Intro to computer networks

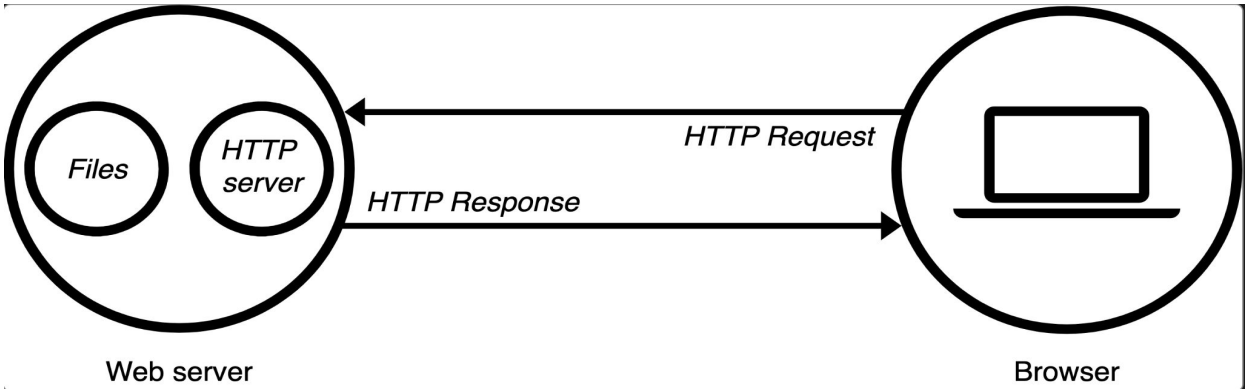
Slides contents are mainly copied from https://book.systemsapproach.org, Computer Networks: A Systems Approach 6th edition Larry Peterson and Bruce Davie

Computer networks

LAN, WAN, Internet

Internet architecture, packet, encapsulation, IP, IP routing, port number, URL, DNS…

Last lecture



How to data transfer between applications in two different machines?

➔ How to “connect” computers?

https://developer.mozilla.org/en-US/docs/Learn/Common_questions/Web_mechanics/What_is_a_web_server

Network

Social Network

- Represents **friendships** between people

https://en.wikipedia.org/wiki/Network_science

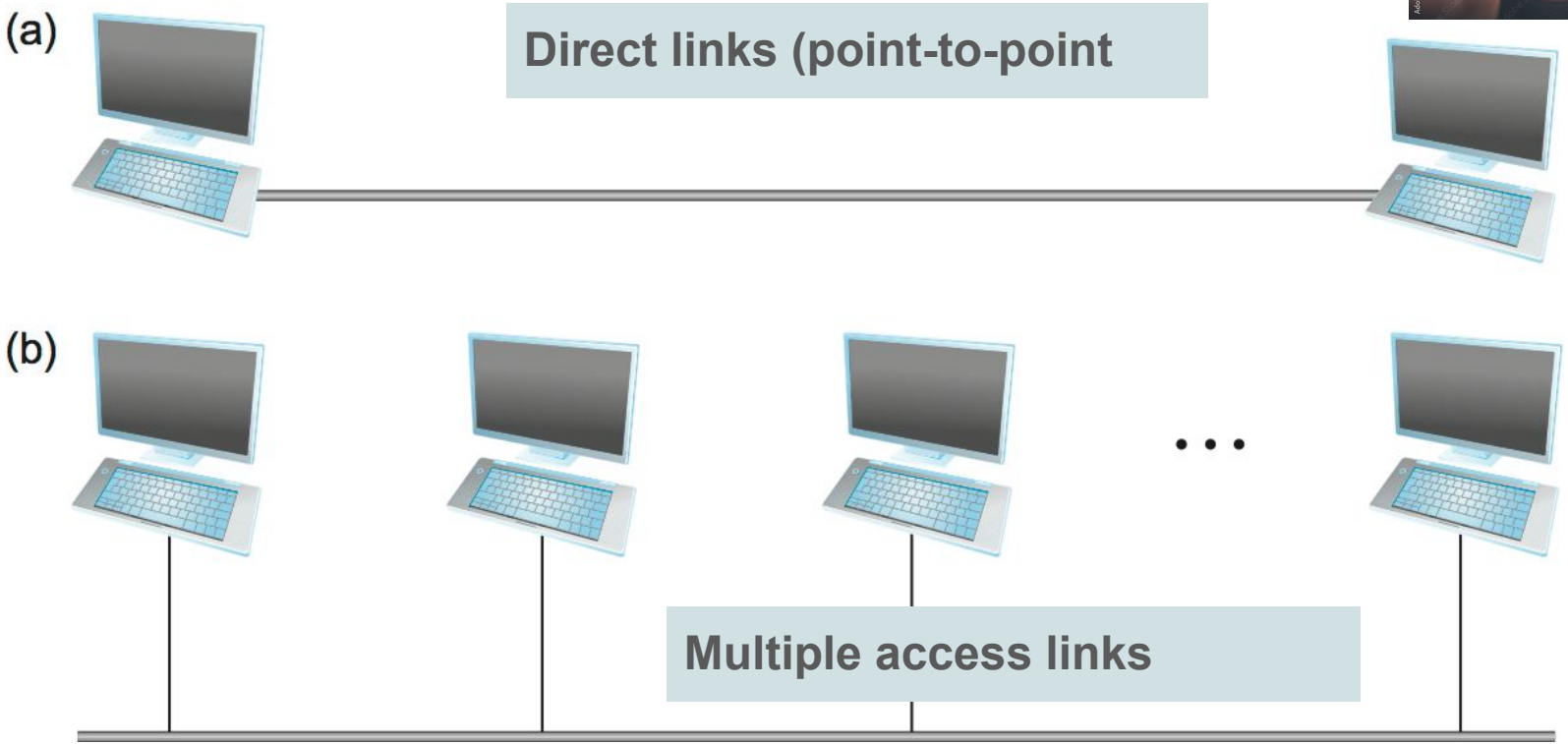
What is computer networking?

Networking, or computer networking,

- is the process of connecting two or more computing devices,
  - such as desktop computers,
  - mobile devices,
  - routers
  - or applications,
- to enable the transmission and exchange of information and resources.

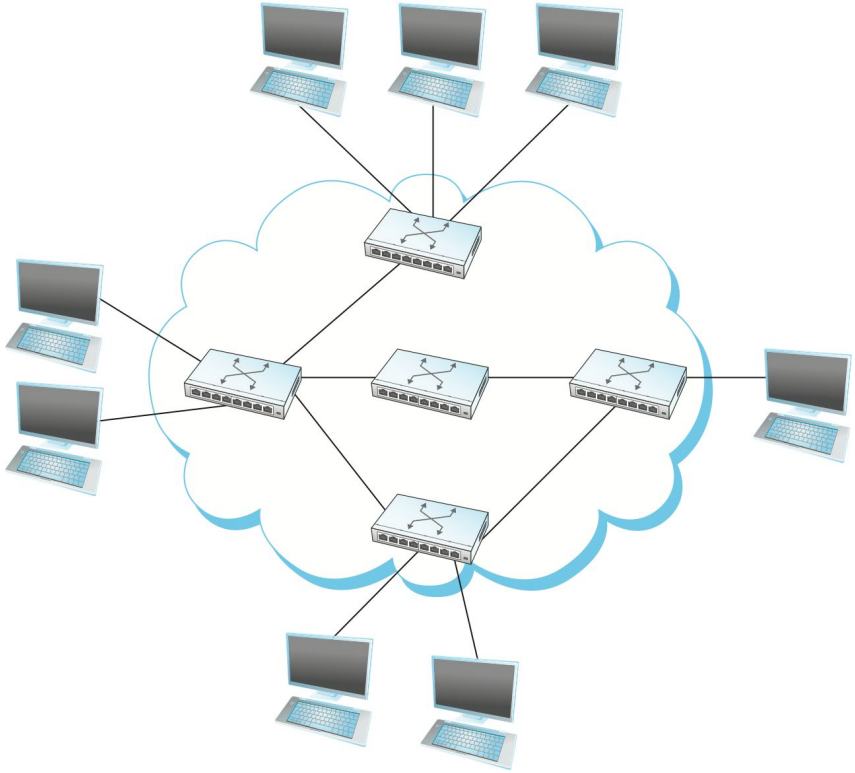
https://www.ibm.com/topics/networking

How to physically connect computers

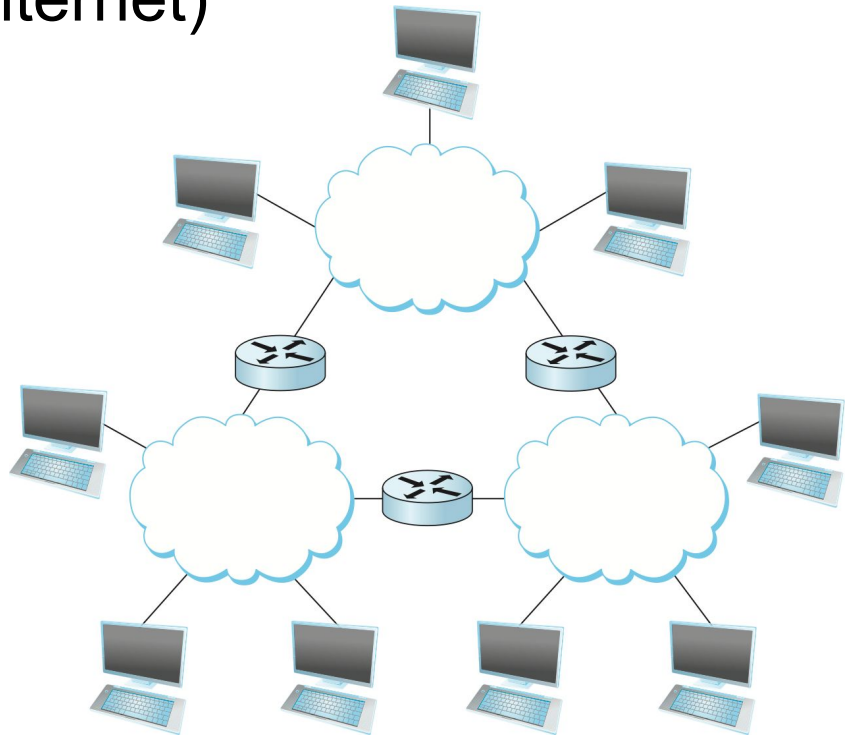


https://book.systemsapproach.org

Switched networks

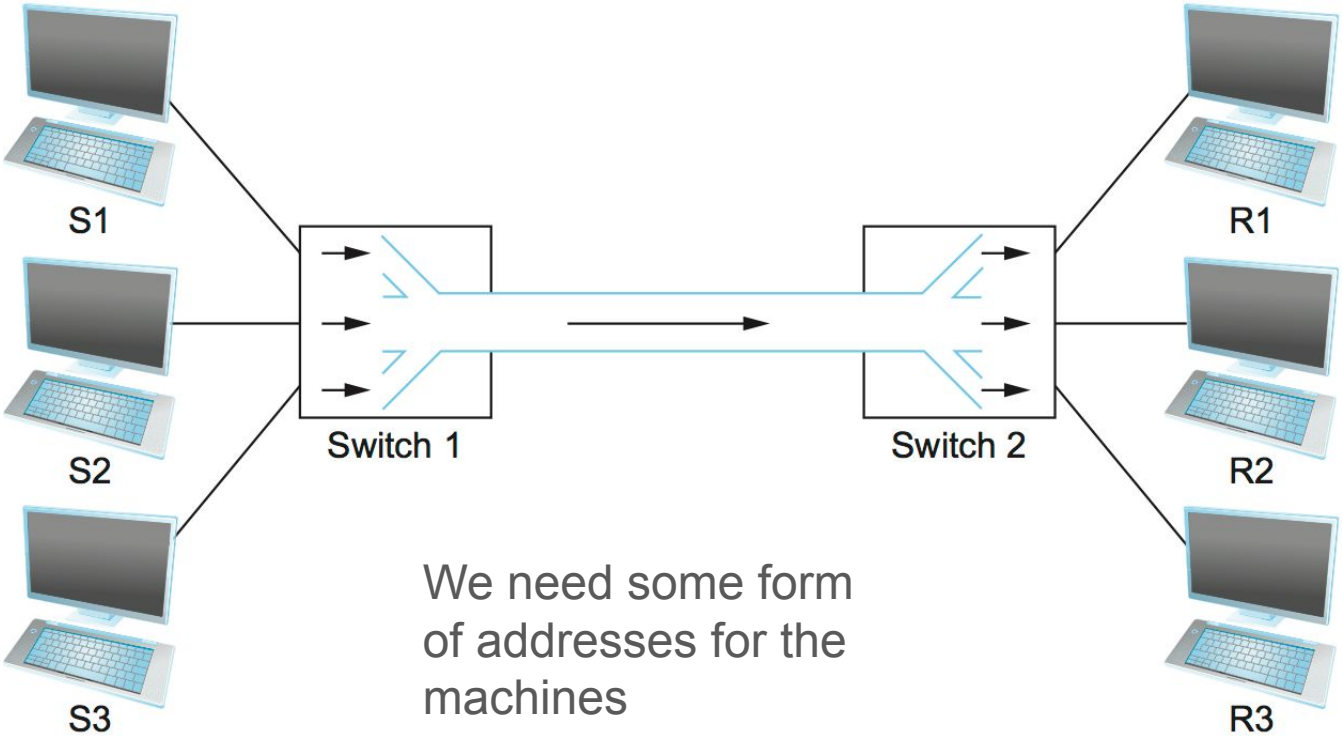


Interconnection of networks (internet)



https://book.systemsapproach.org

a switched network that contains only one physical link

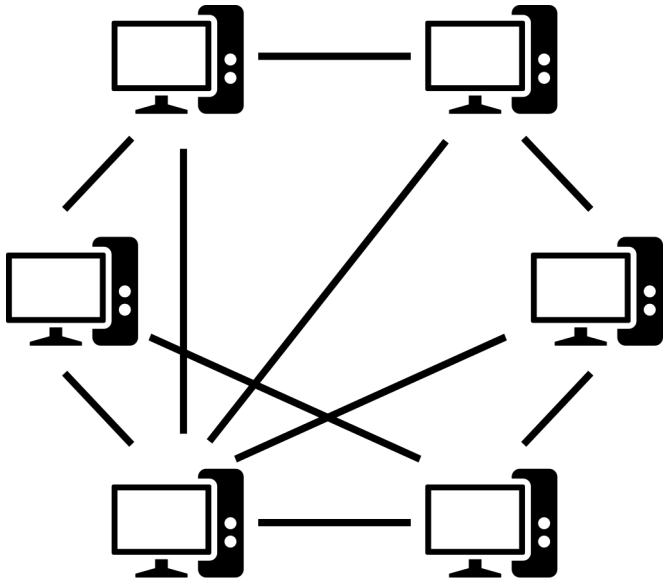


We need some form of addresses for the machines

https://book.systemsapproach.org

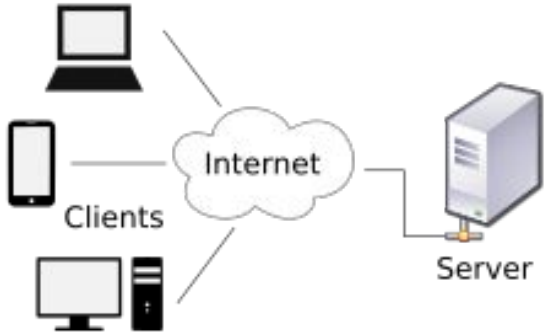
types of computer network architecture

Peer-to-peer architecture



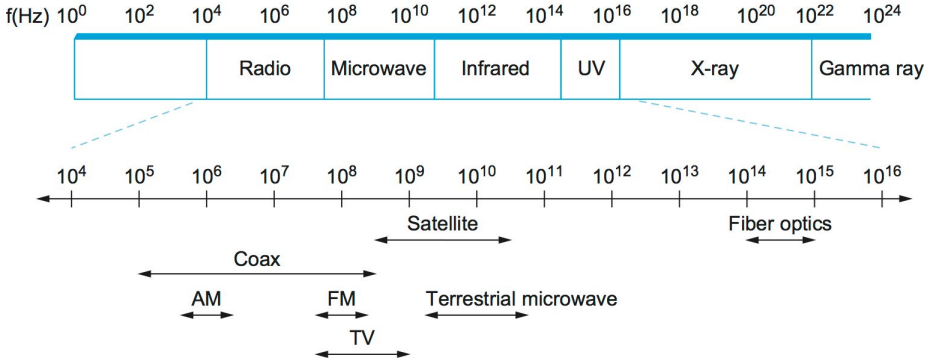
https://en.wikipedia.org/wiki/Peer-to-peer

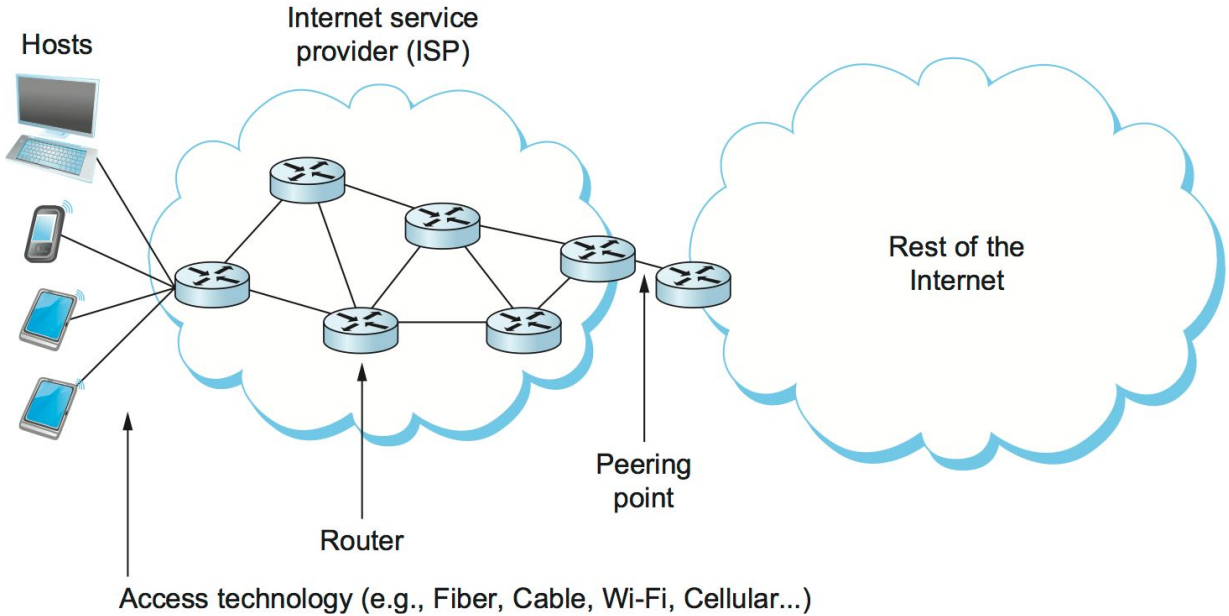
Client-server architecture



https://en.wikipedia.org/wiki/Client%E2%80%93server_model

various types of links that might be found in today's Internet



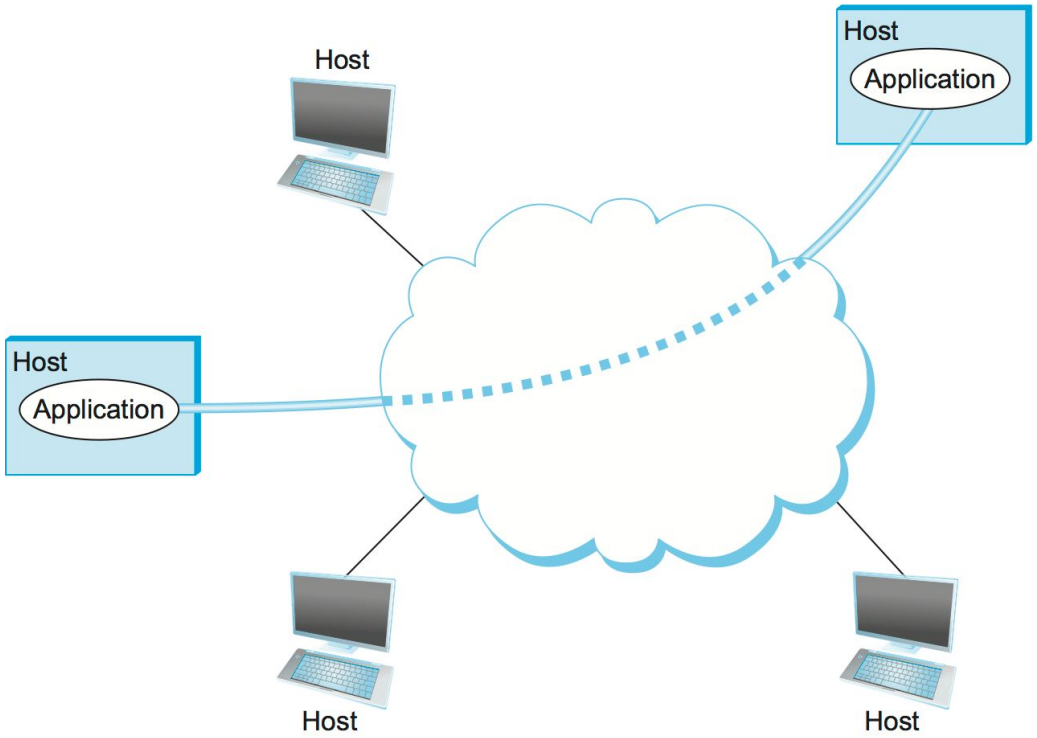


https://book.systemsapproach.org

What are we trying to do?

Data transmission between two applications?

- Many applications needs common services
  - Common communication patterns
    - E.g. request/reply
- Hide the complexity of the network from the applications
  - Provide **logical channels**

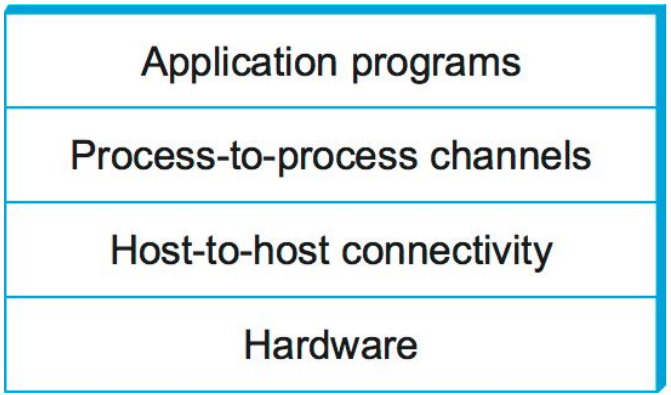


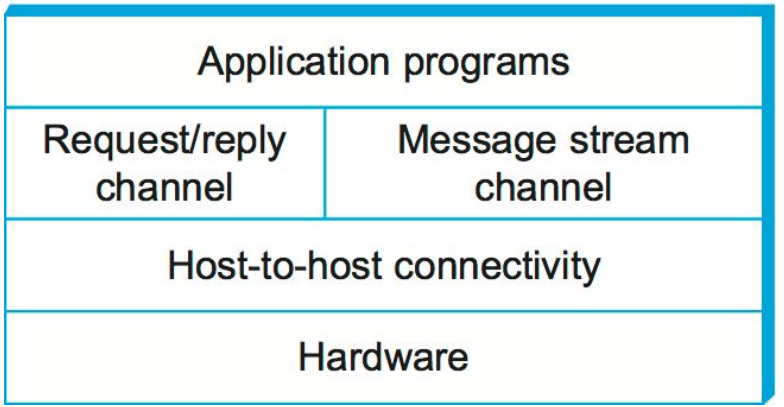
https://book.systemsapproach.org

Layering network architecture

Provide an abstraction for the application that hides the complexity of the network

- Design based on layers: Higher layers use lower layers in the design
- Below two different examples





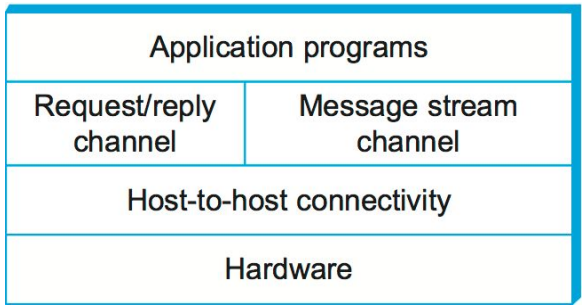
https://book.systemsapproach.org

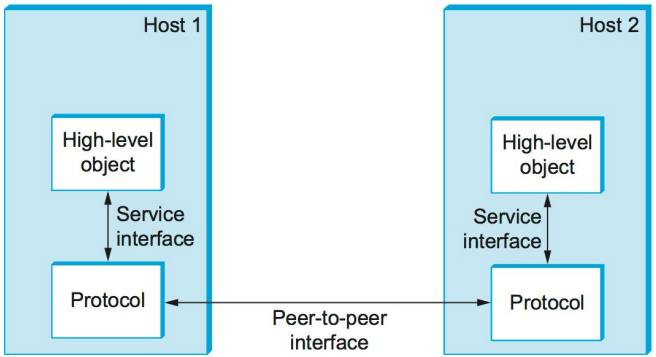
Layering network architecture

a protocol defines

- **a peer interface** to its counterpart (peer) on **another machine**
- **a service interface** to the other objects **on the same computer** that want to use its communication services

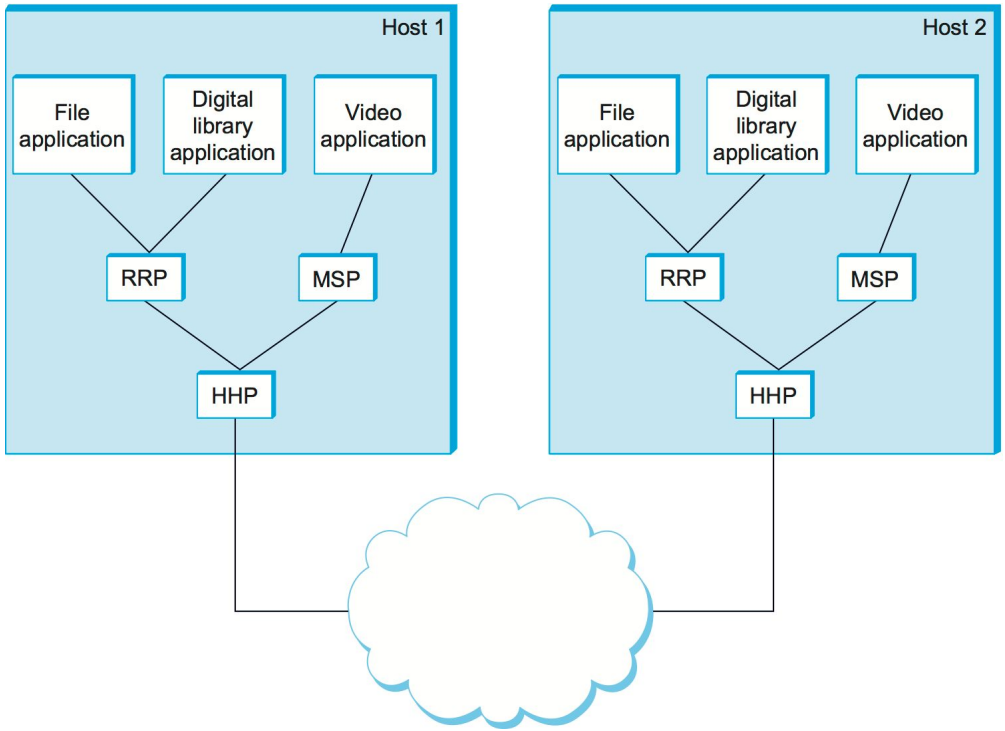
E.g. HTTP





https://book.systemsapproach.org

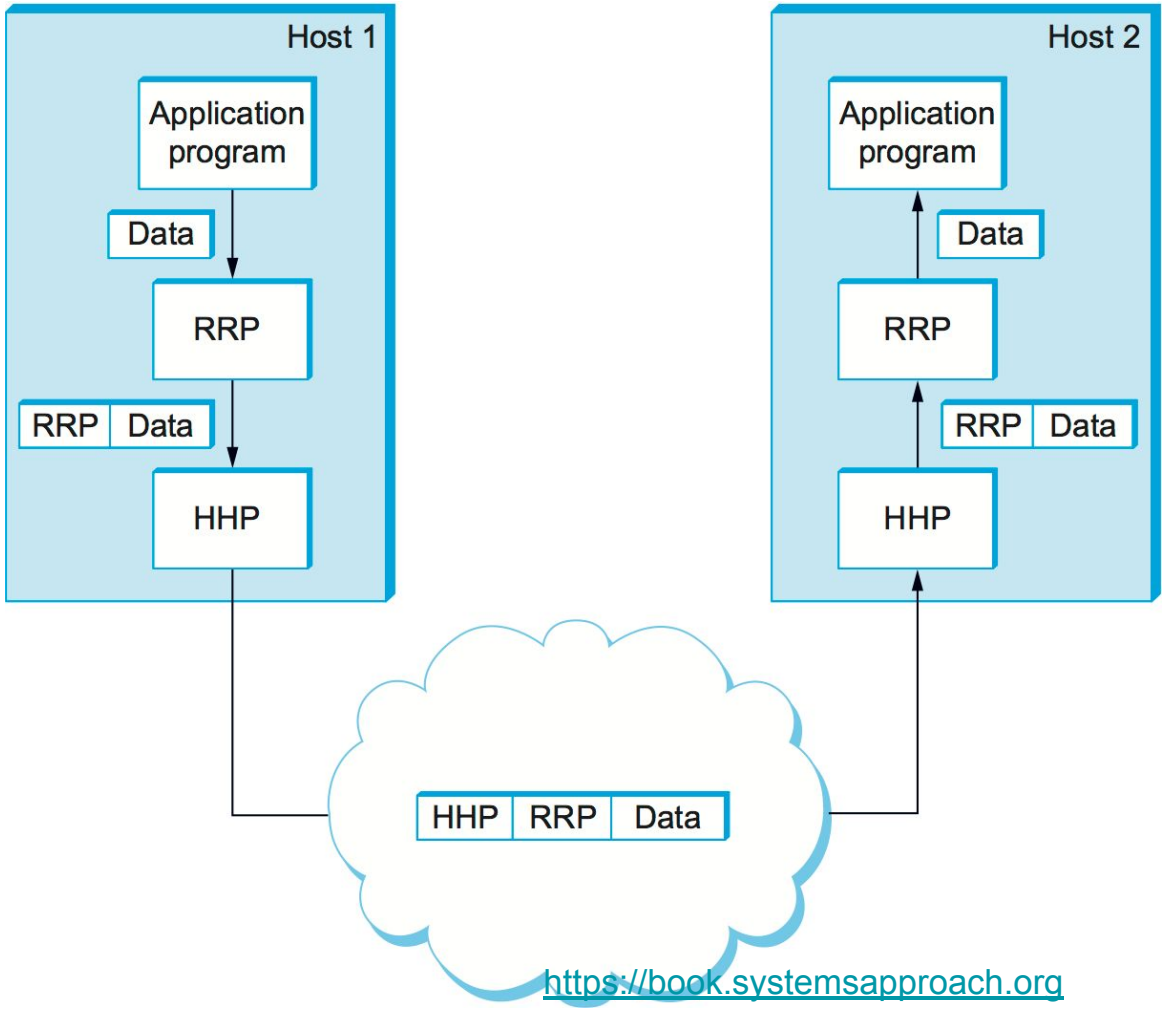
Example of a protocol graph.



https://book.systemsapproach.org

Encapsulation

High-level messages are encapsulated inside of low-level messages

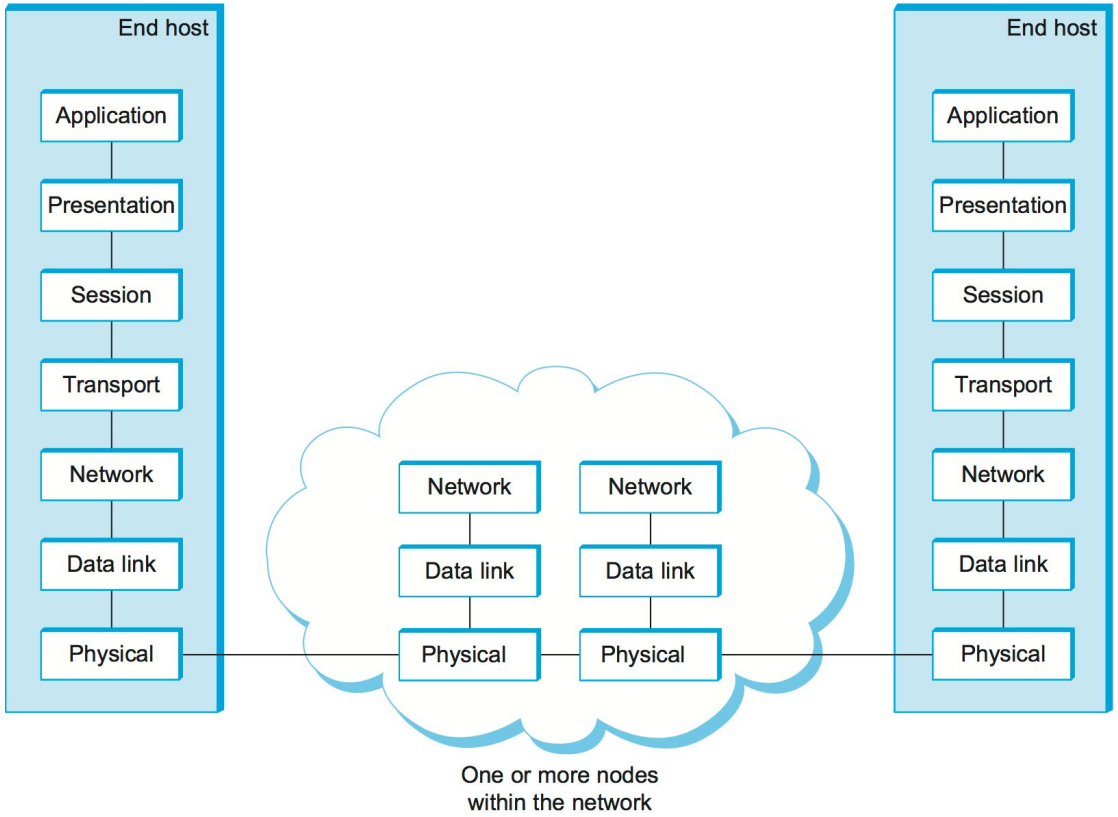


https://book.systemsapproach.org

Open Systems Interconnection (OSI) architecture

Not used in practice
theoretical layering

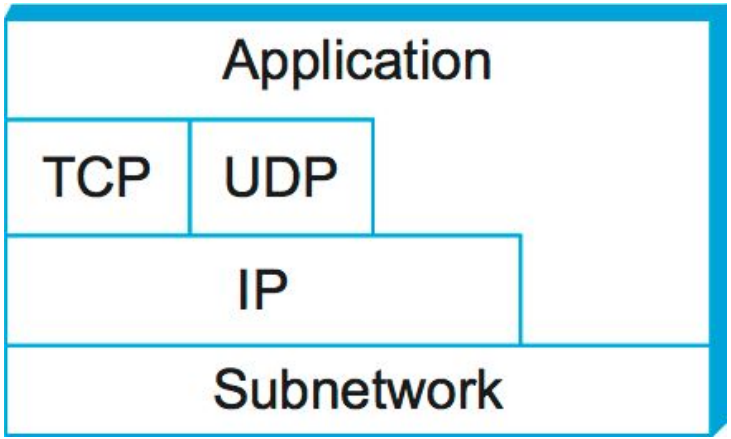
- there is no OSI-based network running today



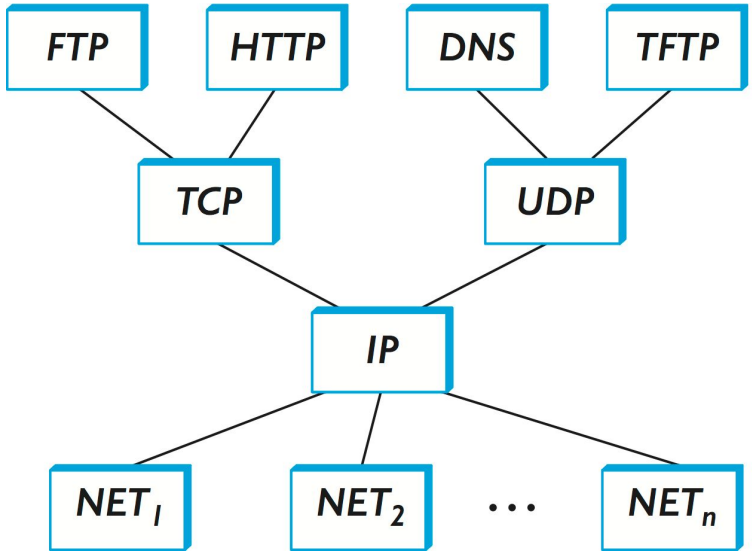
https://book.systemsapproach.org

Internet architecture

The “subnetwork” layer was historically referred to as the “network” layer and is now often referred to as “Layer 2” (influenced by the OSI model)

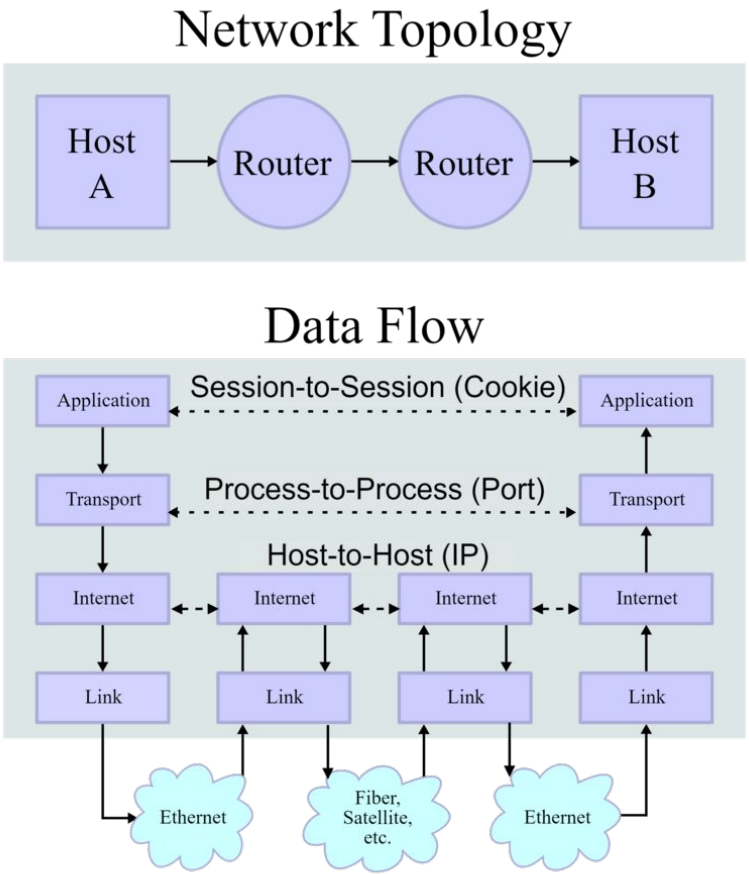


Internet protocol graph.



https://book.systemsapproach.org

Conceptual data flow in a simple network topology of two hosts (A and B) connected by a link between their respective routers.



https://en.wikipedia.org/wiki/Internet_protocol_suite

Internet protocol suite

Application layer

BGP • DHCP (v6) • DNS • FTP • HTTP (HTTP/3) • HTTPS • IMAP • IRC • LDAP • MGCP • MQTT • NNTP • NTP • OSPF • POP • PTP • ONC/RPC • RTP • RTSP • RIP • SIP • SMTP • SNMP • SSH • Telnet • TLS/SSL • XMPP • *more...*

Transport layer

TCP • UDP • DCCP • SCTP • RSVP • QUIC • *more...*

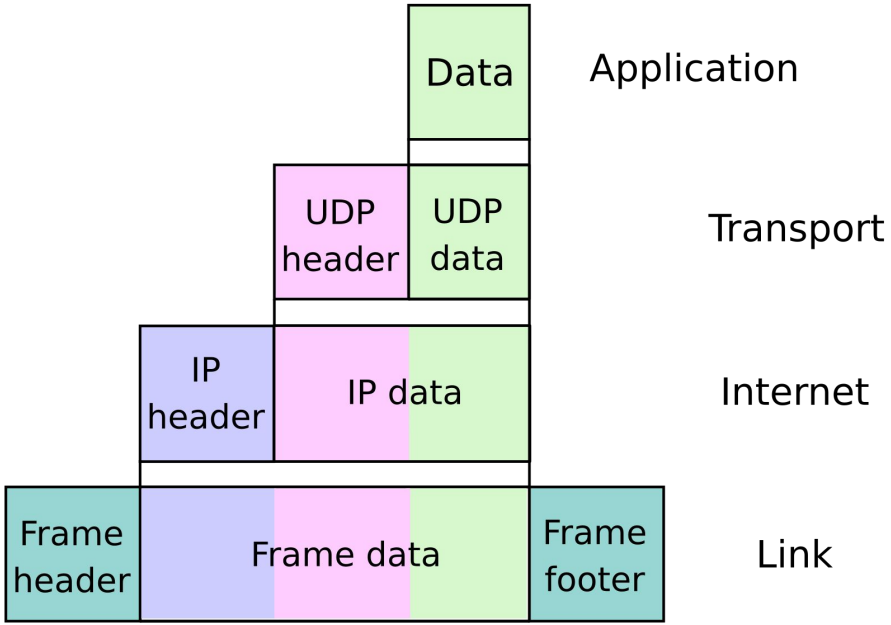
Internet layer

IP (v4 • v6) • ICMP (v6) • NDP • ECN • IGMP • IPsec • *more...*

Link layer

ARP • Tunnels • PPP • MAC • *more...*

Encapsulation of application data carried by UDP to a link protocol frame



https://en.wikipedia.org/wiki/Internet_protocol_suite

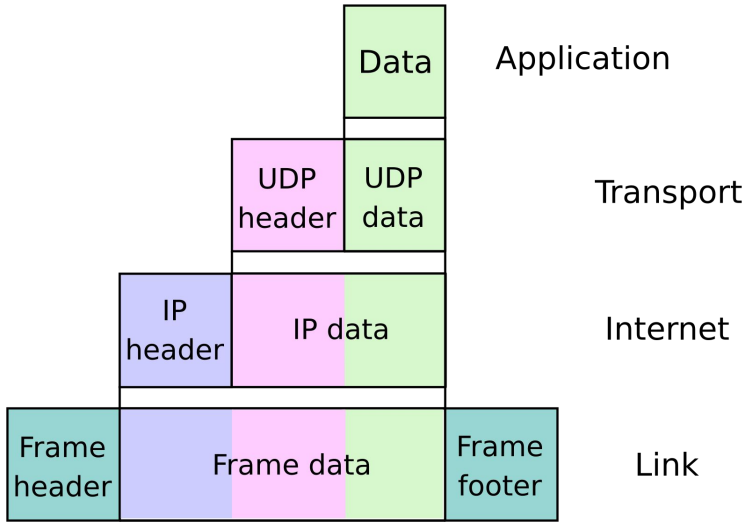
Internet protocol (IP)

IP has the task of delivering packets from the source host to the destination host

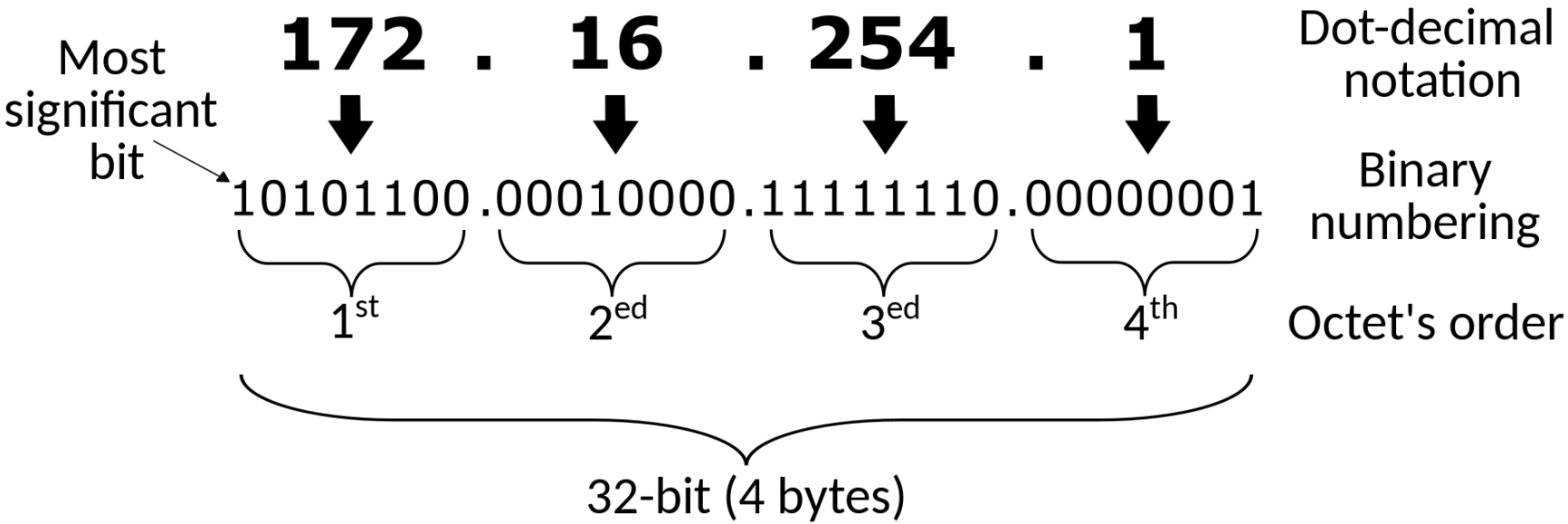
- based on the **IP addresses** in the **packet headers.**

An Internet Protocol address (IP address) is a numerical label

- E.g. 192.0.2.1
- Indicates the address of the machine

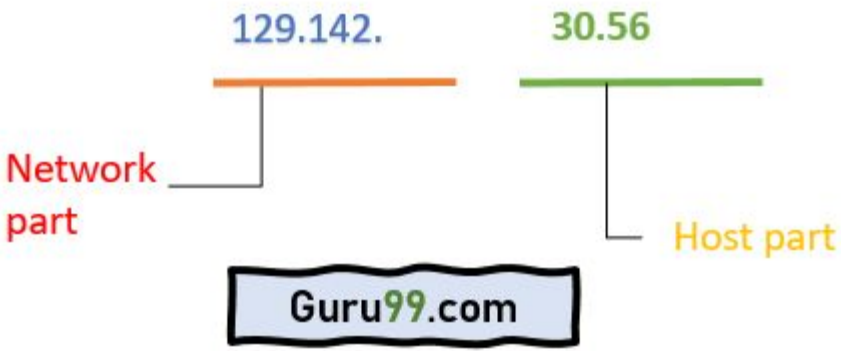


IPv4 addresses



https://en.wikipedia.org/wiki/IP_address

IP address = prefix + suffix



- Classful (A, B,C, D) length is fixed
  - E.g. Class A,
    - network prefix: 8 bit,
    - host 24 bits

- Classless
  - Subnet mask
  - 198.51.100.14/**24**
    - **IP address:** 198.51.100.14
    - **network prefix:** 198.51.100.0
    - **subnet mask:** 255.255.255.0

https://en.wikipedia.org/wiki/Classless_Inter-Domain_Routing

https://www.guru99.com/ip-address-classes.html

IP routing

On the same local area network(LAN), deliver directly

IP forwarding: if not on the same LAN, forward

Two LANs (200.0.0 and 200.0.1) joined by three routers R1,R2,R3

If A sends to D, at 200.0.1.37,

- it puts this address into the IP header,
- notes that 200.0.0 ≠ 200.0.1,
- and thus concludes D is not a local delivery.

A therefore sends the packet to its router R1, using LAN delivery.

- R1 looks up the destination network 200.0.1 in its forwarding table
- and forwards the packet to R2,
  - which in turn forwards it to R3.

R3 now sees that it *is* connected directly to the destination network 200.0.1,

- and delivers the packet via the LAN to D, by looking up D's physical address.

https://intronetworks.cs.luc.edu/current2/html/intro.html

https://en.wikipedia.org/wiki/IP_routing

Loading a website in client-server model

- Request a webpage with an IP
  - IPv4:

    142.251.46.238

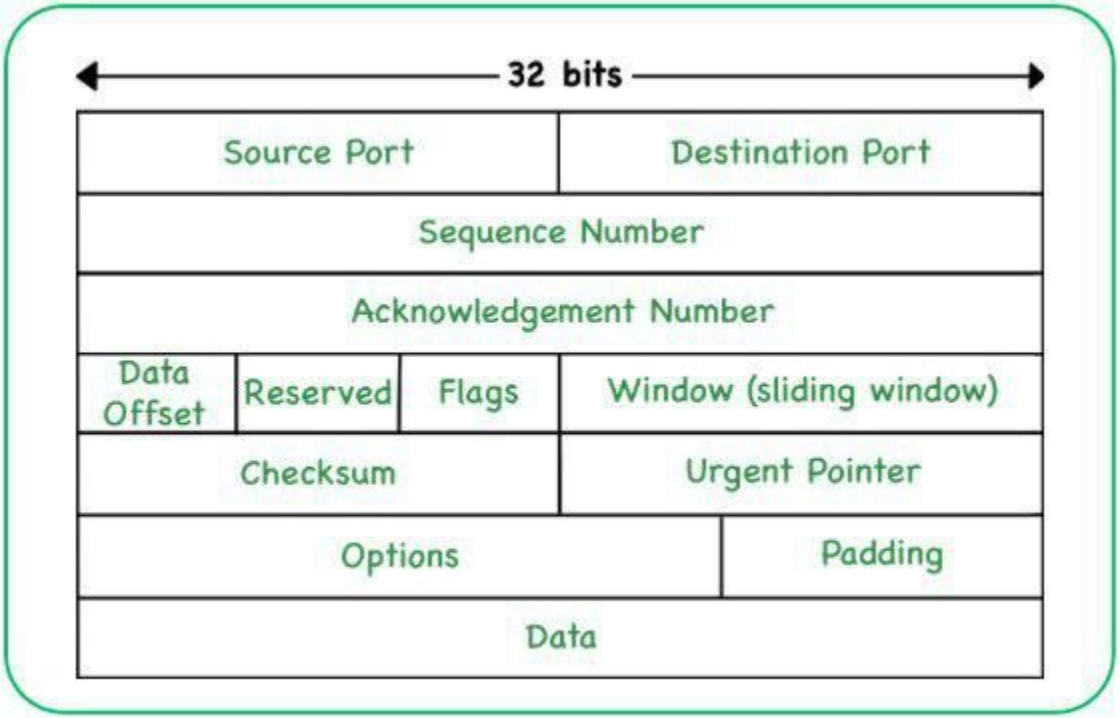
  - IPv6:

    2607:f8b0:4005:806::200e

On server side,
How to know which protocol is used?

How OS delivers packages to the right applications?

Port numbers



TCP/IP Packet Format - GeeksforGeeks

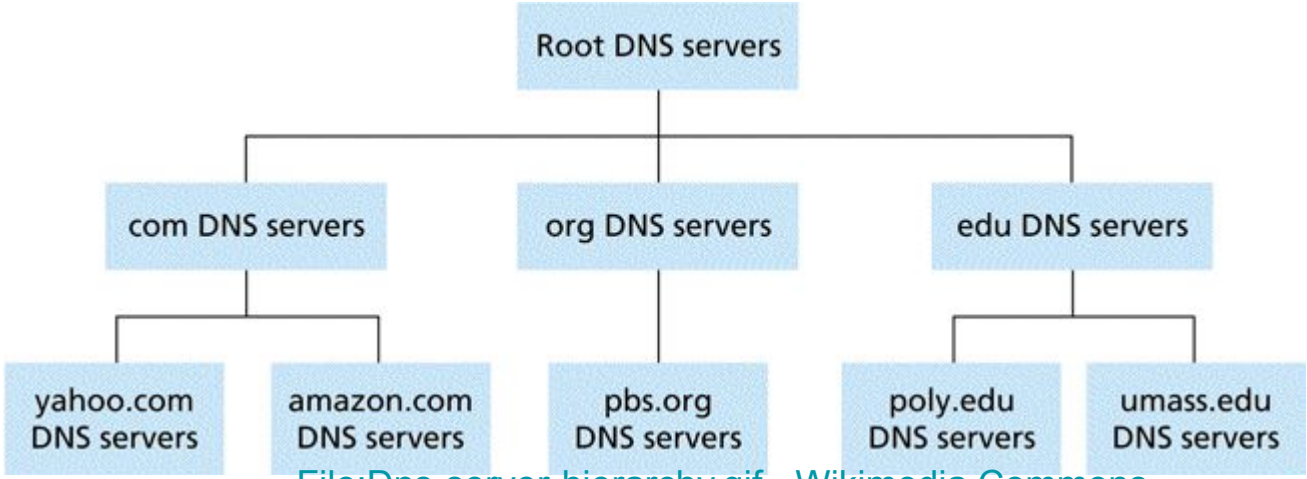
Port numbers

- A connection endpoint
  - Host IP address
  - Transmission protocol
- HTTP port number is 80
  - 142.251.46.238**:80**
- **Example request**
  - $ wget 142.251.46.238:80

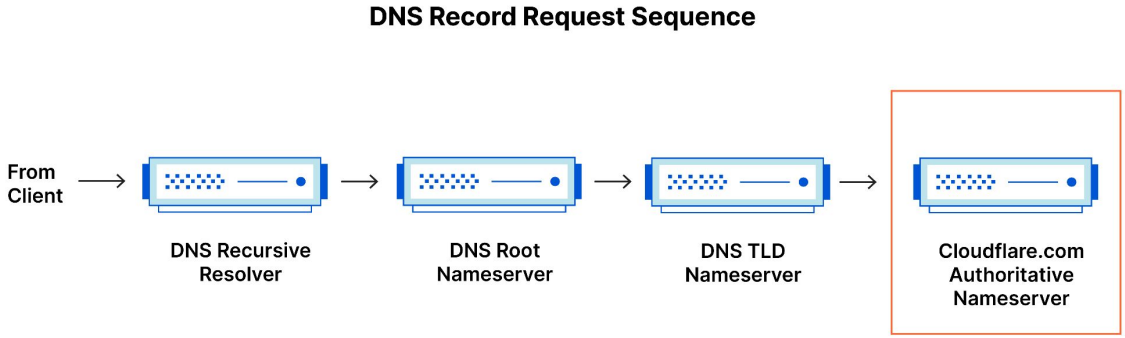
Using a URL

- **Instead of IP addresses**
  - **142.251.46.238**
- Use URL
  - **www.google.com**

Your computer asks **another computer (Domain Name System(DNS) server)** for the IP information

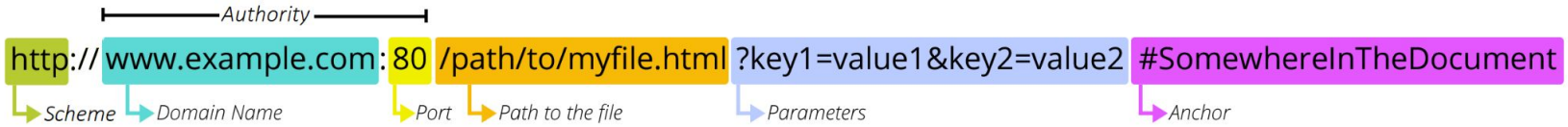


File:Dns-server-hierarchy.gif - Wikimedia Commons



What is DNS? | How DNS works | Cloudflare

URL



www.example.com:80

Domain Name

Port

/path/to/myfile.html

Path to resource

?key1=value1&key2=value2

Parameters

#SomewhereInTheDocument

Anchor

https://developer.mozilla.org/en-US/docs/Learn/Common_questions/Web_mechanics/What_is_a_URL

Next week: how to secure connection?