# Advanced Computer Networks

By: Mohammad Nassiri

Islamic Azad University of Hamedan

2009/2010 Semester II

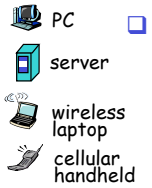


# Review on Basic Concepts of Networking

March 11th 2010

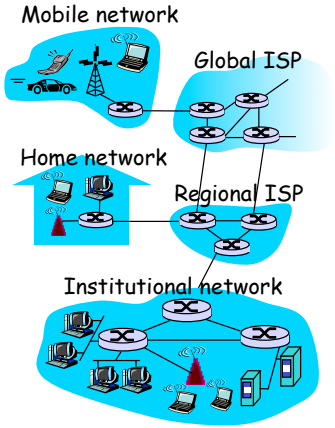


## What's the Internet




- millions of connected computing devices: **hosts = end systems**
  - running **network apps**
- **communication links**
  - fiber, copper, radio, satellite
  - transmission rate = **bandwidth**
- **routers:** forward packets (chunks of data)

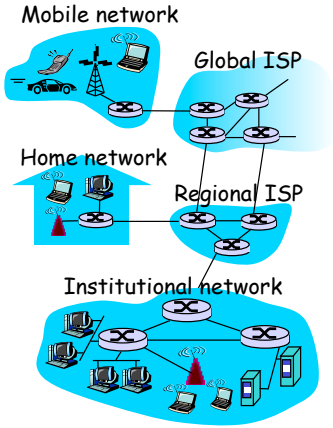







## What's the Internet

- **protocols** control sending, receiving of msgs
  - e.g., TCP, IP, HTTP, Skype, Ethernet
- **Internet:** "network of networks"
  - loosely hierarchical
  - public Internet versus private intranet
- Internet standards
  - RFC: Request for comments
  - IETF: Internet Engineering Task Force



# What's a protocol?

network protocols:

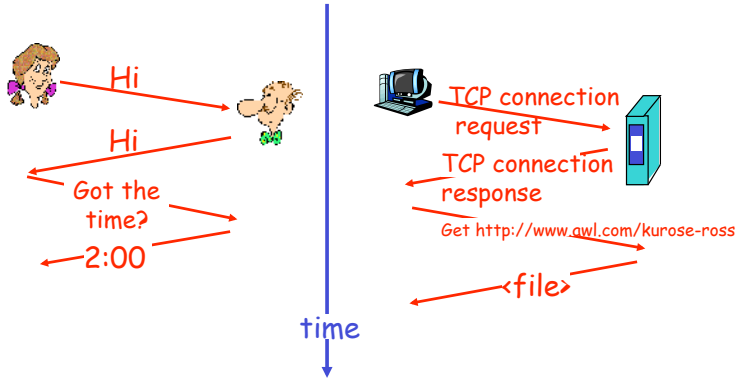
- all communication activity in Internet governed by protocols

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



# What's a protocol?

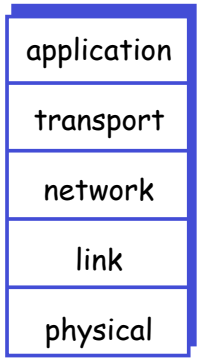
a human protocol and a computer network protocol:






# 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"

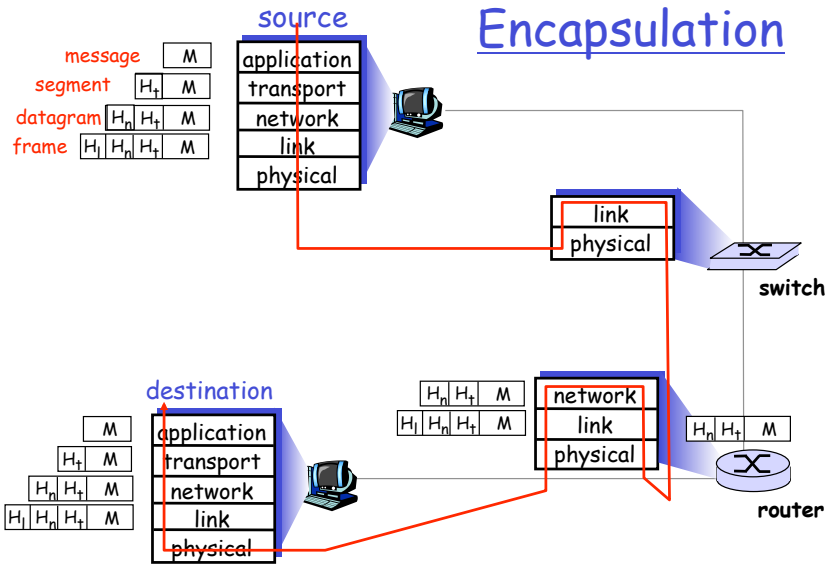





# ISO/OSI reference model

- presentation: allow applications to interpret meaning of data, e.g., encryption, compression, machine-specific conventions
- session: synchronization, checkpointing, recovery of data exchange
- Internet stack "missing" these layers!
  - these services, **if needed**, must be implemented in application
  - needed?

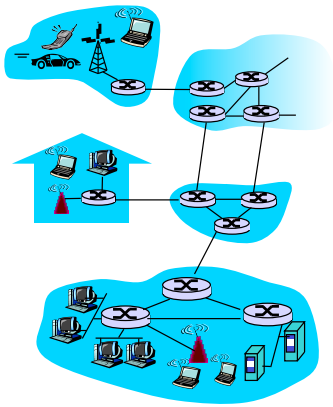
# Encapsulation






# A closer look at network structure:

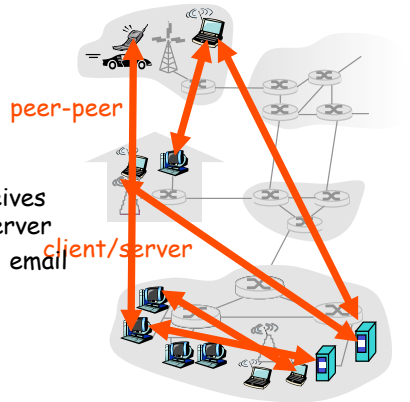
- network edge:
  applications and hosts
- access networks, physical media:
  wired, wireless communication links
- network core:
  - interconnected routers
  - network of networks





# The network edge:

- end systems (hosts):
  - run application programs
  - e.g. Web, email
  - at "edge of network"
- client/server model
  - client host requests, receives service from always-on server
  - e.g. Web browser/server; email client/server
- peer-peer model:
  - minimal (or no) use of dedicated servers
  - e.g. Skype, BitTorrent





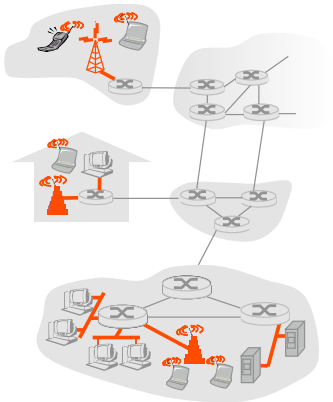

# Access networks and physical media

**Q: How to connect end systems to edge router?**

- residential access nets
- institutional access networks (school, company)
- mobile access networks

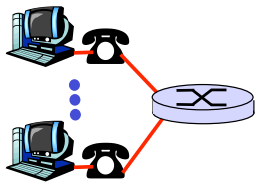
**Keep in mind:**

- bandwidth (bits per second) of access network?
- shared or dedicated?



## Residential access: point to point access

- Dialup via modem
  - up to 56Kbps direct access to router (often less)
  - Can't surf and phone at same time: can't be "always on"



- DSL: digital subscriber line
  - deployment: telephone company (typically)
  - up to 1 Mbps upstream
  - up to 28 Mbps downstream
  - dedicated physical line to telephone central office



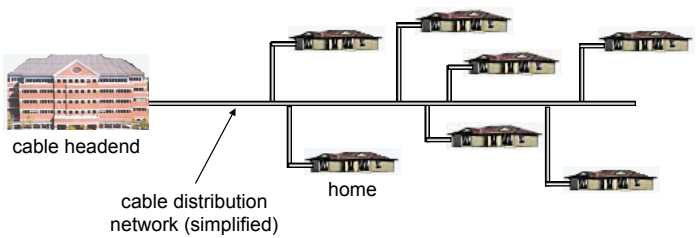
## Residential access: cable modems

- HFC: hybrid fiber coax
  - asymmetric: up to 30Mbps downstream, 2 Mbps upstream
- network of cable and fiber attaches homes to ISP router
  - homes share access to router
- deployment: available via cable TV companies



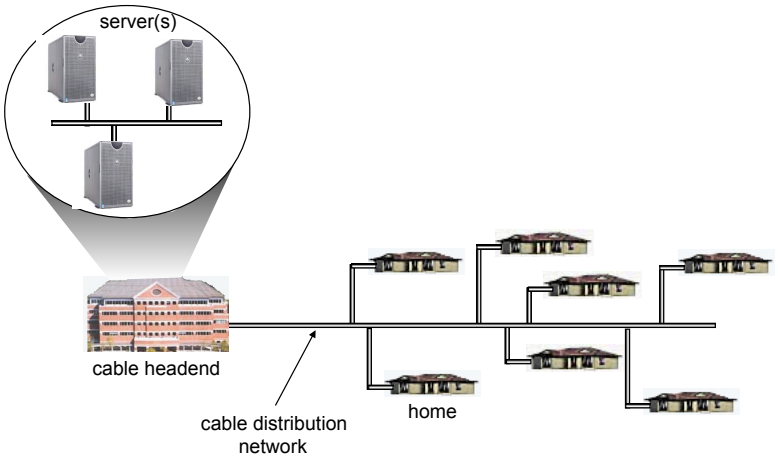
## Cable Network Architecture: Overview

Typically 500 to 5,000 homes

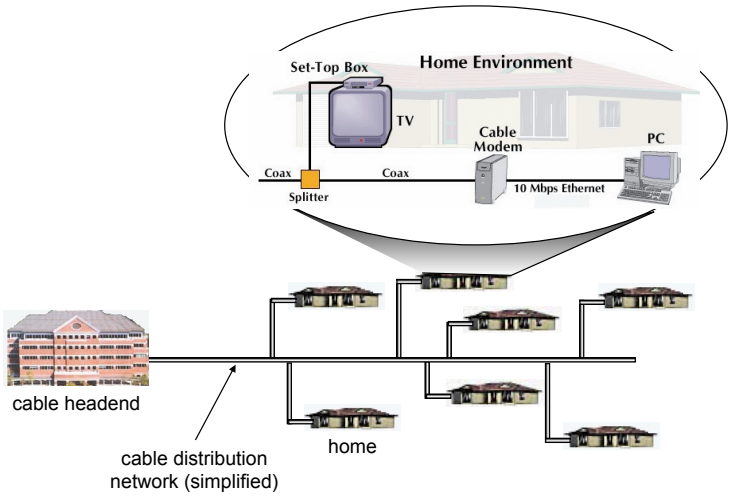





## Cable Network Architecture: Overview

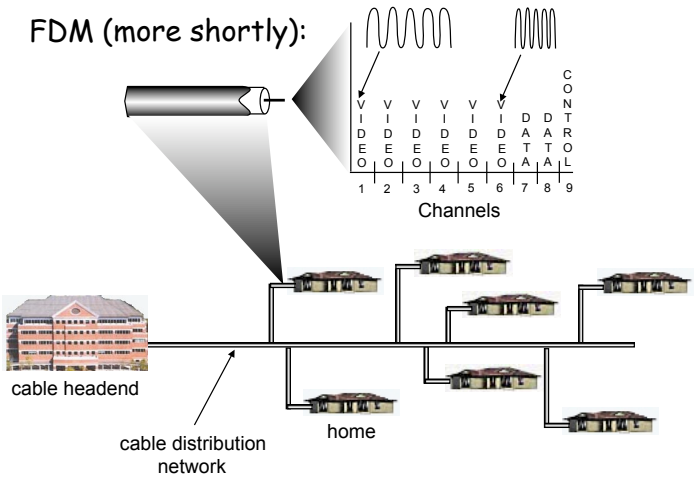


## Cable Network Architecture: Overview





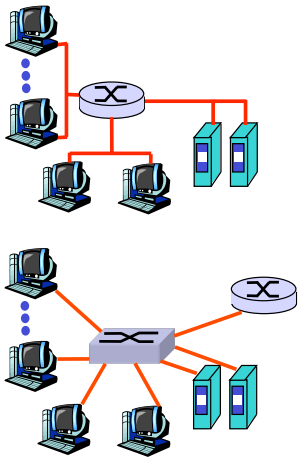

## Cable Network Architecture: Overview






## Company access: local area networks

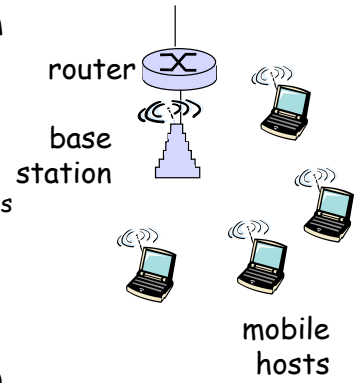
- company/univ local area network (LAN) connects end system to edge router
- Ethernet:
  - 10 Mbs, 100Mbps, 1Gbps, 10Gbps Ethernet
  - modern configuration: end systems connect into **Ethernet switch**





## Wireless access networks

- shared **wireless** access network connects end system to router
  - via base station aka "access point"
- wireless LANs:
  - 802.11b/g (WiFi): 11 or 54 Mbps
- wider-area wireless access
  - provided by telco operator
  - ~1Mbps over cellular system (EVDO, HSDPA)
  - next up (?): WiMAX (10's Mbps) over wide area



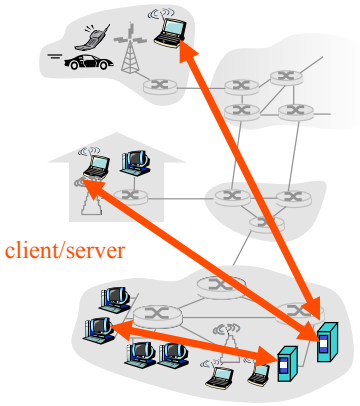
# Application architectures

- Client-server
- Peer-to-peer (P2P)
- Hybrid of client-server and P2P

  - specific protocols:
    - HTTP
    - FTP
    - SMTP, POP, IMAP
    - DNS
    - P2P: BitTorrent, Skype



# Client-server architecture




server:

- always-on host
- permanent IP address
- server farms for scaling

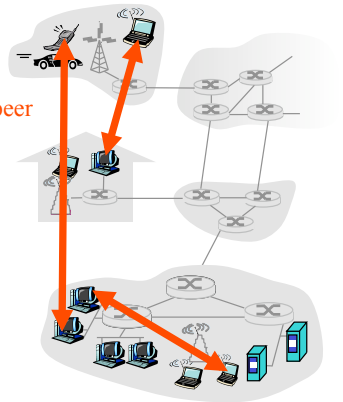
clients:

- communicate with server
- may be intermittently connected
- may have dynamic IP addresses
- do not communicate directly with each other



# Pure P2P architecture

- **no** always-on server
- arbitrary end systems directly communicate
- peers are intermittently connected and change IP addresses




Highly scalable but difficult to manage



# App-layer protocol defines

- Types of messages exchanged,
  - e.g., request, response
- Message syntax:
  - what fields in messages & how fields are delineated
- Message semantics
  - meaning of information in fields
- Rules for when and how processes send & respond to messages

Public-domain protocols:

- defined in RFCs
- allows for interoperability
- e.g., HTTP, SMTP

Proprietary protocols:

- e.g., Skype

## Transport service requirements of common apps

| Application | Data loss | Throughput | Time Sensitive |
|---|---|---|---|
| file transfer | no loss | elastic | no |
| e-mail | no loss | elastic | no |
| Web documents | no loss | elastic | no |
| real-time audio/video | loss-tolerant | audio: 5kbps-1Mbps<br>video:10kbps-5Mbps | yes, 100's msec |
| stored audio/video | loss-tolerant | same as above | yes, few secs |
| interactive games | loss-tolerant | few kbps up | yes, 100's msec |
| instant messaging | no loss | elastic | yes and no |



## Internet transport protocols services

### TCP service:

- **connection-oriented:** setup required between client and server processes
- **reliable transport** between sending and receiving process
- **flow control:** sender won't overwhelm receiver
- **congestion control:** throttle sender when network overloaded
- **does not provide:** timing, minimum throughput guarantees, security

### UDP service:

- unreliable data transfer between sending and receiving process
- does not provide: connection setup, reliability, flow control, congestion control, timing, throughput guarantee, or security

Q: why bother? Why is there a UDP?



## Internet apps: application, transport protocols

| Application | Application layer protocol | Underlying transport protocol |
|---|---|---|
| e-mail | SMTP [RFC 2821] | TCP |
| remote terminal access | Telnet [RFC 854] | TCP |
| Web | HTTP [RFC 2616] | TCP |
| file transfer | FTP [RFC 959] | TCP |
| streaming multimedia | HTTP (eg Youtube), RTP [RFC 1889] | TCP or UDP |
| Internet telephony | SIP, RTP, proprietary (e.g., Skype) | typically UDP |



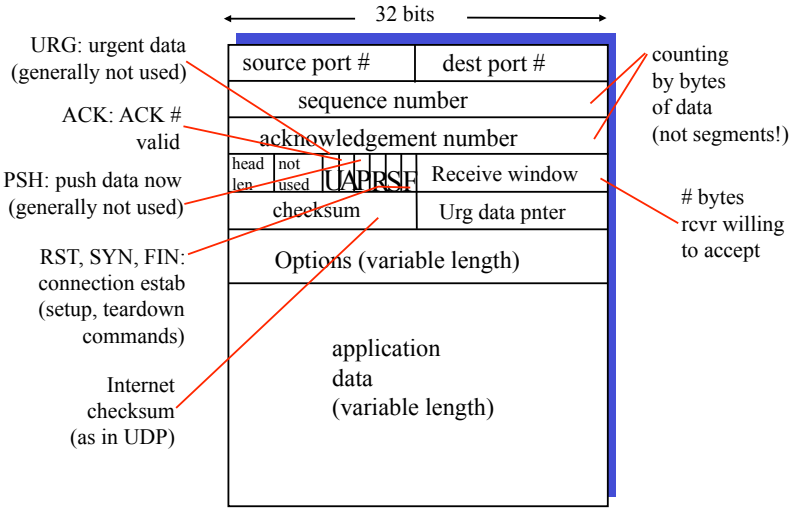
## TCP: Overview

RFCs: 793, 1122, 1323, 2018, 2581

- point-to-point:
  - one sender, one receiver
- reliable, in-order **byte steam:**
- pipelined:
  - TCP congestion and flow control set window size
- **send & receive buffers**


- full duplex data:
  - bi-directional data flow in same connection
  - MSS: maximum segment size
- connection-oriented:
  - handshaking (exchange of control msgs) init's sender, receiver state before data exchange
- flow controlled:
  - sender will not overwhelm receiver

# TCP segment structure






# UDP: User Datagram Protocol [RFC 768]

- "no frills," "bare bones" Internet transport protocol
- "best effort" service, UDP segments may be:
  - lost
  - delivered out of order to app
- **connectionless:**
  - no handshaking between UDP sender, receiver
  - each UDP segment handled independently of others

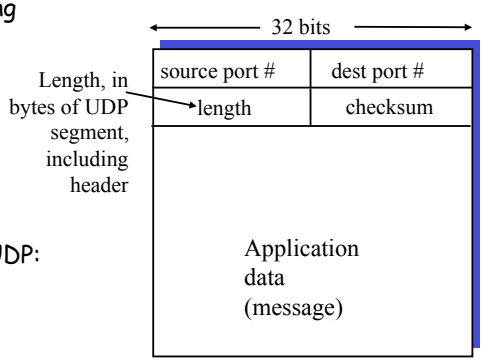
**Why is there a UDP?**

- no connection establishment (which can add delay)
- simple: no connection state at sender, receiver
- small segment header
- no congestion control: UDP can blast away as fast as desired



# UDP: more

- often used for streaming multimedia apps
  - loss tolerant
  - rate sensitive
- other UDP uses
  - DNS
  - SNMP
- reliable transfer over UDP: add reliability at application layer
  - application-specific error recovery!

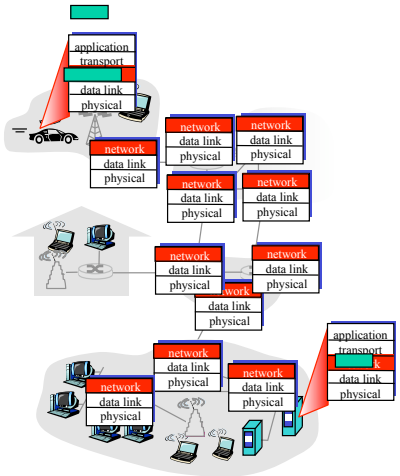



UDP segment format



# Network layer

- transport segment from sending to receiving host
- on sending side encapsulates segments into datagrams
- on rcving side, delivers segments to transport layer
- network layer protocols in **every** host, router
- router examines header fields in all IP datagrams passing through it



# Two Key Network-Layer Functions

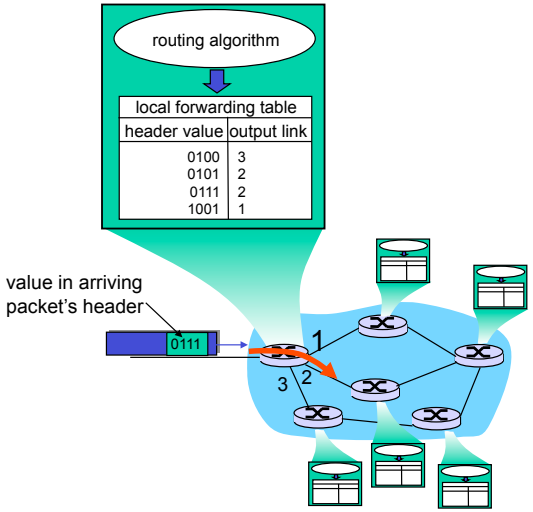
- **forwarding:** move packets from router's input to appropriate router output
- **routing:** determine route taken by packets from source to dest.
  - **routing algorithms**

analogy:

- routing: process of planning trip from source to dest
- forwarding: process of getting through single interchange



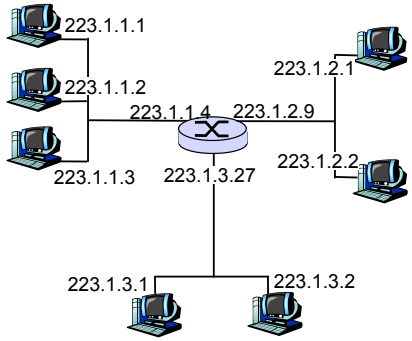
# Interplay between routing and forwarding






# IP Addressing: introduction

- IP address: 32-bit identifier for host, router **interface**
- **interface:** connection between host/router and physical link
  - router's typically have multiple interfaces
  - host typically has one interface
  - IP addresses associated with each interface



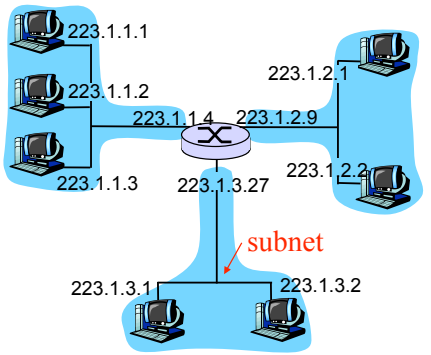

223.1.1.1 = 11011111 00000001 00000001 00000001

223 1 1 1



# Subnets

- IP address:
  - subnet part (high order bits)
  - host part (low order bits)
- **What's a subnet ?**
  - device interfaces with same subnet part of IP address
  - can physically reach each other without intervening router

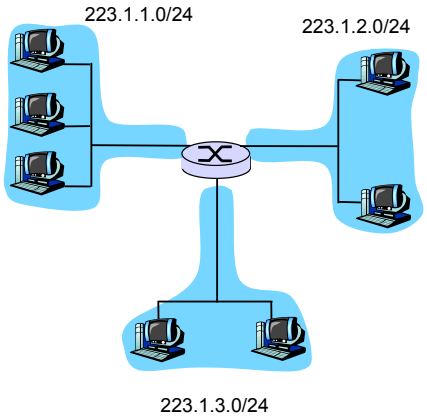



network consisting of 3 subnets

# Subnets

Recipe

- To determine the subnets, detach each interface from its host or router, creating islands of isolated networks. Each isolated network is called a subnet.

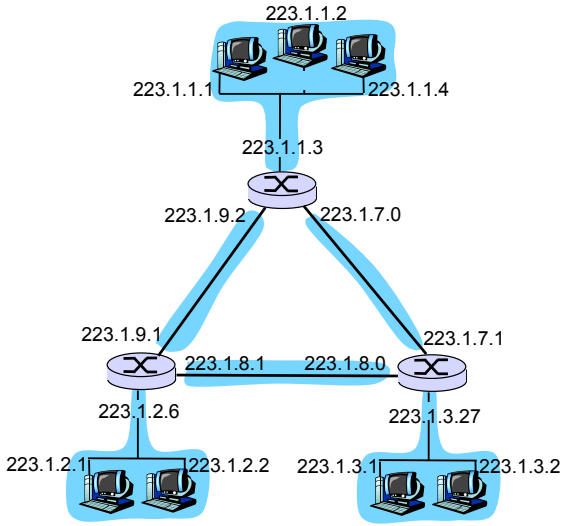



Subnet mask: /24



# Subnets

How many?






# IP addressing: CIDR

CIDR: Classless InterDomain Routing

- subnet portion of address of arbitrary length
- address format: a.b.c.d/x, where x is # bits in subnet portion of address


200.23.16.0/23

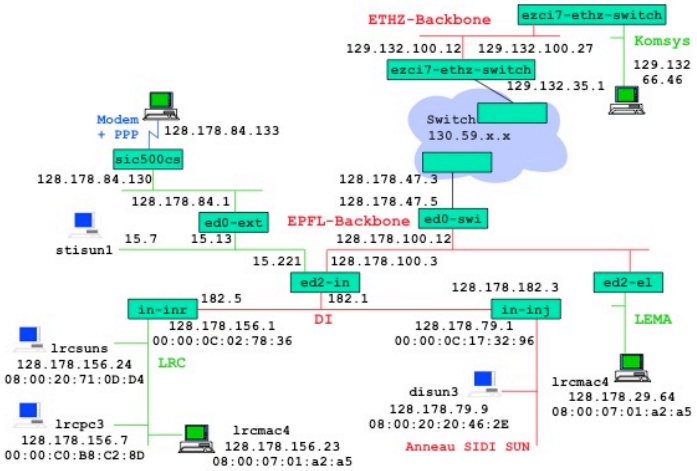


# IP addresses: how to get one?

Q: How does a **host** get IP address?

- hard-coded by system admin in a file
  - Windows: control-panel->network->configuration->tcp/ip->properties
  - UNIX: /etc/rc.config
- DHCP: Dynamic Host Configuration Protocol: dynamically get address from as server
  - "plug-and-play"

# Example of Network






# Routing in Internet

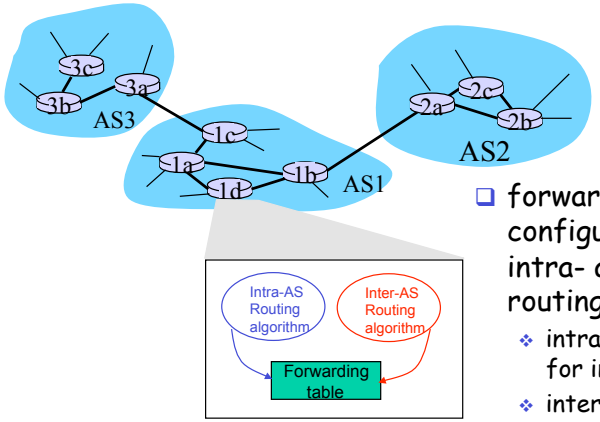
- aggregate routers into regions, "autonomous systems" (AS)
- routers in same AS run same routing protocol
  - "intra-AS" routing protocol
  - routers in different AS can run different intra-AS routing protocol

Gateway router

- Direct link to router in another AS



# Interconnected ASes



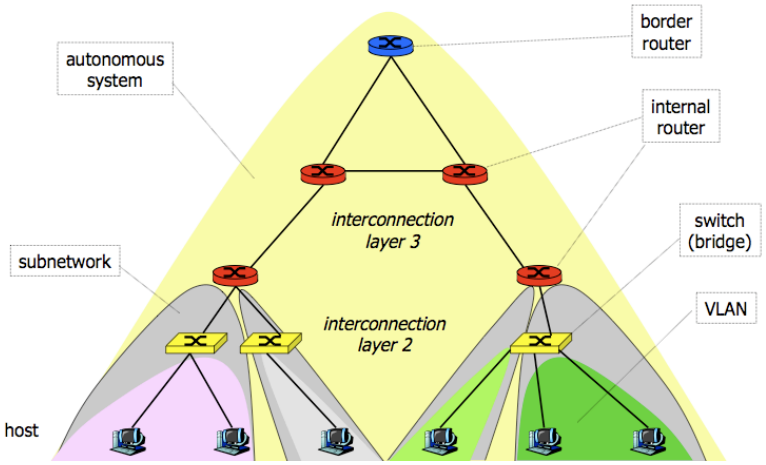

- forwarding table configured by both intra- and inter-AS routing algorithm
  - intra-AS sets entries for internal dests
  - inter-AS & intra-As sets entries for external dests



# Intra-AS Routing

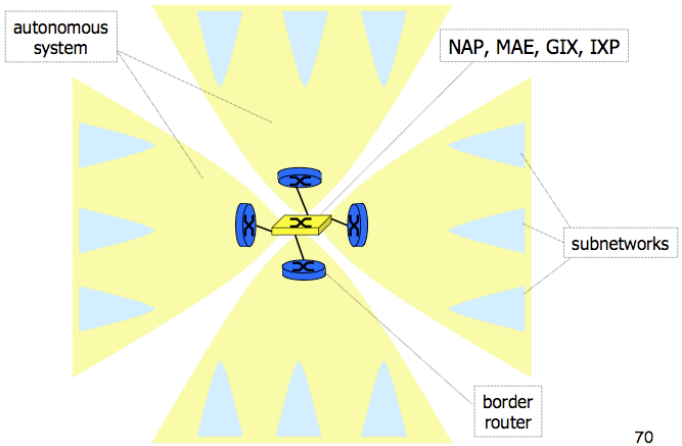
- also known as Interior Gateway Protocols (IGP)
- most common Intra-AS routing protocols:
  - RIP: Routing Information Protocol
  - OSPF: Open Shortest Path First
  - IGRP: Interior Gateway Routing Protocol (Cisco proprietary)

# Autonomous Systems





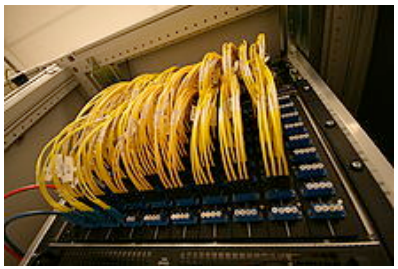

# Interconnection of AS




70



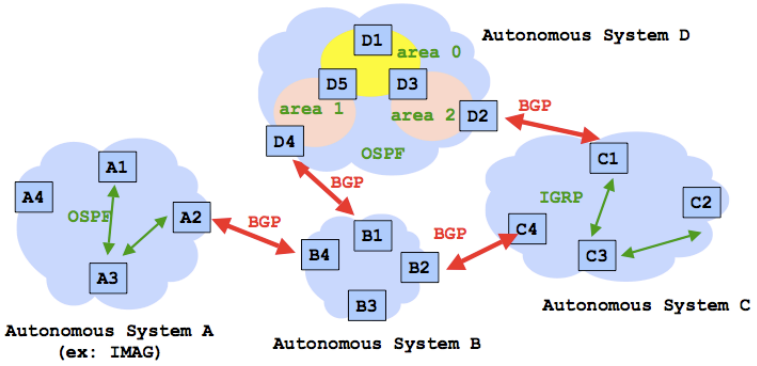
# Example of IXP



London Internet Exchange (LINX )



# Hierarchical Routing




- Hierarchical routing protocols
  - internal (RIP, OSPF, EIGRP)
  - external (BGP)

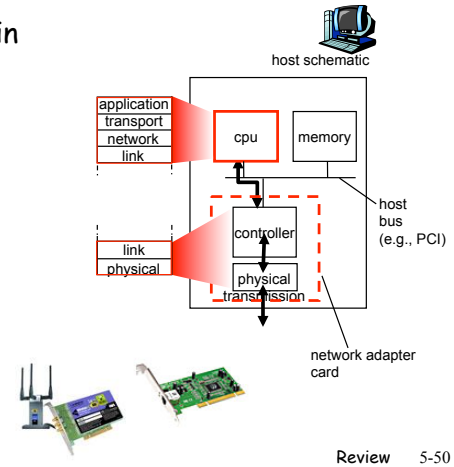
# The Data Link Layer

- Principles behind data link layer services:
  - error detection, correction
  - sharing a broadcast channel: multiple access
  - link layer addressing
  - reliable data transfer, flow control: **done!**
- link layer technologies



# Where is the link layer implemented?

- in each and every host
- link layer implemented in "adaptor" (aka **network interface card** NIC)
  - Ethernet card, PCMCI card, 802.11 card
  - implements link, physical layer
- attaches into host's system buses
- combination of hardware, software, firmware






# Multiple Access Links and Protocols

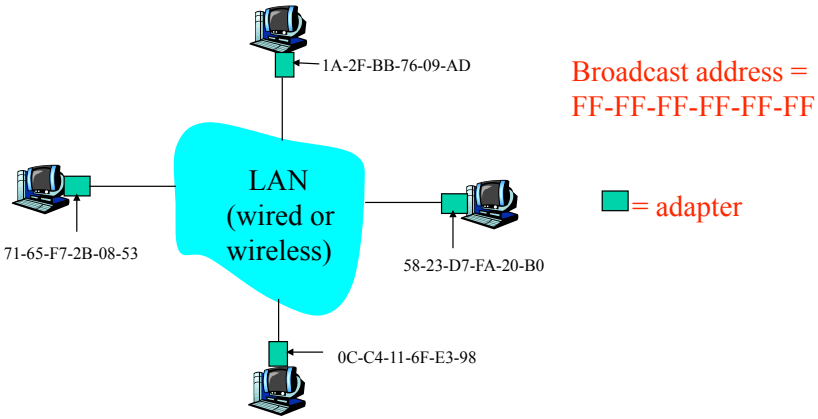
Two types of "links":

- point-to-point
  - PPP for dial-up access
  - point-to-point link between Ethernet switch and host
- broadcast (shared wire or medium)
  - old-fashioned Ethernet
  - upstream HFC
  - 802.11 wireless LAN




# LAN Addresses and ARP

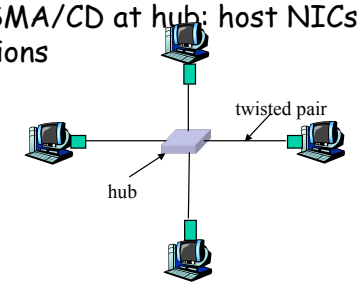
Each adapter on LAN has unique LAN address



# Hubs

… physical-layer ("dumb") repeaters:

- bits coming in one link go out **all** other links at same rate
- all nodes connected to hub can collide with one another
- no frame buffering
- no CSMA/CD at hub: host NICs detect collisions





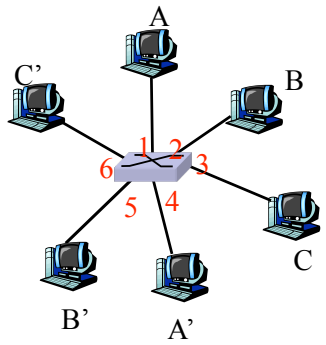

# Switch

- link-layer device: smarter than hubs, take **active** role
  - store, forward Ethernet frames
  - examine incoming frame's MAC address, selectively forward frame to one-or-more outgoing links when frame is to be forwarded on segment, uses CSMA/CD to access segment
- **transparent**
  - hosts are unaware of presence of switches
- **plug-and-play, self-learning**
  - switches do not need to be configured



# Switch: allows **multiple** simultaneous transmissions

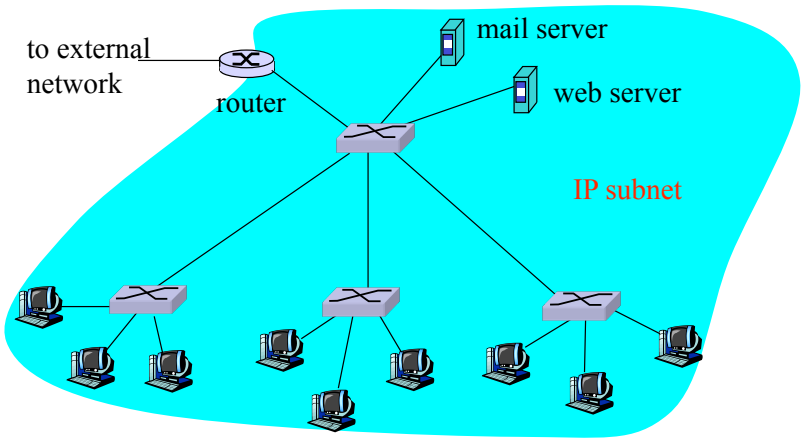
- hosts have dedicated, direct connection to switch
- switches buffer packets
- Ethernet protocol used on **each** incoming link, but no collisions; full duplex
  - each link is its own collision domain
- **switching:** A-to-A' and B-to-B' simultaneously, without collisions
  - not possible with dumb hub




switch with six interfaces (1,2,3,4,5,6)

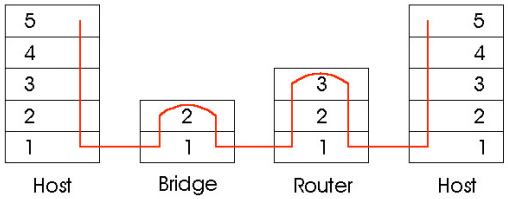


# Institutional network



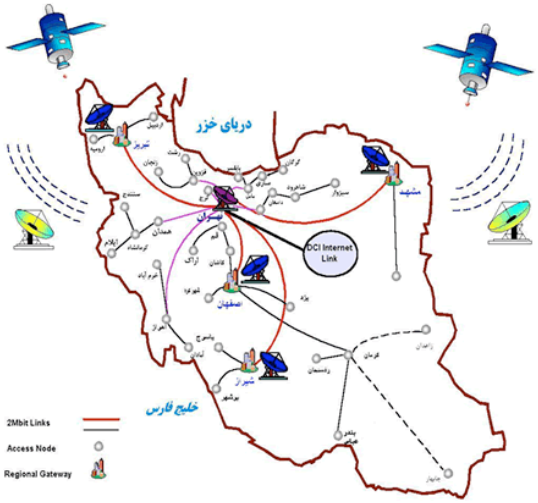
# Switches vs. Routers

- both store-and-forward devices
  - routers: network layer devices (examine network layer headers)
  - switches are link layer devices
- routers maintain routing tables, implement routing algorithms
- switches maintain switch tables, implement filtering, learning algorithms





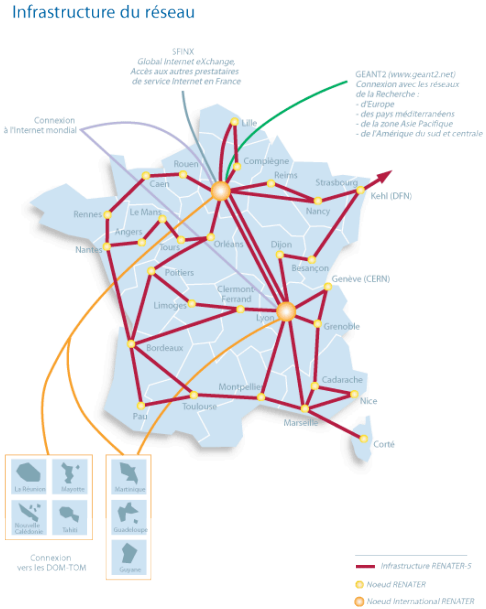

# Scientific Infra. Of Iran

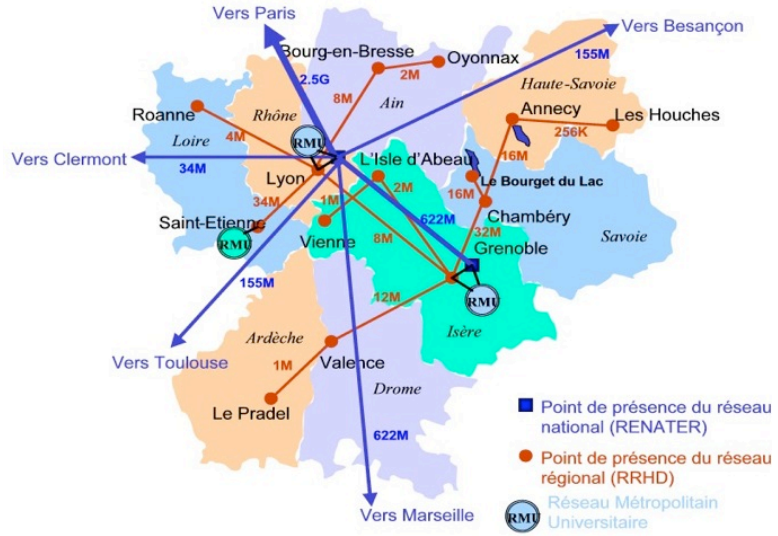




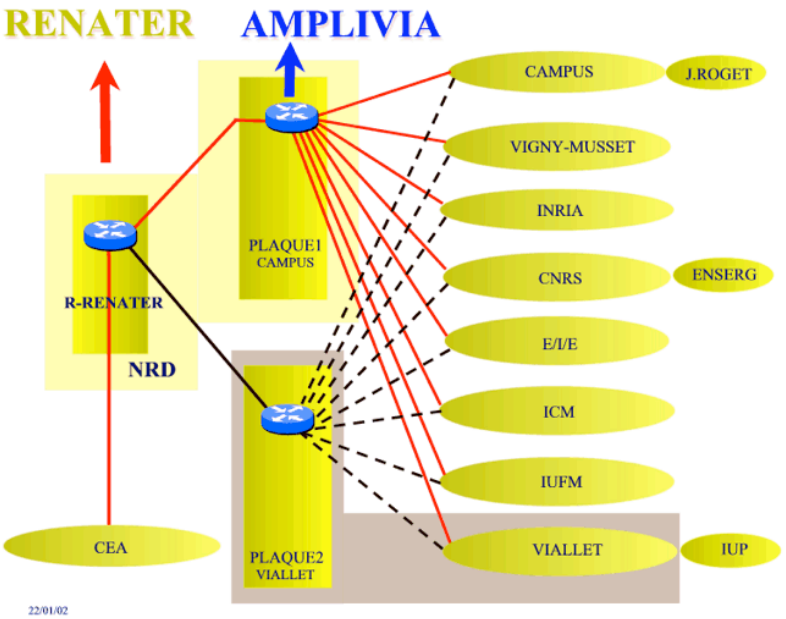




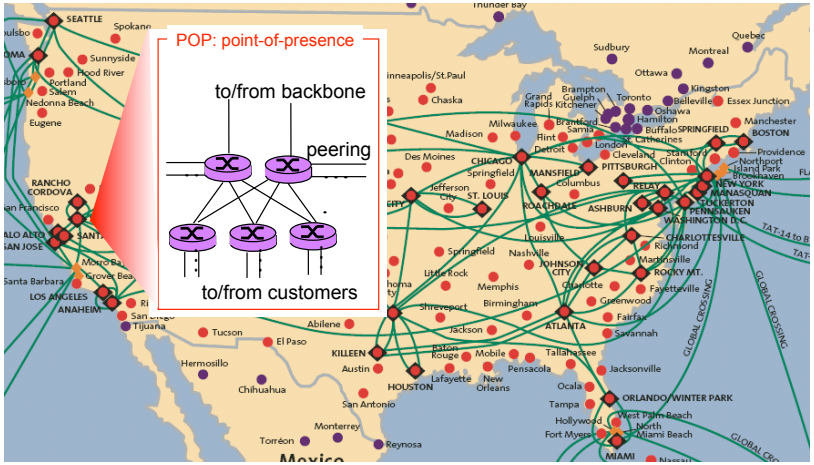







# Tier-1 ISP: e.g., Sprint






# Contents

- Introduction
  - TCP/IP model
- Interconnection Layer 2
  - VLANs and bridges, spanning tree protocol
- Interconnection Layer 3
  - IPv6
  - Routing (RIP, OSPF, BGP)
- Congestion control
- Quality of service

## Contents

- MPLS, multicast
- Mobility
- Network management
- …



## Course Grade

- Assignments
  - Including simulations, hands-on assignments
- Research Project
  - Choose papers among highly-cited recent published papers in wireless domain.
- Mid-term exam
- Final exam



## Reference Materials

- Book Chapters
  - Based on subject
- Research Papers
- Standards



## Backup Slides

# Residential access: cable modems

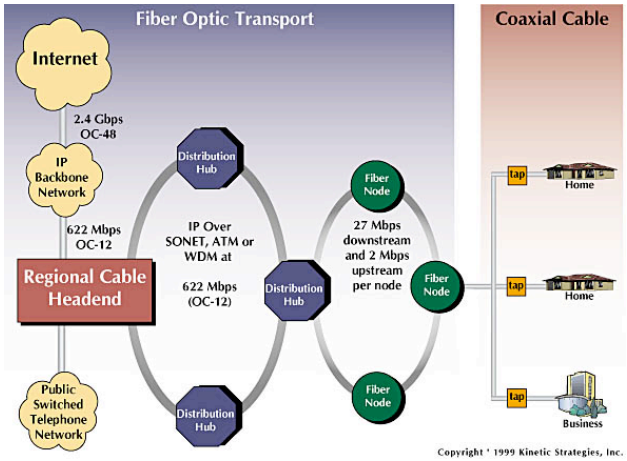



Diagram: http://www.cabledatacomnews.com/cmic/diagram.html