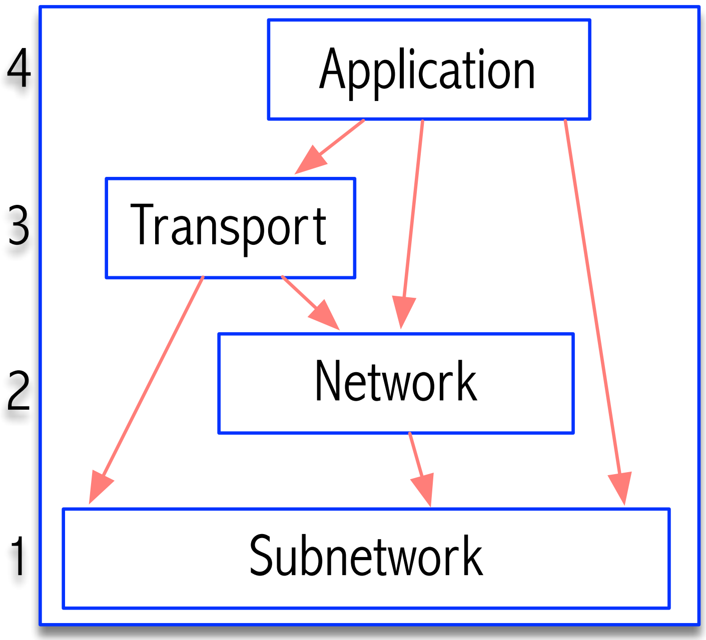
Internet Architecture

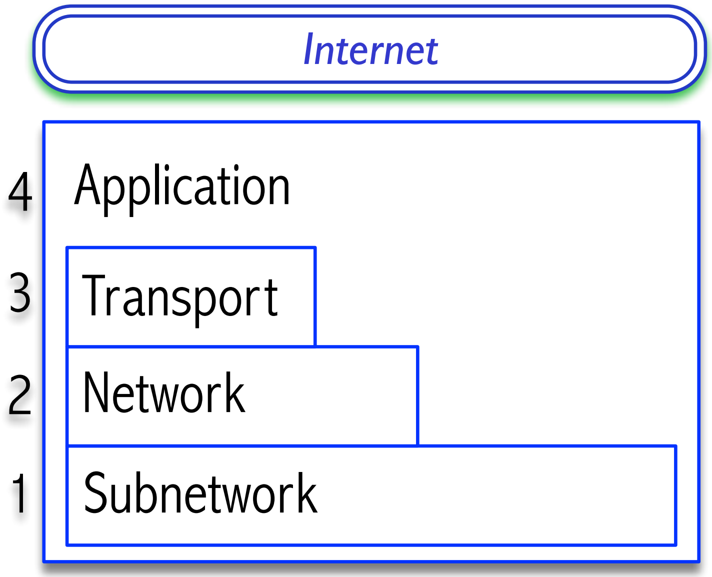
- Network complexity is broken down into 4 layers
  - Each *layer*
    - Offers a set of services to the upper layers
    - The mechanism that attains each service is a protocol
    - An upper layer avails one service from a lower layer by calling its interface



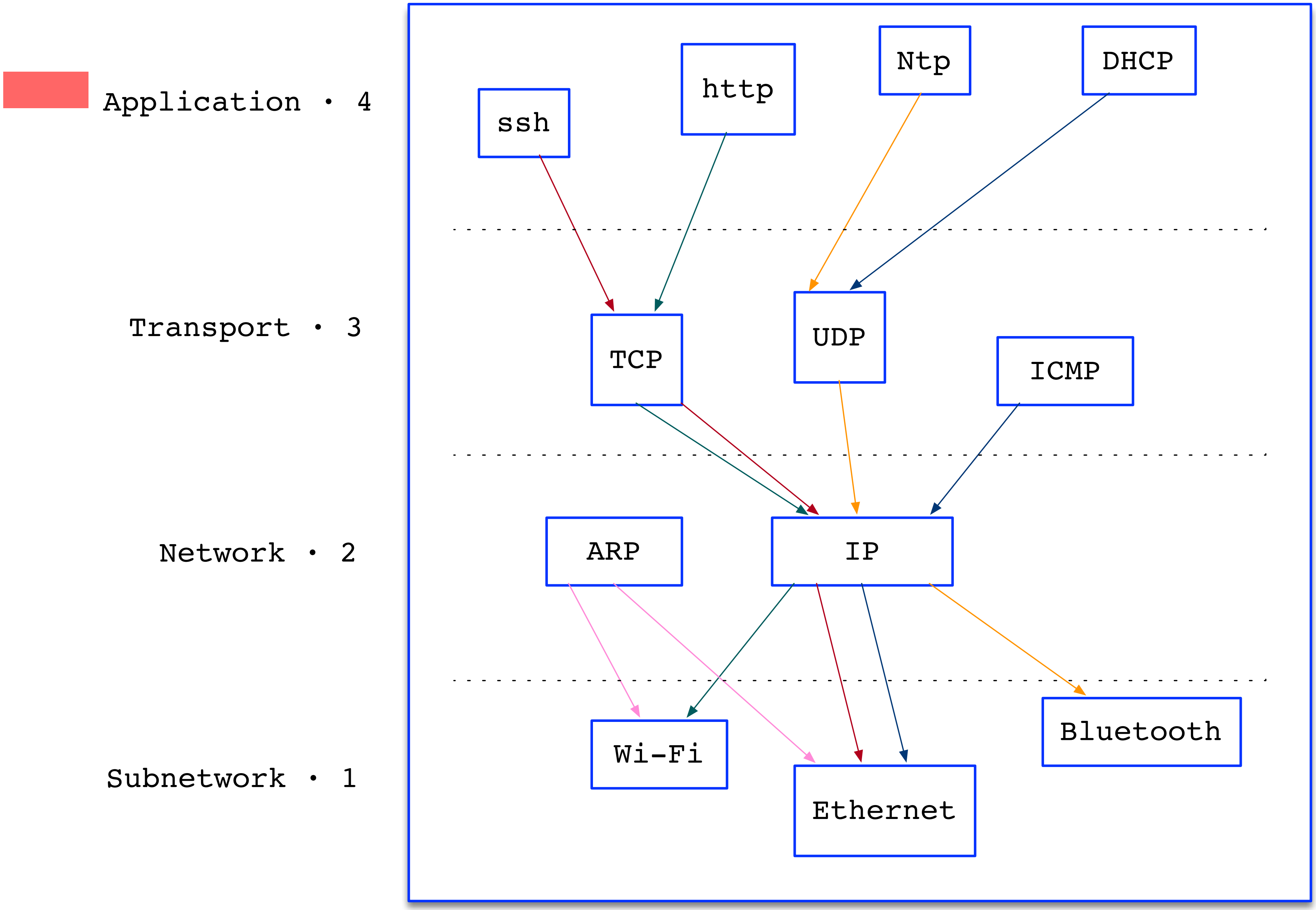
- 1. Subnetwork: Ethernet, Wi-Fi, Bluetooth
- 2. Network: Only IP !!!
- 3. Transport: TCP and UDP
- 4. Application: Whatsup and innumerably others

Internet Architecture

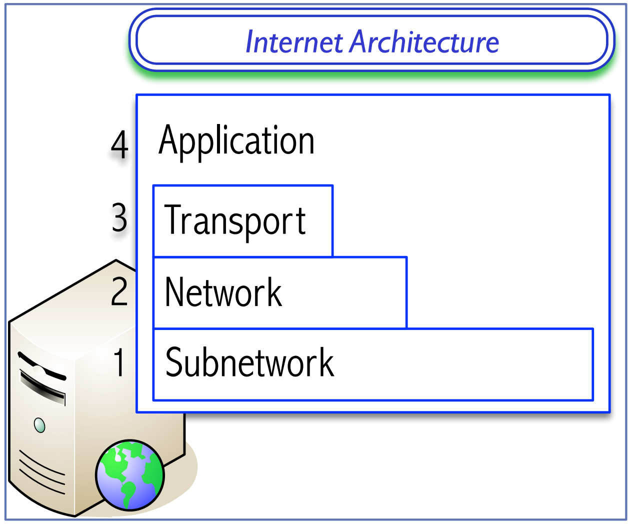


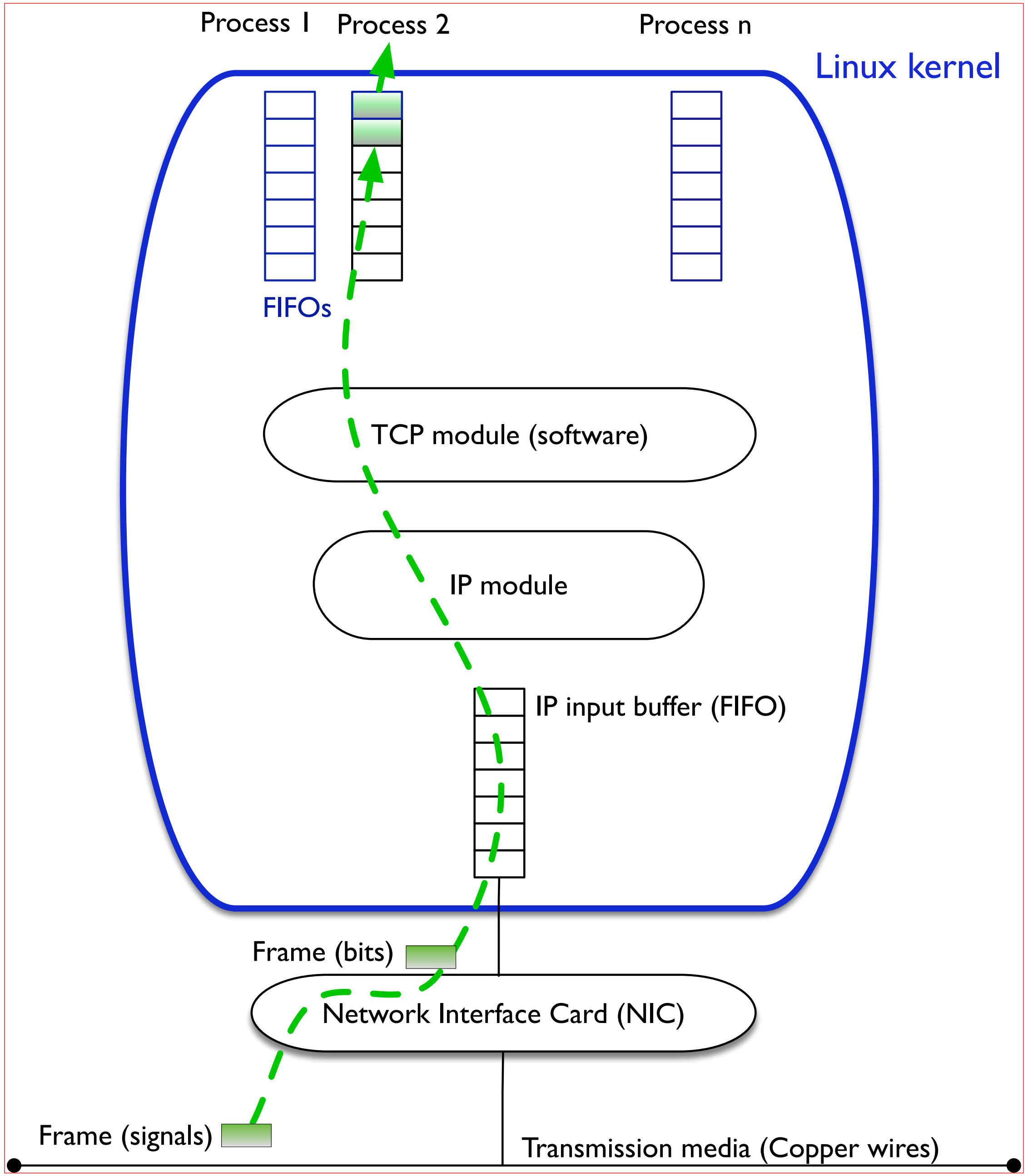


Typical Internet Protocol Stack



Implementation of protocols





Internet Architecture, standard

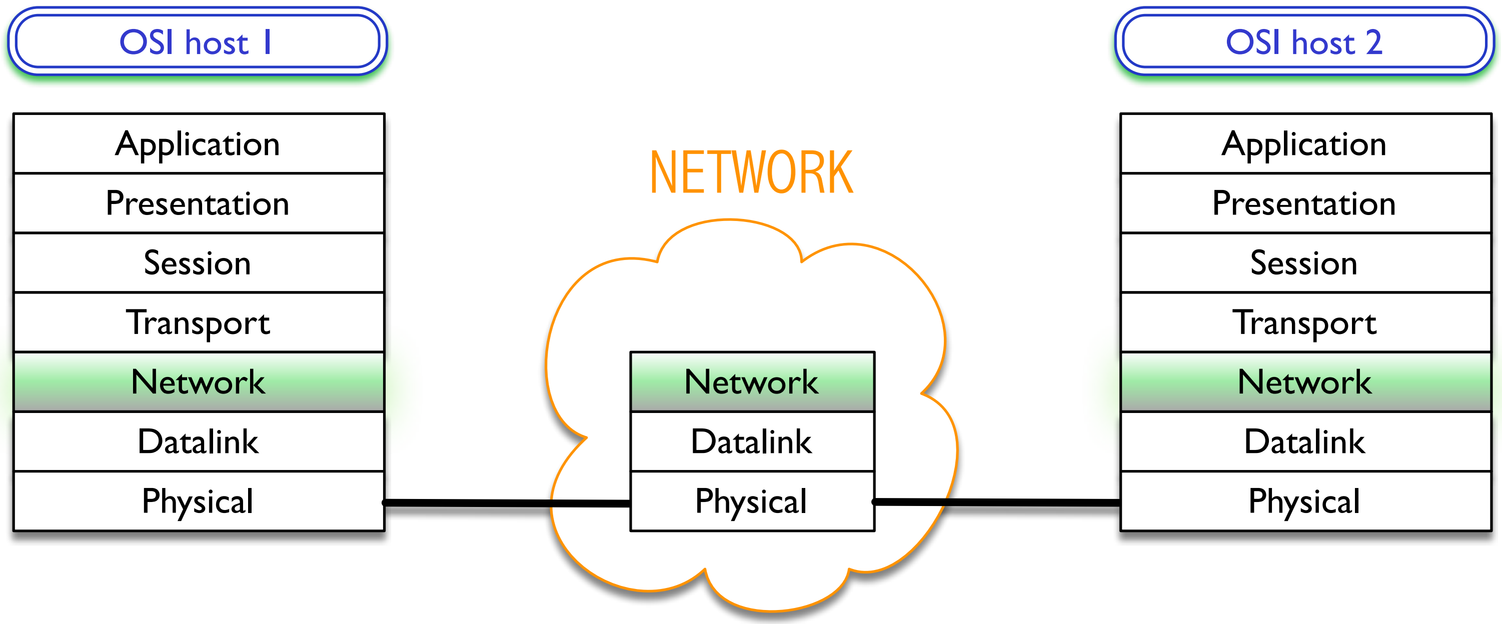
- Specified by Internet Engineering Task Force in 1970
- RFC 1287
- Derived from the TCP/IP Protocol suite
  - In any implementation of IA, programs can call any layer's service interface
  - Only one network protocol: IP
    - Many application protocols
    - A few transports
    - Many link protocols
    - A **glass-shaped architecture**
  - IETF accepts protocols officially if they offer one reference implementation at least

17 Protocols and their Services

Protocols offer services

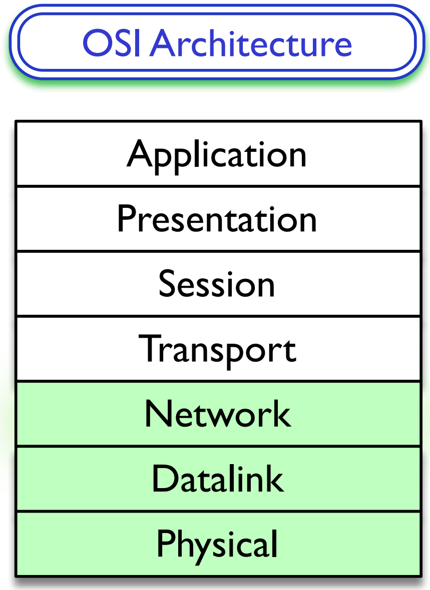
7-layer OSI Architecture

- A Reference Model, actually
- Use of this architecture is limited to some specific protocols
  - IEEE 802.3, 802.1Q, 802.1P, 802.11 (Wifi), 802.15 (WiMax)
- Layering is strict



Description of OSI Layers

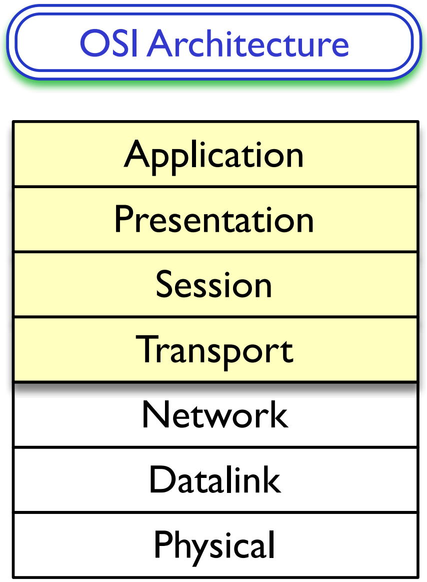
- Physical Layer
  - Handles the transmission of raw bits over a communication link
- Data Link Layer
  - Collects a stream of bits into a larger aggregate called a *frame*
  - Network adaptor along with device driver in OS implement the protocol in this layer
  - Frames are actually delivered to hosts
- Network Layer
  - Handles routing among nodes within a packet-switched network
  - Unit of data exchanged between nodes in this layer is called a *packet*



The lower three layers are implemented **on all network nodes**

Description of OSI Layers

- Transport Layer
  - Implements a process-to-process channel
  - Unit of data exchanges in this layer is called a *message*
- Session Layer
  - Provides a name space that is used to tie together the potentially different transport streams that are part of a single application
- Presentation Layer
  - Concerned about the format of data exchanged between peers
- Application Layer
  - Standardize common type of exchanges

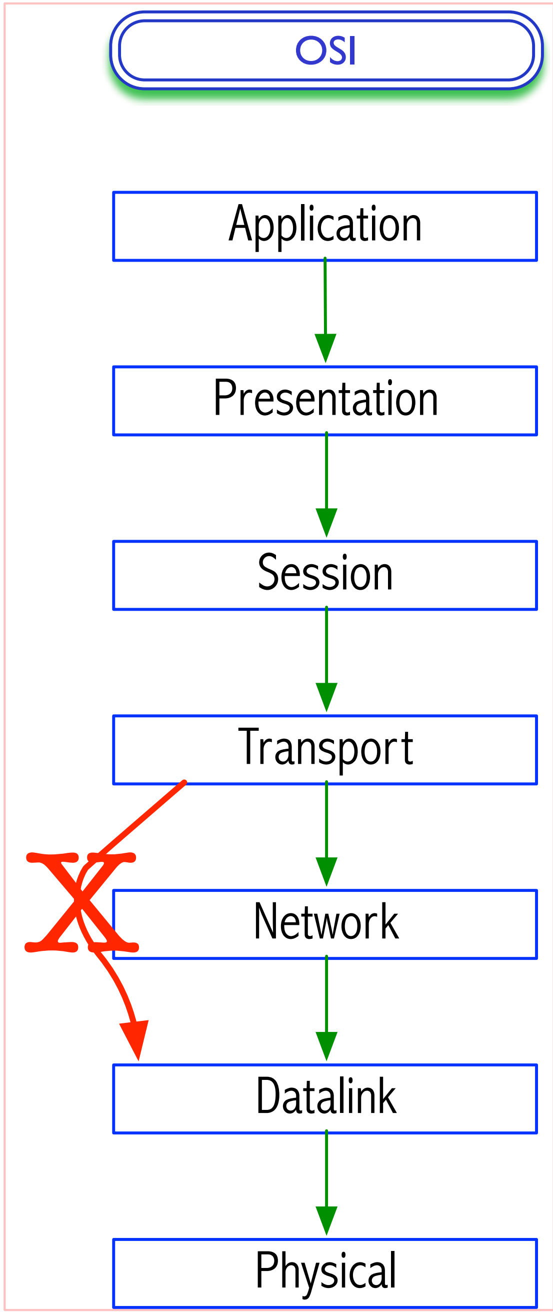


The transport layer and the higher layers typically run only on end-hosts and not on the intermediate switches and routers

OSI, strict layering

A layer only uses the services provided by the layer below

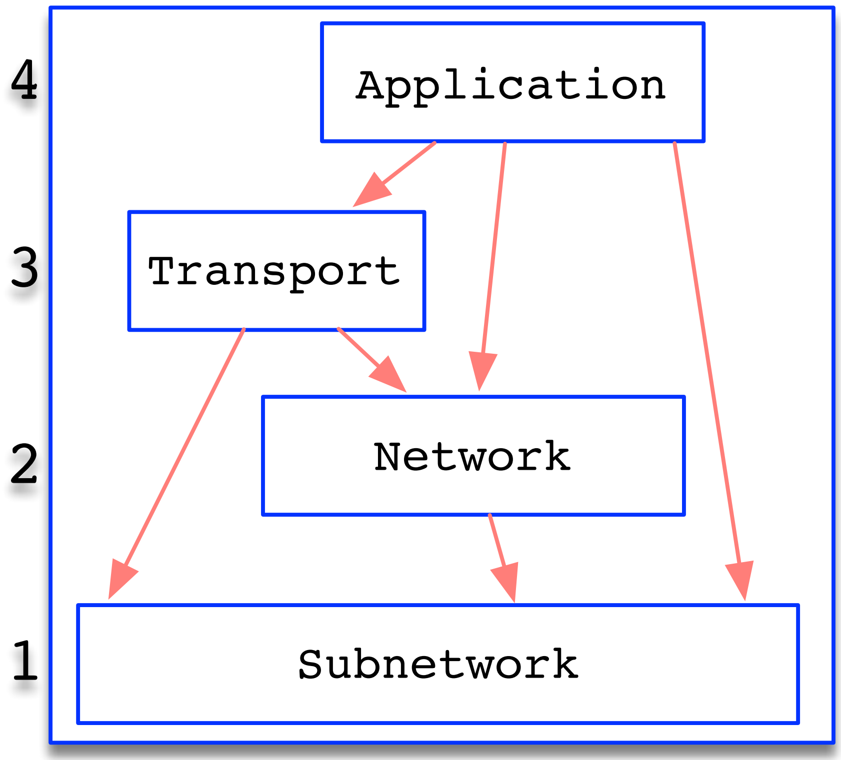
- The internal mechanisms of each layer remain hidden
  - Layer N+1 knows nothing about the internal mechanisms of layer N
- Example:
  - Transport layer can only use the Network layer



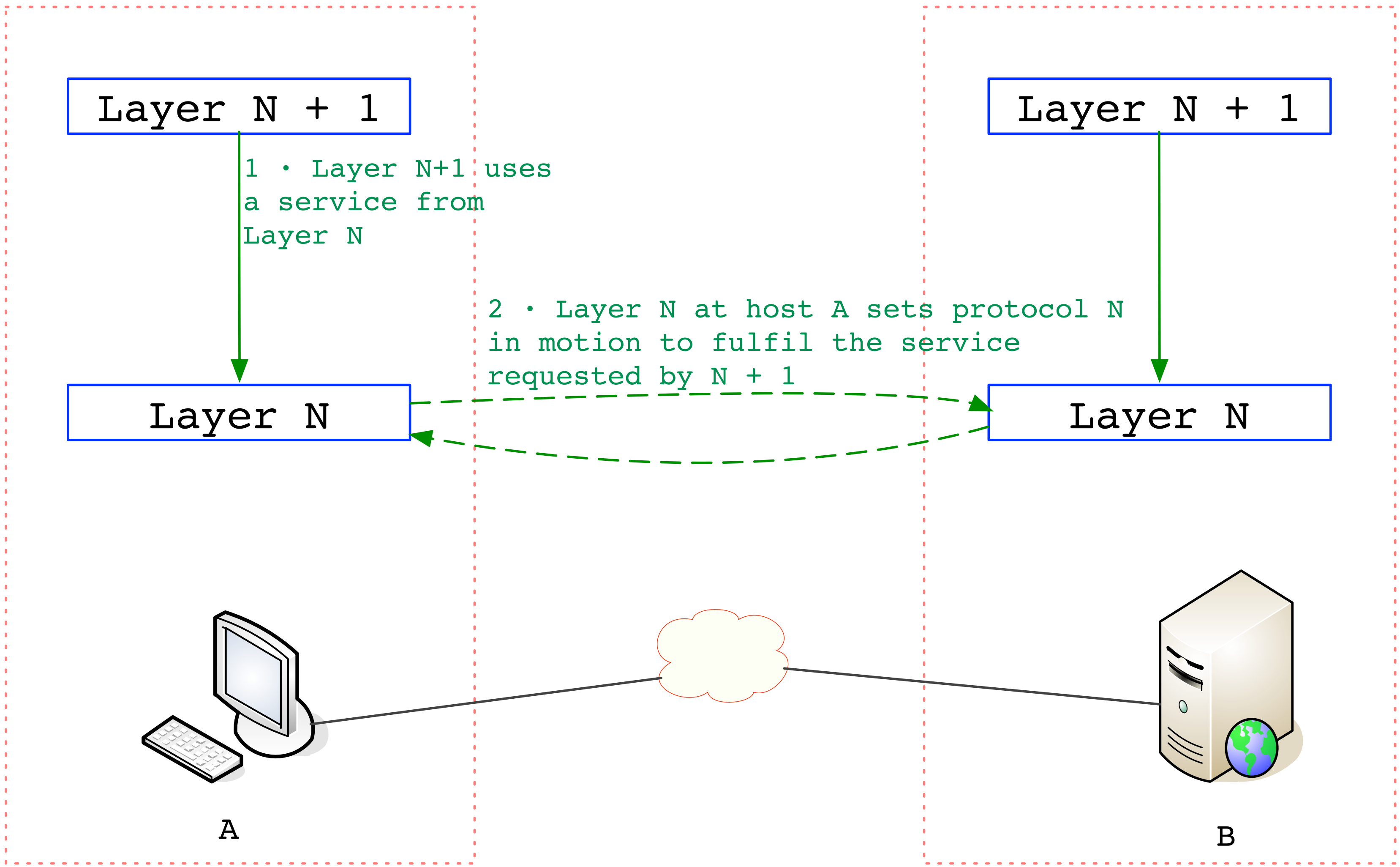
Layering in Internet Architecture is non-strict

A layer may use the services provided by *any* layer below

- The internal mechanisms of each layer remain hidden
  - Layer N+1 knows nothing about the internal mechanisms of layer N
- Example:
  - An Application protocol may use whichever lower layer



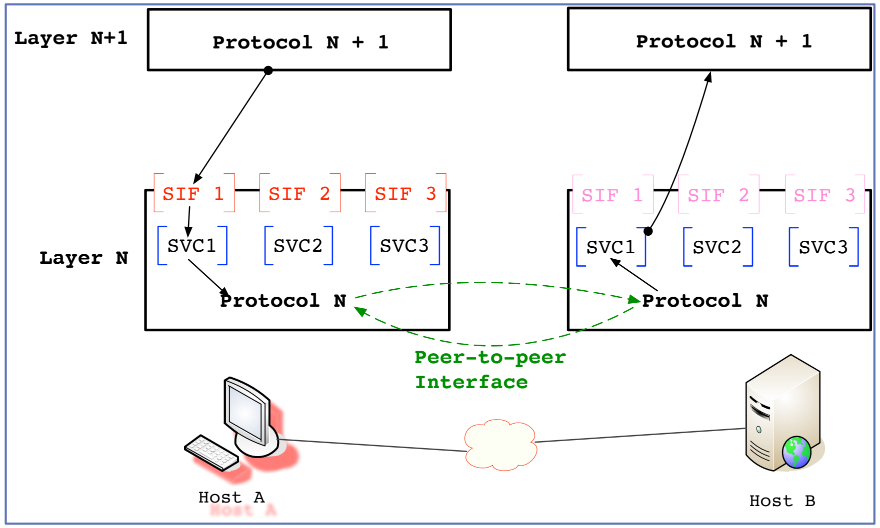
Protocol: The foreman of a service



N+1 represents *any* upper layer in IA (Internet Architecture)

Layer N+1 uses a service at Layer N

- Layer N
  - Several services: SVC1, SVC2
  - Each service is accessed through its **Service Interface:** SIF1, SIF2
  - The protocol N (Host A) fulfils the functionality offered by SVC by exchanging messages with protocol N at Host B
  - These messages comprise the **Peer-to-Peer Interface**

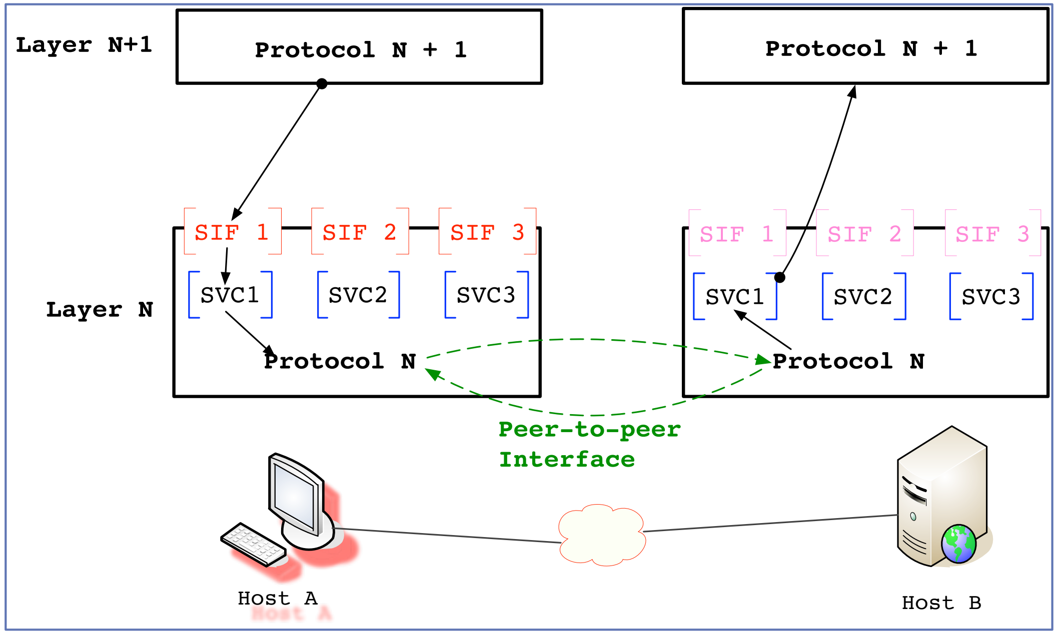


N+1 represents *any* upper layer in IA (Internet Architecture)

Example: A runs Linux; B runs Windows

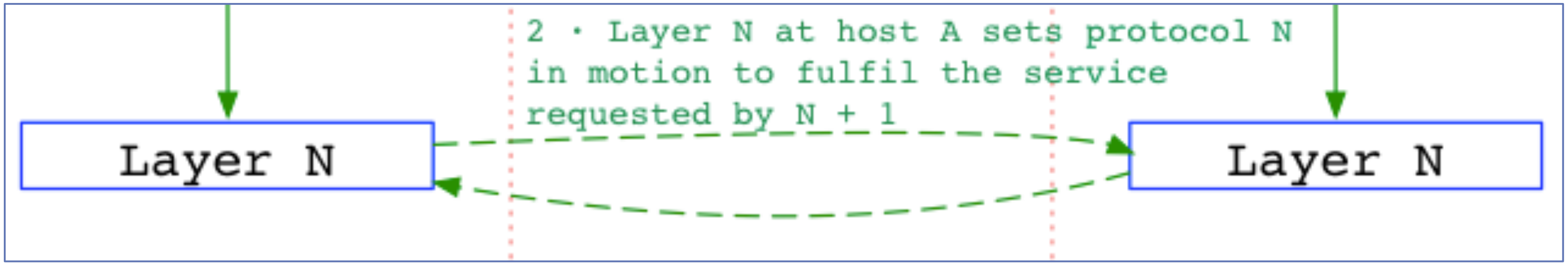
- Equal layers at A and B must implement the **same protocol**
  - *Same peer-to-peer interface*

- However, **Service Interfaces at A and B** might be present differences



Peer-to-peer interface

- The syntax and the semantics of the messages exchanged by the two peers must follow a formal specification
  - ASN.1, Abstract Syntax Notation
- Normally, we refer to the peer-to-peer interface with the same word: protocol
- Protocols of Internet are specified by the IETF
  - RFC: Request For Comments
  - Example: The ICMP protocol is specified in RFC 792



Encapsulation and Multiplexing: *Concept*

- What information is sent from N+1 to N through the SIF (Service Interface)?
  - Protocol N+1 sends a N+1 Data Unit to Protocol N
  - Protocol N encapsulates the N+1 Data Unit into a fresh N Data Unit:
    - Payload(N+1) + Header(N)
    - This scheme is reproduced at each service use
  - **Data Unit**: A bit string produced by a protocol
  - **Encapsulation**: Appending a Header to a Data Unit

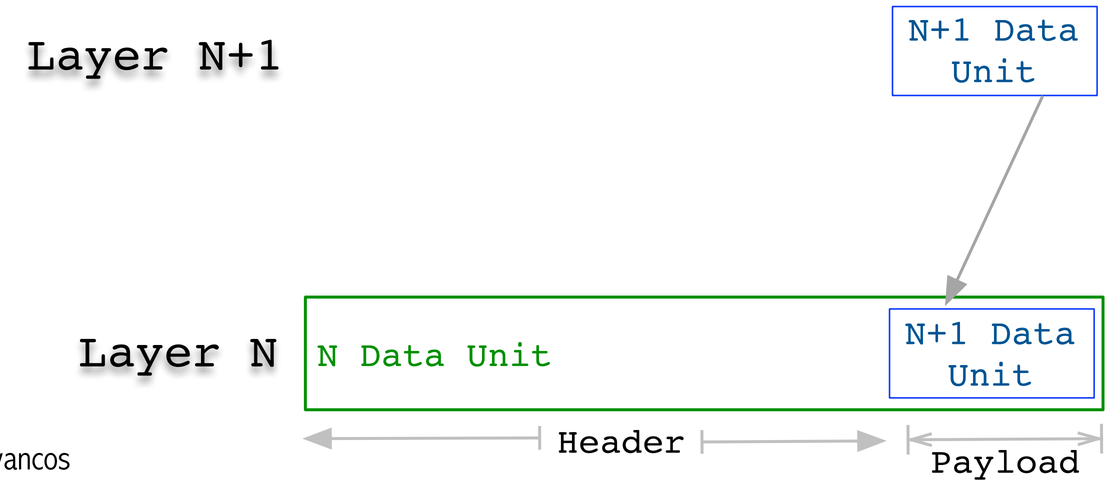
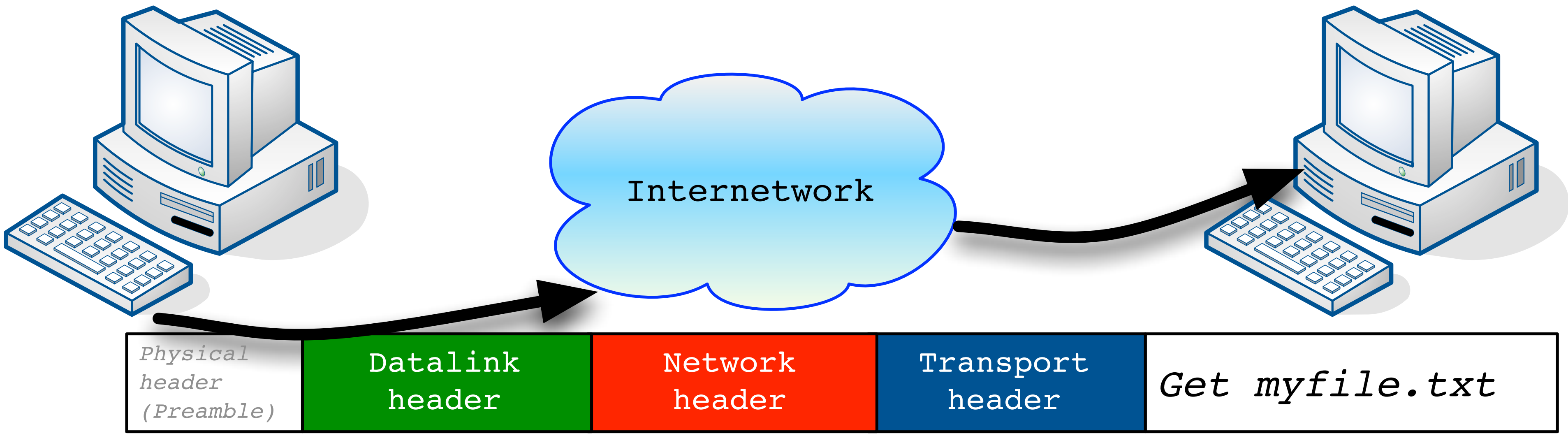
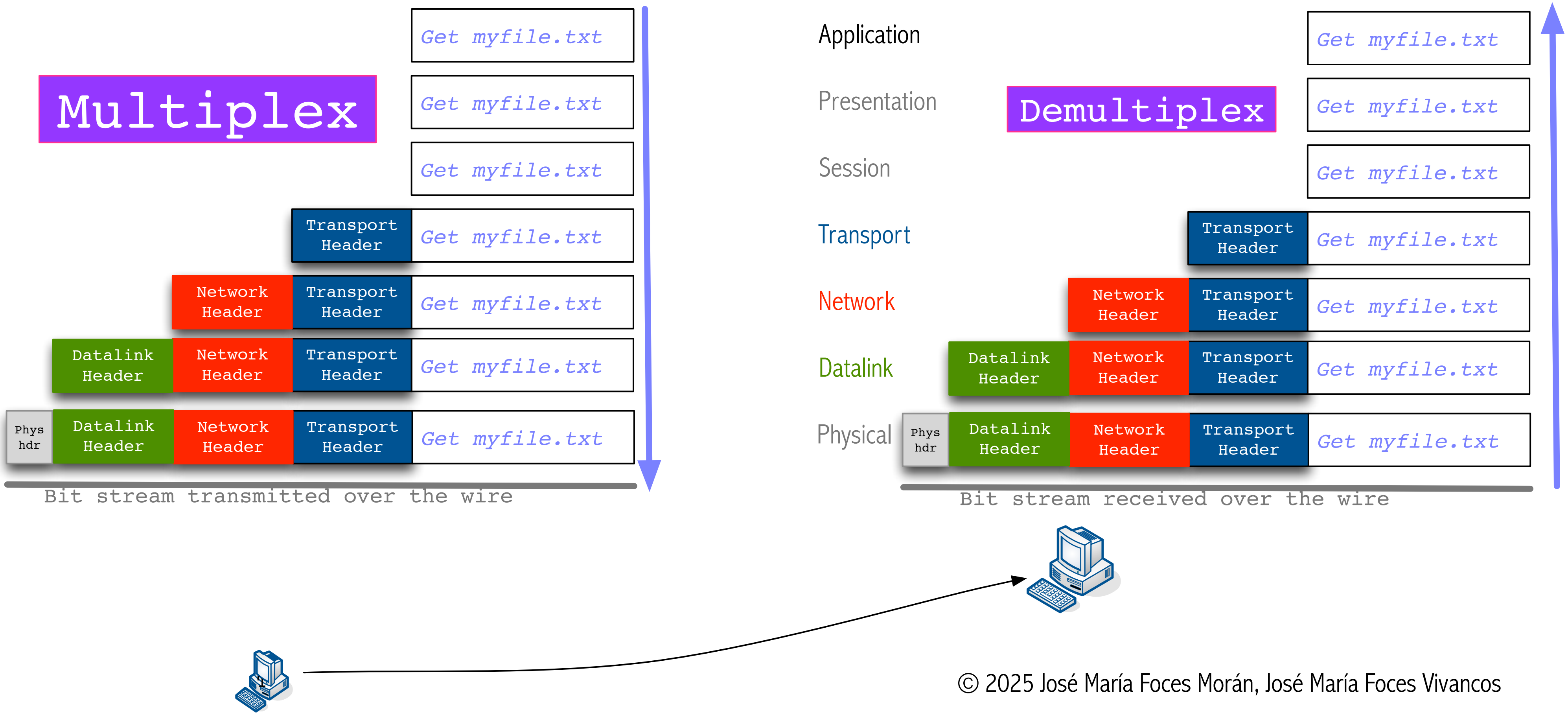


Illustration of encapsulation in OSI



Multiplexing

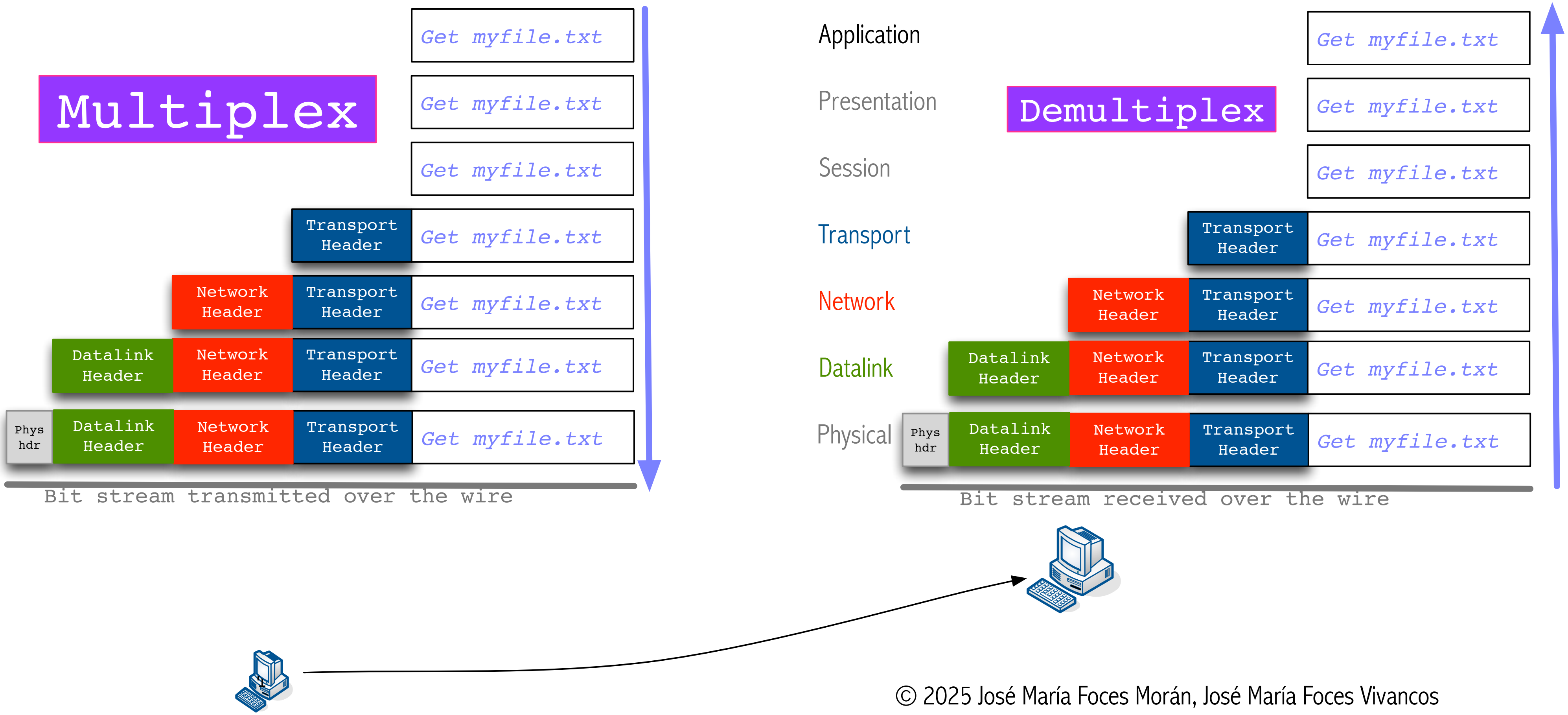
- Transmitter multiplexes several flows by having each layer add its header which contains addressing information



Demultiplexing

30

- Receiver demultiplexes several flows by having each layer analyze its header which contains addressing information about the upper-layer protocol that is to receive the payload



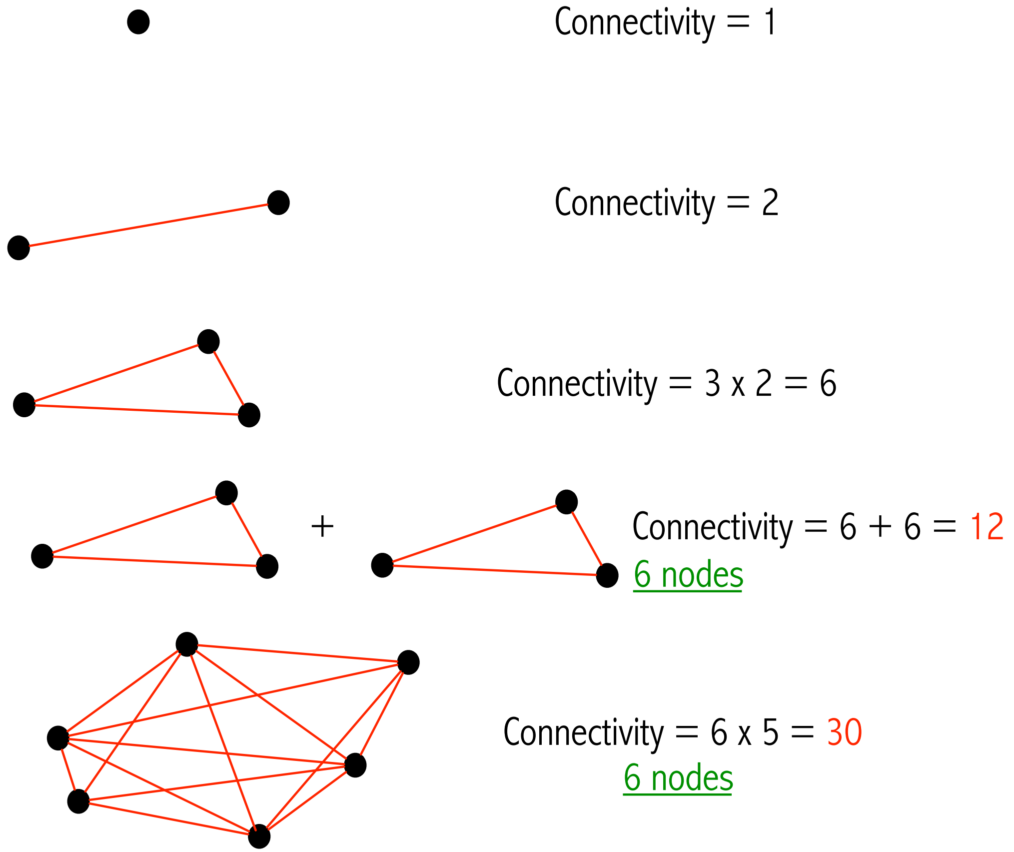
31

Connectivity

Computer Networks connect computers; the many more, the better, with a limit!

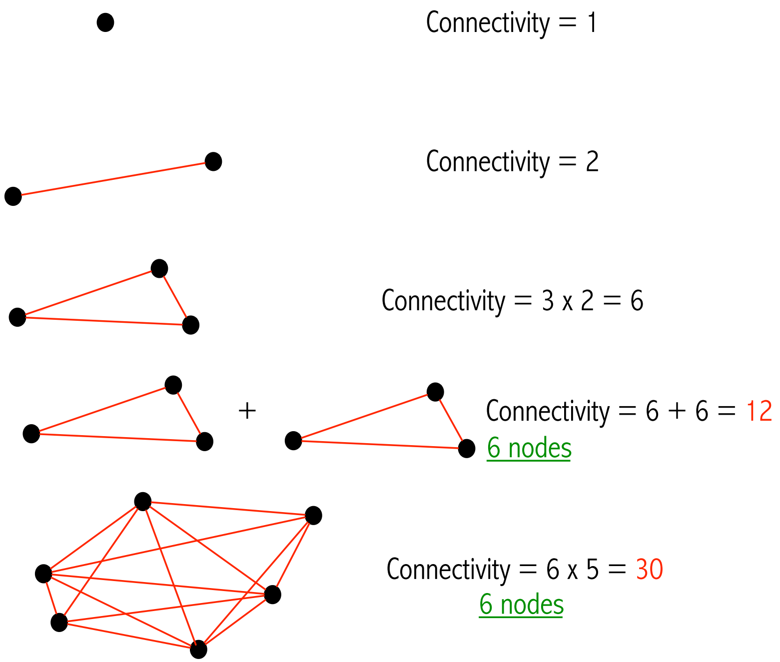
Theoretical connectivity

- Connectivity is the capacity of connection of a network
- If a network has N hosts, its connectivity is: N·(N-1) ≅ **N^2**
- Metcalf's law: The connectivity of a network grows fast as we add more nodes (**N^2**)



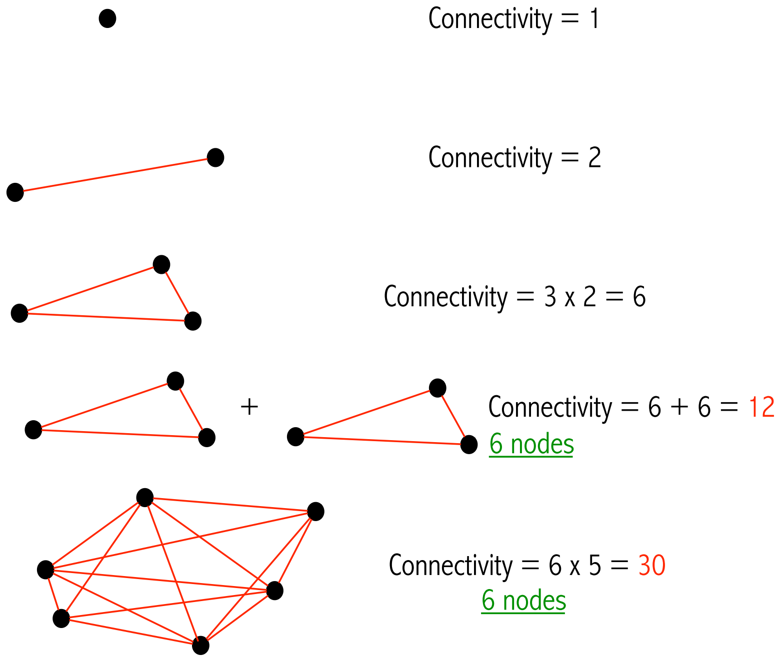
Increasing connectivity whilst preserving the capability for communication

- Metcalf's law
  - Increased connectivity means increased value
- Nodes communicate by sending/receiving messages
  - The bandwidth available at each link is limited
  - Links at highly demanded locations may become a bottleneck
- What's a figure of merit that will tell whether communication has been preserved after increasing the connectivity?
  - Is connectivity scalable?



Increasing connectivity whilst preserving the capability for communication

- The total number of packets per second that successfully make it to their destinations is known as:
  - Throughput, the figure of merit
  - Overall network productivity
  - Overall bps, or pps (packets per second), etc.



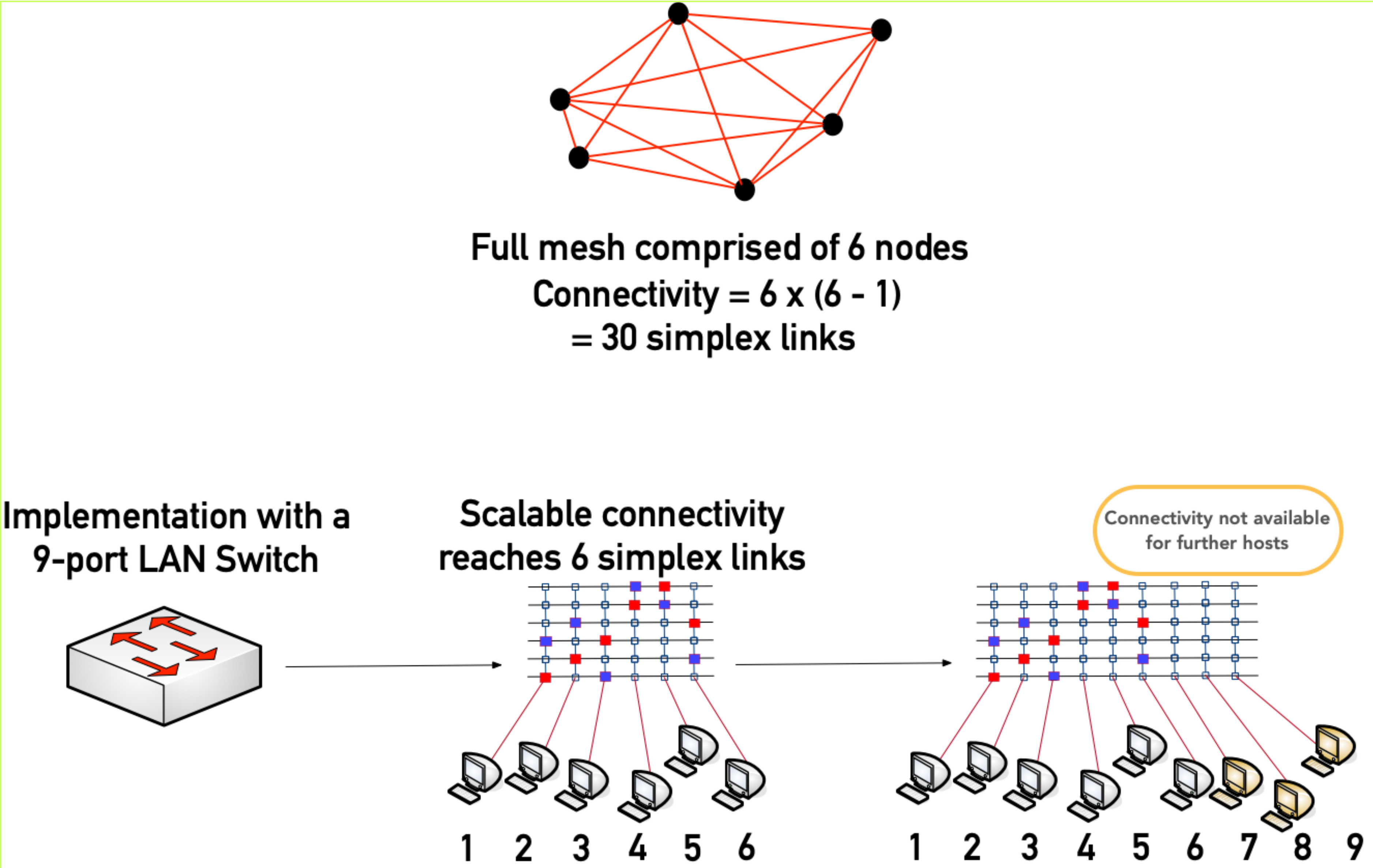
Scalable connectivity

- Not all network technologies use the available connectivity with the same efficiency

- Ethernet can function efficiently up to certain network size: we say that Ethernet scales well up to that limit.

- Then, how come the Internet has **4000M** hosts? How can the Internet scale to such a huge size so well?
  - Each network has a limited size
  - Interconnecting networks is the key:
    - With IP gateways
    - IP protocol

Switch won't scale to 9 hosts



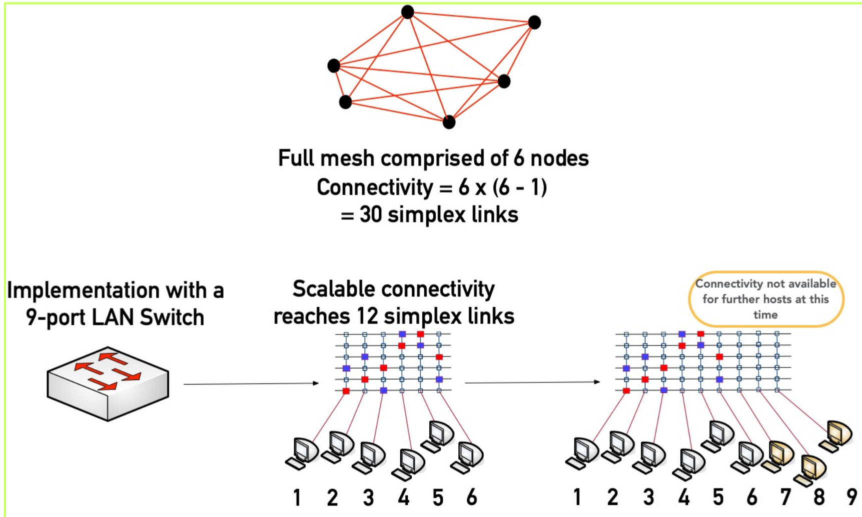
Based on textbook *Conceptual Computer Networks* by:

Switch won’t scale to 9 hosts

6 full-duplex communication flows are possible

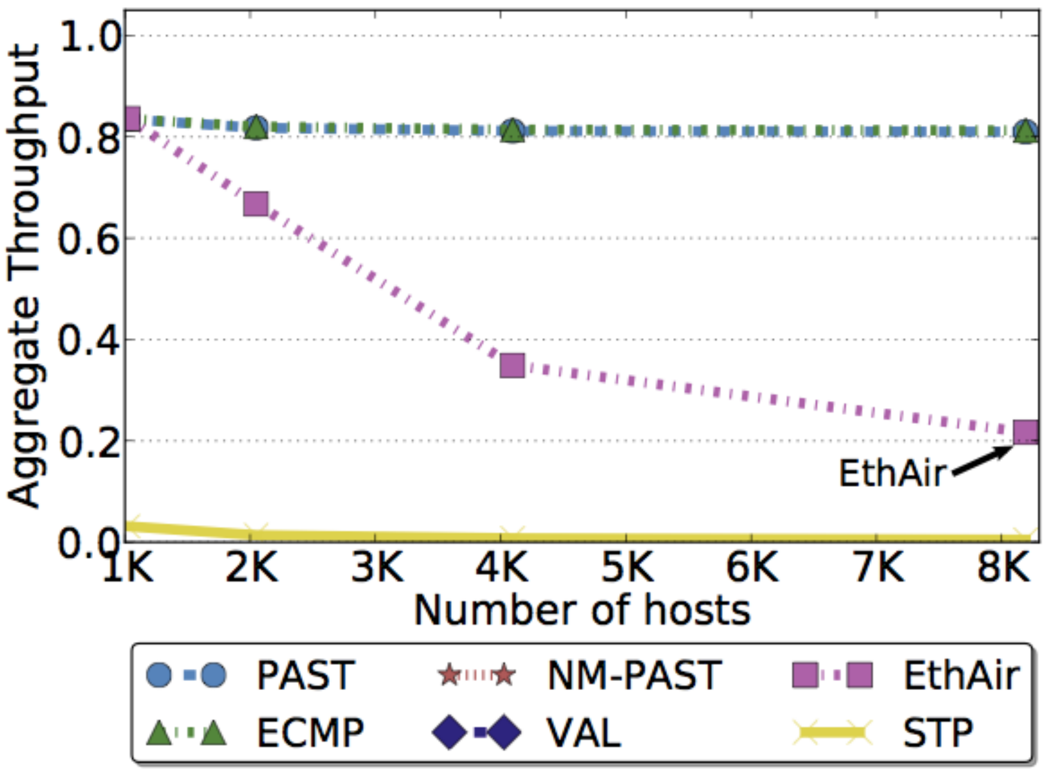
- The overall productivity of this switch will be bounded by
  - The available number of connection points
  - The available number of horizontal lines
- Throughput, the total pps or bps will be bounded by the limited switch resources
  - PPS = Average number of Packets Per Second that the switch can successfully deliver
  - Bps = Average number of bits per second that the switch can successfully deliver

Physically connecting more than 6 hosts will not achieve a Throughput improvement



Example: Network Throughput of various wireless technologies

EthAir scales poorly



Example: Response time (s) of a multi-server application

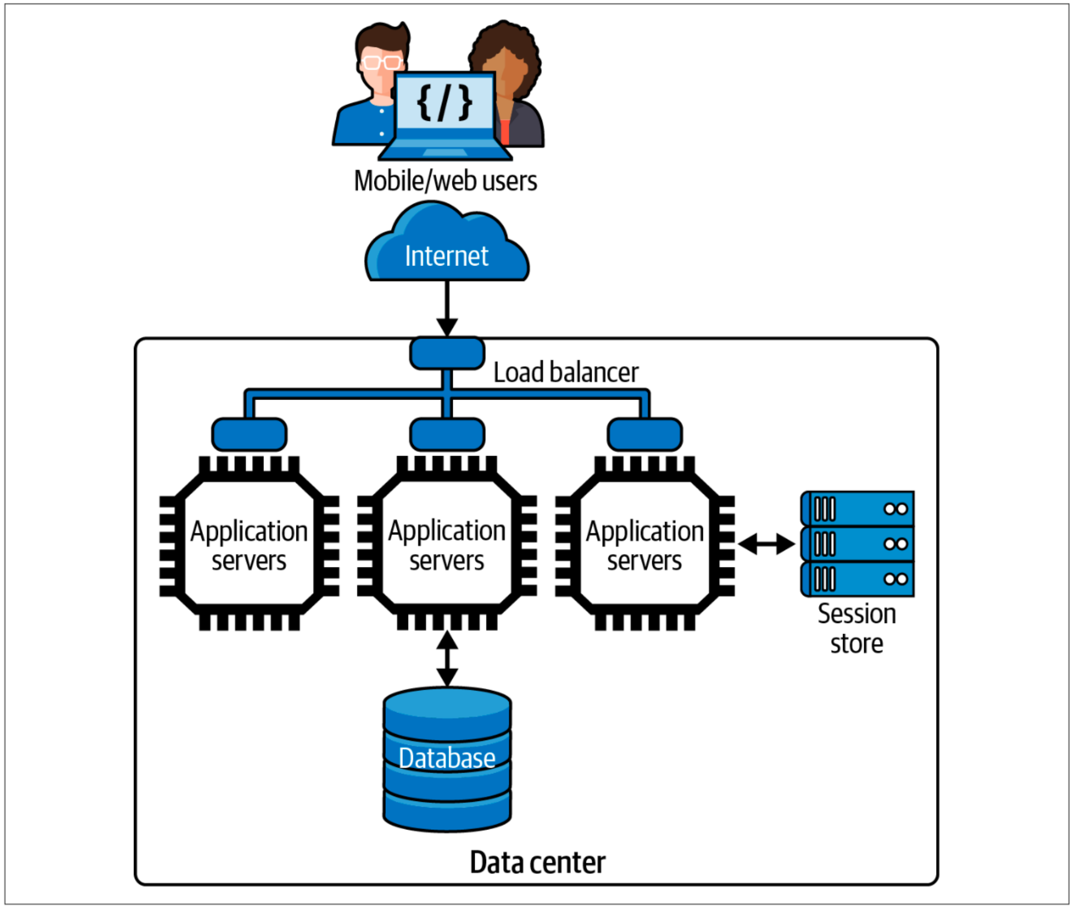


Figure 2-2. Scale-out architecture

Foundations of Scalable Systems
by Ian Gorton

Printed in the United States of America.

Published by O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472.

Example: Mean response time (s) of an application vs. offered load (Requests/s)

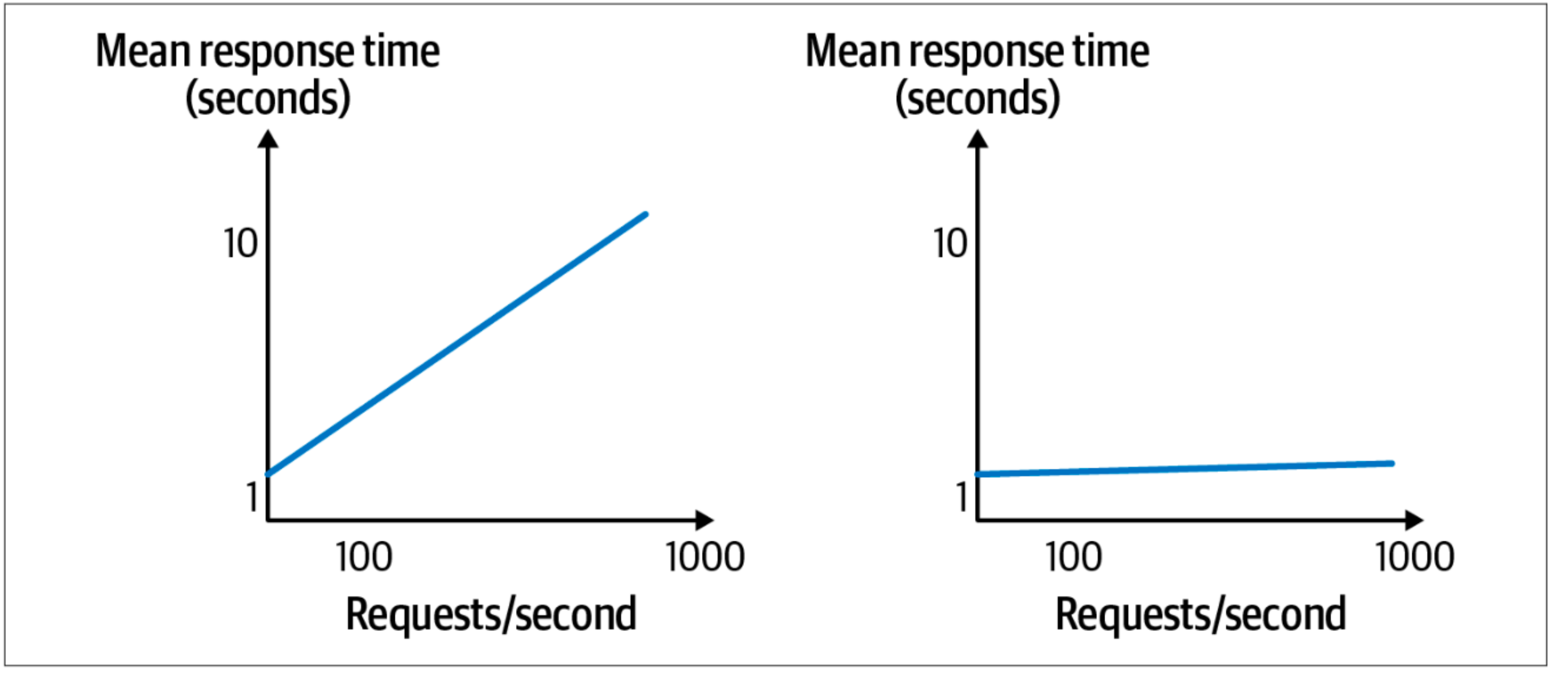


Figure 1-2. Scaling an application; non-scalable performance is represented on the left, and scalable performance on the right

Foundations of Scalable Systems
by Ian Gorton

Printed in the United States of America.

Published by O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472.

Examples (1/2)

- Mesh connectivity of a 5-node mesh network

- Draw the 5-node mesh

- The throughput of a network is 10^6 packets/s. Average packet size is 30 Bytes. Calculate the average network throughput in bps

Examples (2/2): transmission speed of a WIFI

intel. Products Support Solutions Developers Partners Foundry

Support Knowledge Base

Different Wi-Fi Protocols And Data Rates

Content Type: Product Information & Documentation | Article ID: 000005725 | Last Reviewed: 10/28/2021

Click ⌄ or the topic for details:

▼ **IEEE 802.11 Wi-Fi protocol summary**

Protocol	Frequency	Channel Width	MIMO	Maximum data rate (theoretical)
802.11ax	2.4 or 5GHz	20, 40, 80, 160MHz	Multi User (MU-MIMO)	2.4 Gbps[1]
802.11ac wave2	5 GHz	20, 40, 80, 160MHz	Multi User (MU-MIMO)	1.73 Gbps[2]
802.11ac wave1	5 GHz	20, 40, 80MHz	Single User (SU-MIMO)	866.7 Mbps[2]
802.11n	2.4 or 5 GHz	20, 40MHz	Single User (SU-MIMO)	450 Mbps[3]
802.11g	2.4 GHz	20 MHz	N/A	54 Mbps
802.11a	5 GHz	20 MHz	N/A	54 Mbps
802.11b	2.4 GHz	20 MHz	N/A	11 Mbps
Legacy 802.11	2.4 GHz	20 MHz	N/A	2 Mbps

[1] 2 Spatial streams with 1024-QAM modulation.
[2] 2 Spatial streams with 256-QAM modulation.
[3] 3 Spatial streams with 64-QAM modulation.

Feedback

End of Ch 1 Section 1