Course Code: CSE3003

Course Title: Computer Networks

R.Nandha kumar

Syllabus

Module No. 1	**Introduction**	**6 Hours**
Computer network and its history, progress and application, Internet, Network architecture, Networking devices. OSI Model, TCP/IP Protocol stack, Networking in different OS.		
Module No. 2	**Physical Layer**	**8 Hours**
Data communication technologies, Analog and digital communication. Encoding mechanisms, Packet Switching, Circuit Switching.		
Module No. 3	**Data Link Layer**	**8 Hours**
Framing, HDLC, PPP, Error detection, Error Correction, MAC Protocols, Reliable Transmission, Ethernet, 802.3, 802.5, 802.11, PPP,ATM.		
Module No. 4	**Network Layer**	**7 Hours**
IP addressing schemes, IPV4, Subnetting, IPV6, shift from IPV4 to IPV6, ICMP, DHCP, ARP. Routing Protocols: Distance-vector and link-state routing. RIP, OSPF, BGP Multicasting.		
Module No. 5	**Transport Layer**	**8 Hours**
Connection Oriented and connection less service, TCP and UDP, Port Addressing, Remote Proced Call, Flow Control vs Congestion Control, Quality of Service.		
Module No. 6	**Application Layer Protocols**	**8 Hours**
Application Layer Protocols: World wide web and HTTP, HTTPS, Domain names: DNS, File Transfer: FTP, Electronic mail: SMTP, Peer to peer networking, Torrent, VPN, Session managemen Data compression techniques.		

CAT - I

Introduction

Computer network and its history progress and application
Internet
Network architecture
Networking devices
OSI Model
TCP/IP Protocol stack
Networking in different OS.

Physical Layer

Data communication technologies
Analog and digital communication
Encoding mechanisms
Packet Switching
Circuit Switching

Data Link Layer

Framing
HDLC
PPP
Error detection
Error Correction

Computer networking progress and application

- Network technology
- It is a coordinated set of software and hardware(for example, drivers, network adapters, cables and connectors)
- Mechanism

Network Architecture

- In general, architecture is the art and science of designing and constructing, or the discipline dealing with the principles of design and building .
- Network architecture Includes
  - Addressing and routing,
  - Network management,
  - Performance, and security.
- Understanding of the relationships between (architectural) components of the network.

Network Architecture

- The physical entities (routers, switches, multiplexers, servers, etc.) in the network, or
- Functional entities instead of physical entities.
  - The set of high-level design principles that constitute the network architecture is applied to
    - How the network functions and operates.
  - Network functions are closely coupled to users, their applications, and their devices.
  - This allows user requirements to be directly represented in the network architecture.
  - Measures the network success,
    - How well user, application, and device requirements are supported through these functions.

Component Architectures

Function	Description of Capability	Example Subset of Mechanisms Used to Achieve Capability
Addressing/Routing	Provides robust and flexible connectivity between devices	• Addressing: Ways to allocate and aggregate address space • Routing: Routers, routing protocols, ways to manipulate routing flows
Network Management	Provides monitoring, configuring, and troubleshooting for the network	• Network management protocols • Network management devices • Ways to configure network management in the network
Performance	Provides network resources to support requirements for capacity, delay, RMA	• Quality of Service • Service-Level Agreements • Policies
Security	Restricts unauthorized access, usage, and visibility within network to reduce the threat and effects of attacks	• Firewalls • Security policies and procedures • Filters and access control lists

FIGURE 5.3 Functions, Capabilities, and Mechanisms

Contents

Protocols

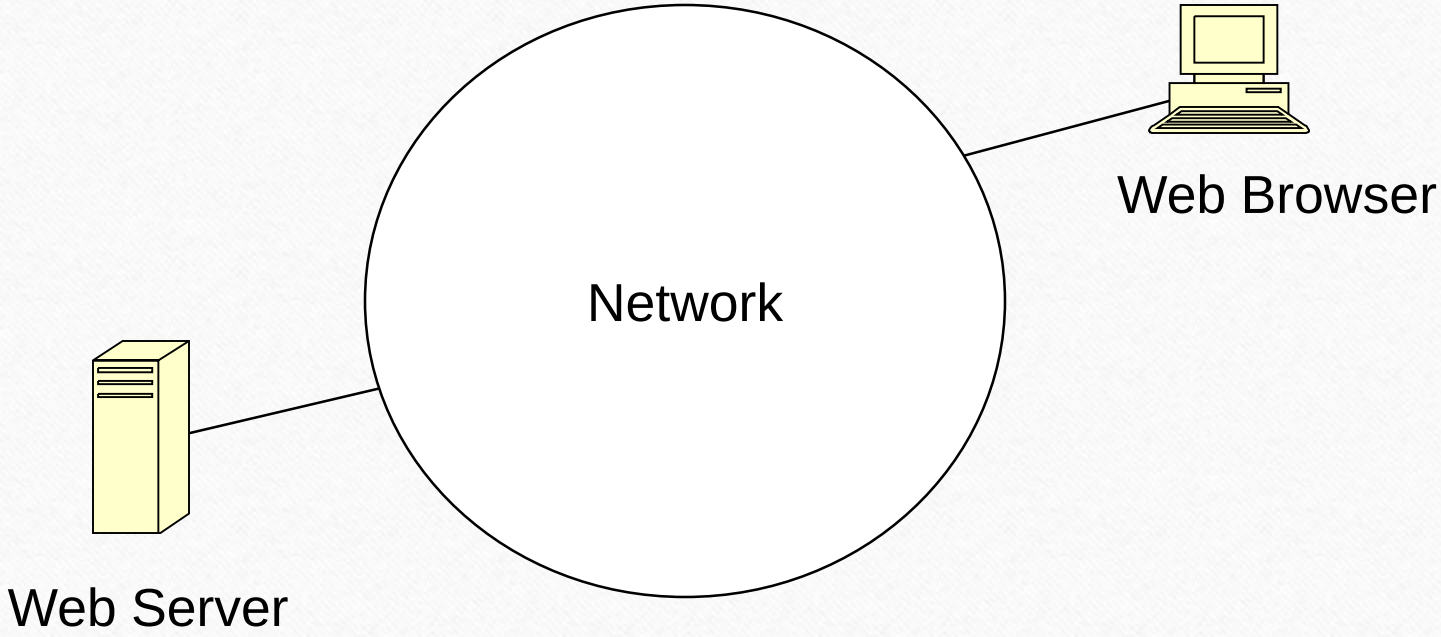
- The term protocol refers to a well-known set of rules and formats to be used in order to perform a task. For example, a task of communicating between processes.
- Parts of a protocol:
  - A specification of a sequence of messages that must be exchanged.
  - A specification of the format of the data in the messages.
- Existence of well-known (standard) protocols enables the separate components of the distributed systems to be developed independently in different languages and on different platforms.

Four elements of a protocol:

A set of rules governing the communication between two peer entities. It must define the format and the order of messages as well as actions taken on the transmission and receipt of a message.

- syntax: format, what is a valid message?
  - "GET /~hugue/index.html HTTP/1.1\nHOST: www.cs.umd.edu\n\n"
- Semantics: what does it mean?
  - Get file /~hugue/index.html using the http 1.1 protocol.
- Action:
  - read file /~hugue/index.html from the disk, send it through the socket using the http 1.1 protocol and close the socket
- Timing: relative order of messages.
  - Reply follows the request

1.1 Internet Protocols



Four-Layer Model

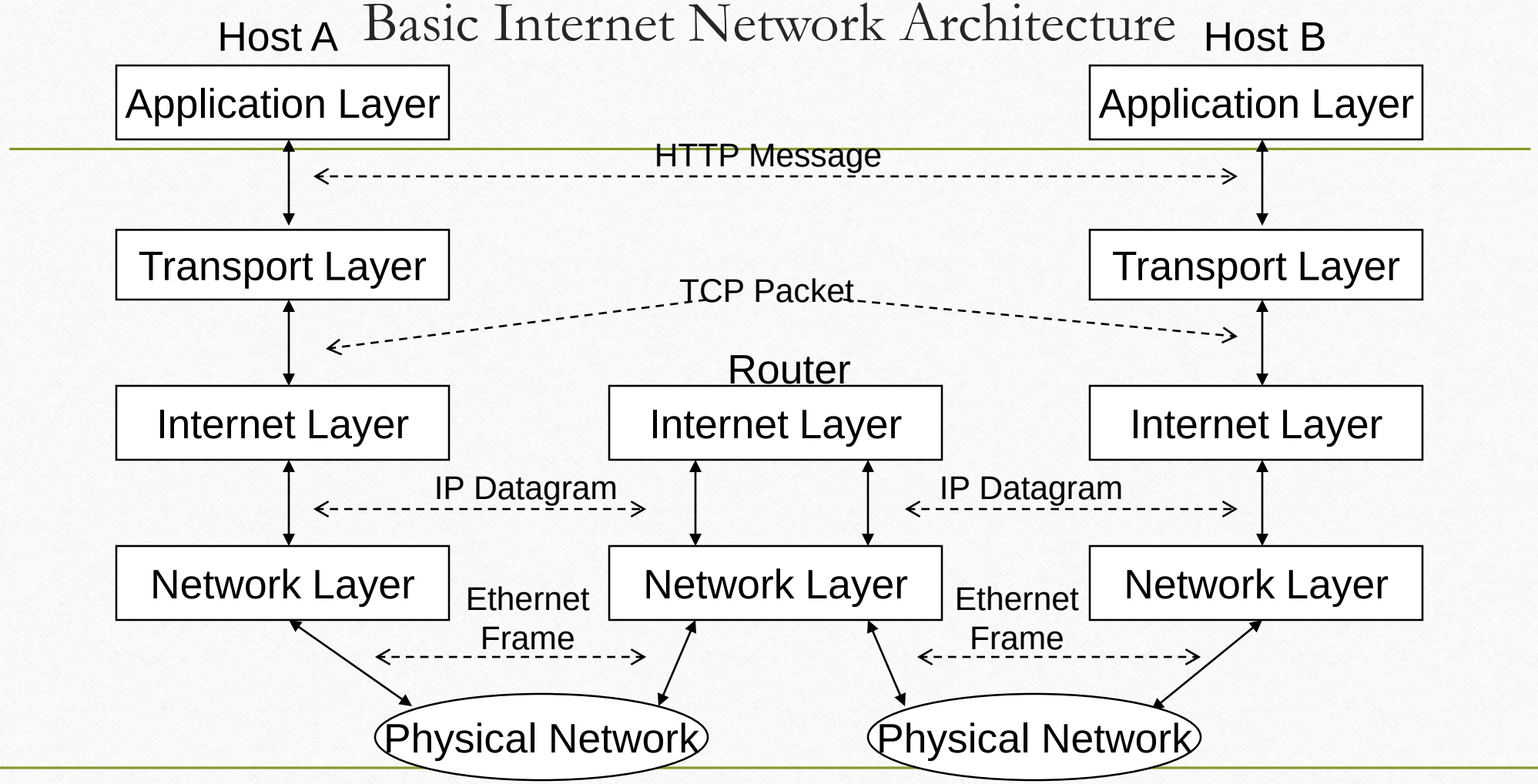
Distributed data communications involves three primary components:

- Applications
- Computers
- Networks

Four corresponding layers

- Application layer
- Transport layer
- Internet layer
- Network Interface

Basic Internet Network Architecture



Application Layer

- How does a web browser retrieve data from a web server?
- Application Protocol: Hypertext Transfer Protocol (HTTP).
- Users invoke applications which “speak” using application protocol.
- Applications interact with a transport protocol to send or receive data.
- Other applications: FTP, SMTP, DNS, SMB, …

Application Layer Example

- HTTP outline:
  - GET /~hugue/index.html HTTP/1.1
  - Host: www.cse.vit.edu

GET /~hugue/index.html HTTP/1.1
Host: www.cse.vit.edu

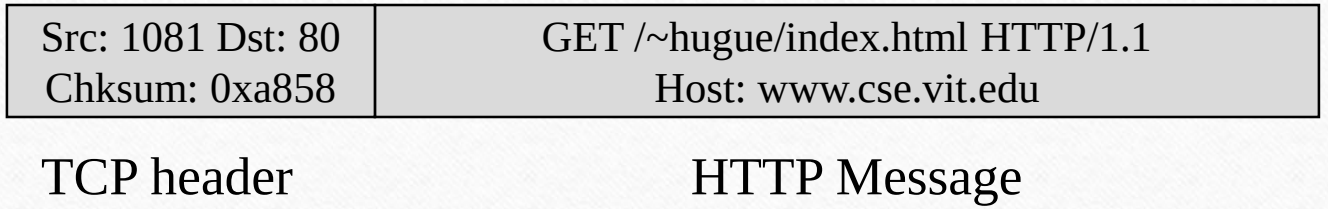
HTTP Message

Transport Layer

- Provides end-to-end communication between applications.
- Transport Protocol: Transport Control Protocol (TCP)
  - a *reliable*, *connection-oriented* transport protocol.
- Divides stream of application messages into packets.
- Interacts with Internet Layer to send or receive data.
- In general, a transport protocol may be
  - reliable or unreliable,
  - connection-oriented or connectionless,
  - and flow may or may not be regulated.
- Others: UDP, ICMP.

Transport Layer Example

- TCP outline:
  - Source Port: 1081
  - Destination Port: 80
  - Checksum: 0xa858

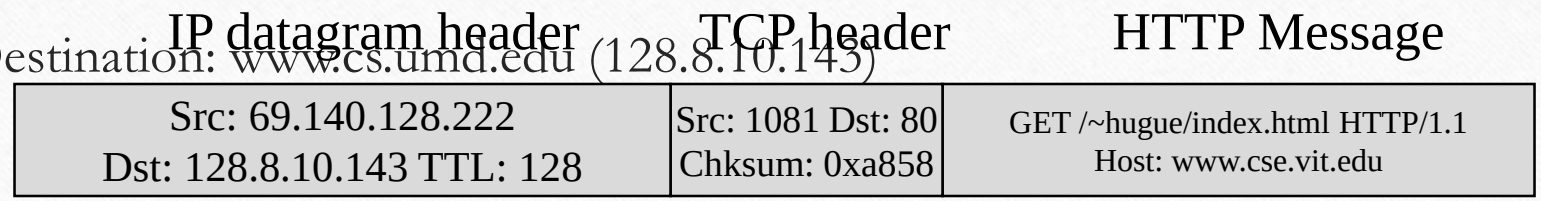


Internet Layer

- Responsible for routing communications between one machine and another.
- Accepts requests to send packets to destination address.
- Internet Protocol (IP) encapsulates packets in IP datagram with IP header and uses routing algorithm to decide whether to send directly or indirectly.
- Also handles incoming IP datagrams.
  - If addressed to local machine, remove the IP datagram header and pass up to transport layer.

Internet Layer Example

- IP outline:
  - Time to live: 128
  - Header checksum: 0x57d1
  - Source: my home pc (69.140.128.222)
  - Destination: www.cs.umd.edu (128.8.10.143)

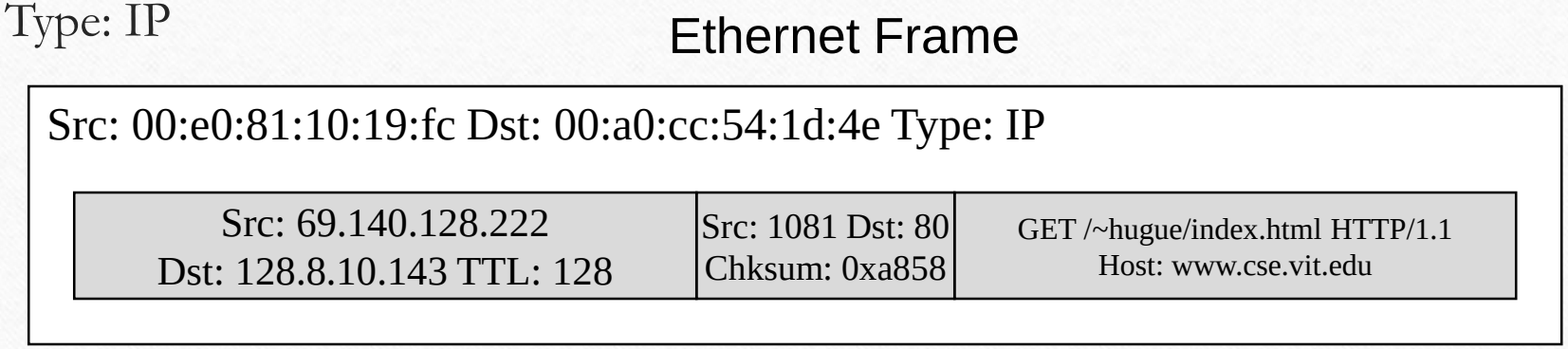


Network Interface Layer

- Accepts IP datagrams and transmits over specific networks.
- Maybe a simple device driver (e.g. an Ethernet driver) or a complex subsystem with further data link protocols.

Network Interface Layer Example

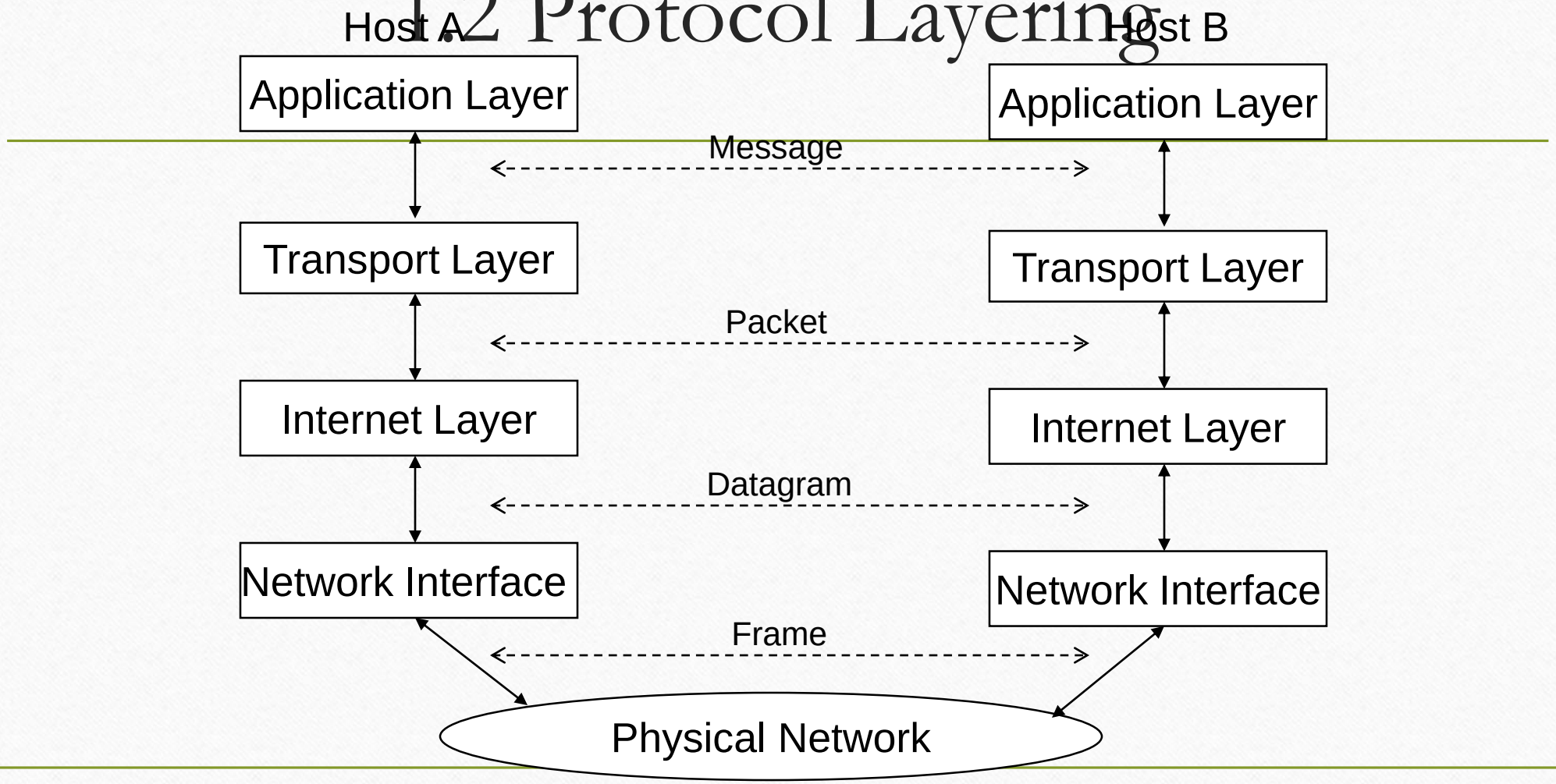
- Ethernet outline:
  - Destination: 00:a0:cc:54:1d:4e
  - Source: 00:e0:81:10:19:fc
  - Type: IP



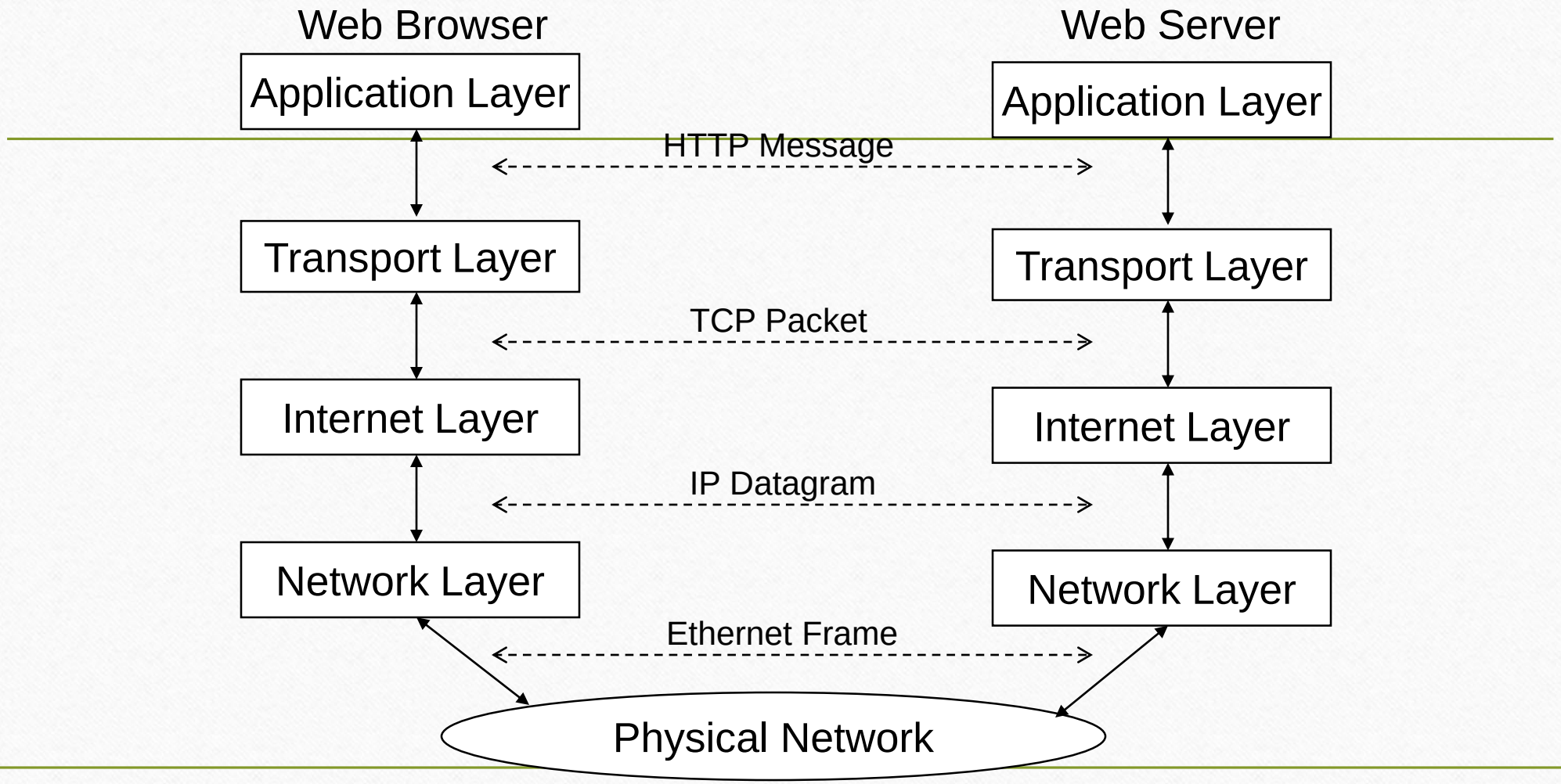
Ports and Addresses

- Ports are destination points within a host computer.
- Processes are attached to the ports, enabling them to communicate.
- Transport layer addresses are composed of network address of the host computer and a port number.
- In the Internet every host is assigned a unique IP number which is used in routing.
- In an Ethernet each host is responsible for recognizing that the messages meant for it.

1.2 Protocol Layering



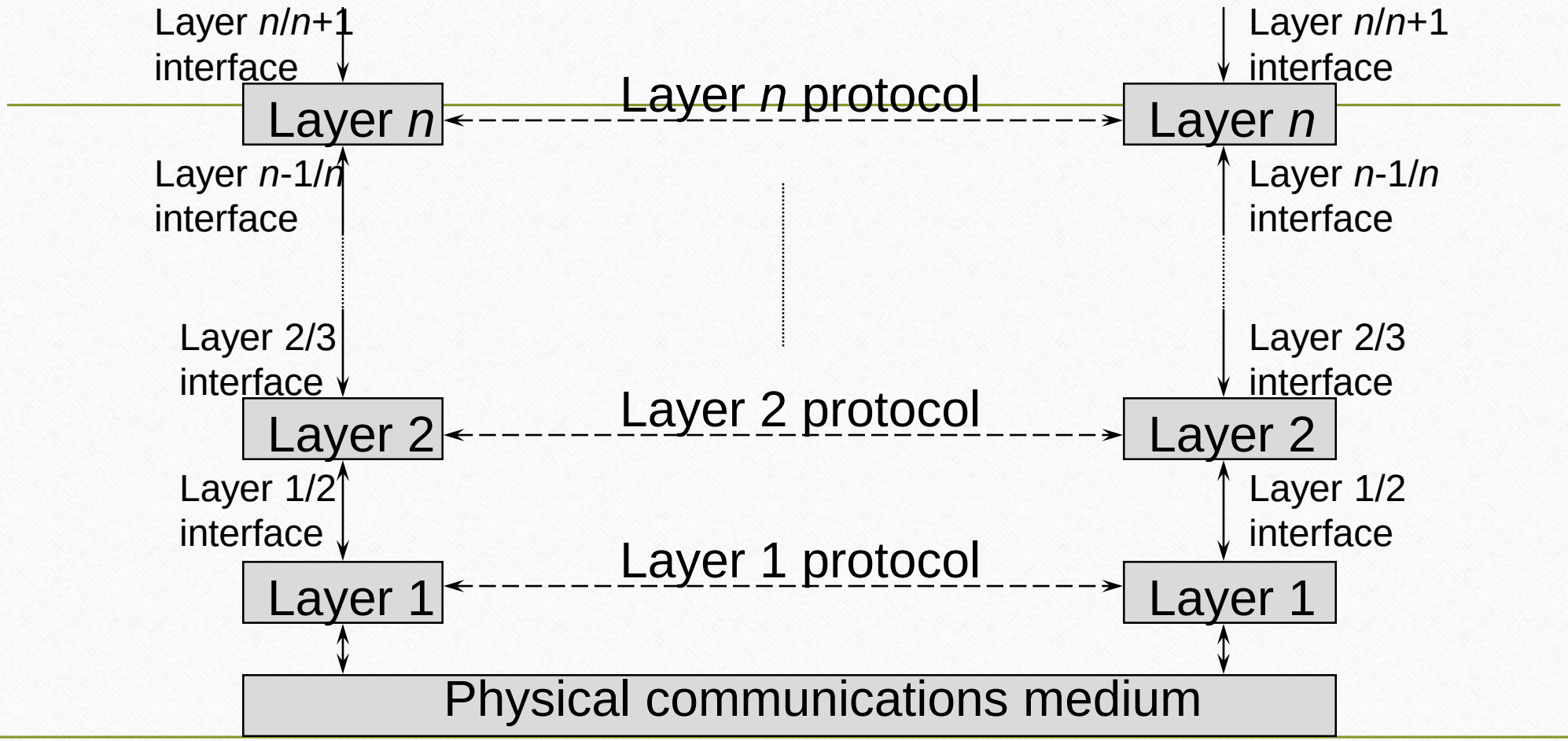
Protocol Layering



Protocol Hierarchies

- Protocols are stacked vertically as series of 'layers'.
- Each layer offers *services* to layer above, shielding implementation details.
- Layer n on one machine communicates with layer n on another machine (they are *peer processes/entities*) using Layer n Protocol.

Layers, Protocols & Interfaces



Layer/Interface Design

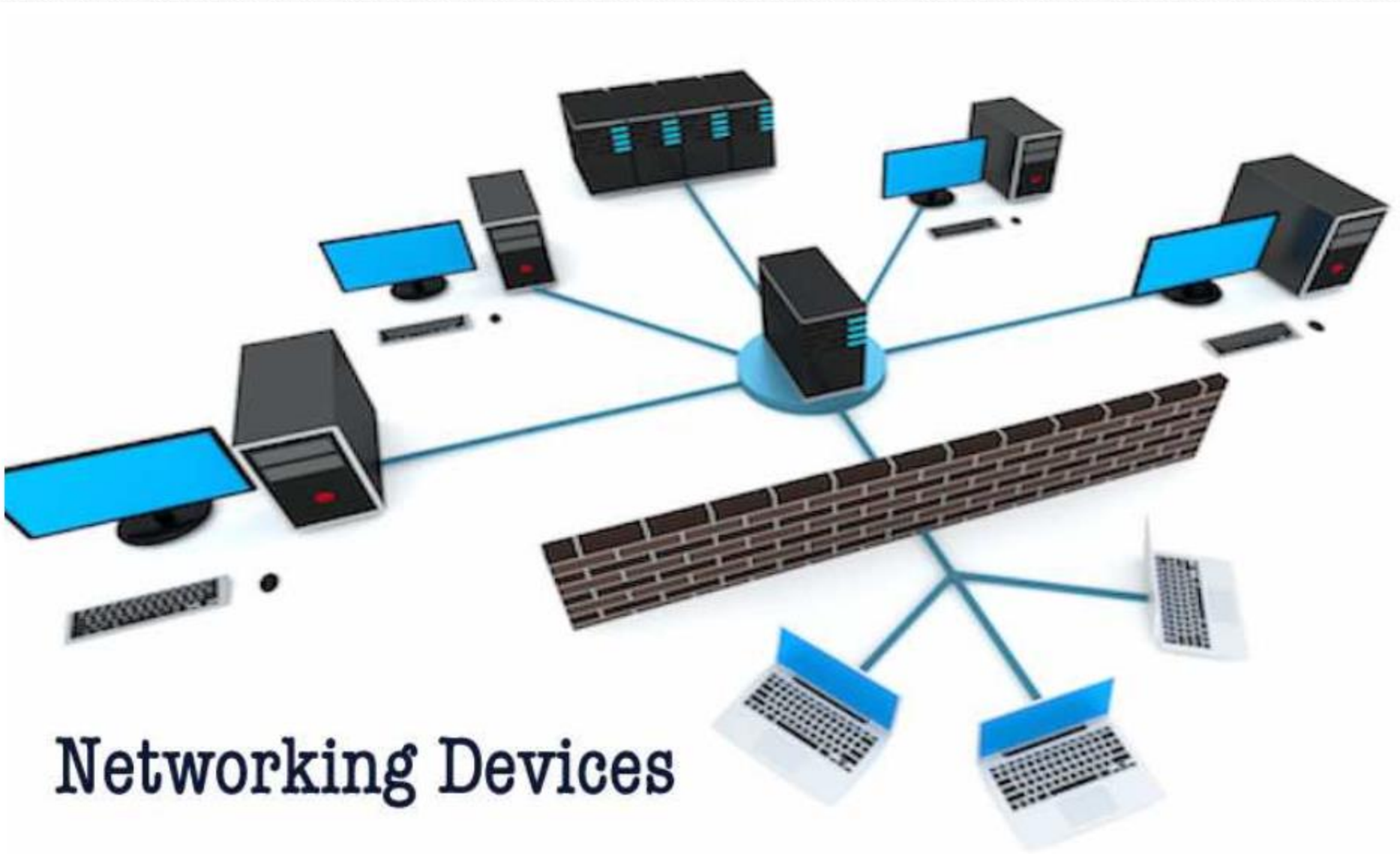
- Important objective is ‘clean’ interfaces, having minimal set of well-defined services.
- Clean-cut interfaces enable:
  - minimisation of inter-layer communications
  - easy replacement of individual layers
- Set of layers and protocols is the Network Architecture.

Virtual & Actual Communications

- Important to understand difference between:
  - virtual and actual communications,
  - protocols and interfaces.
- Peer processes ‘think’ of communications as being ‘horizontal’ using protocol.
- Actual communications is via interfaces (and the physical communications medium).
- Peer process idea is key to network design.

Design Issues

- Some issues affect many layers, e.g:
  - need to address data (say who it's for),
  - possible need for setting up connections,
  - data transfer rules (simplex, half-duplex, ...),
  - error management,
  - deal with message component re-ordering,
  - flow control,
  - routing
  - security


Networking Devices