CC 431 Network (1), Fall 2012

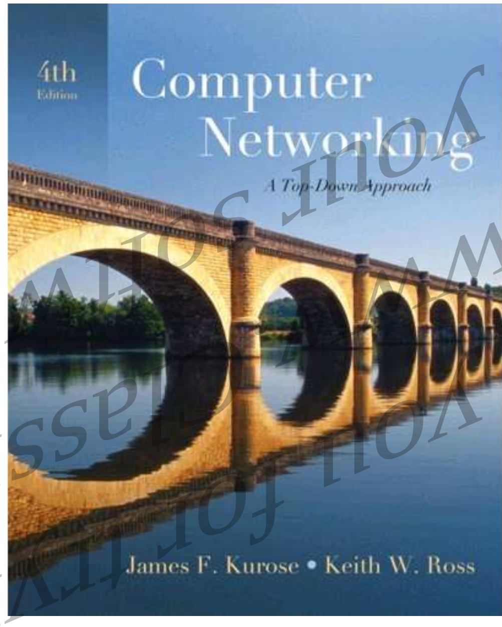
Instructor: Dr. eng Abdulhalim Dandoush

Adandoush at gmail.com

Adandoush.com

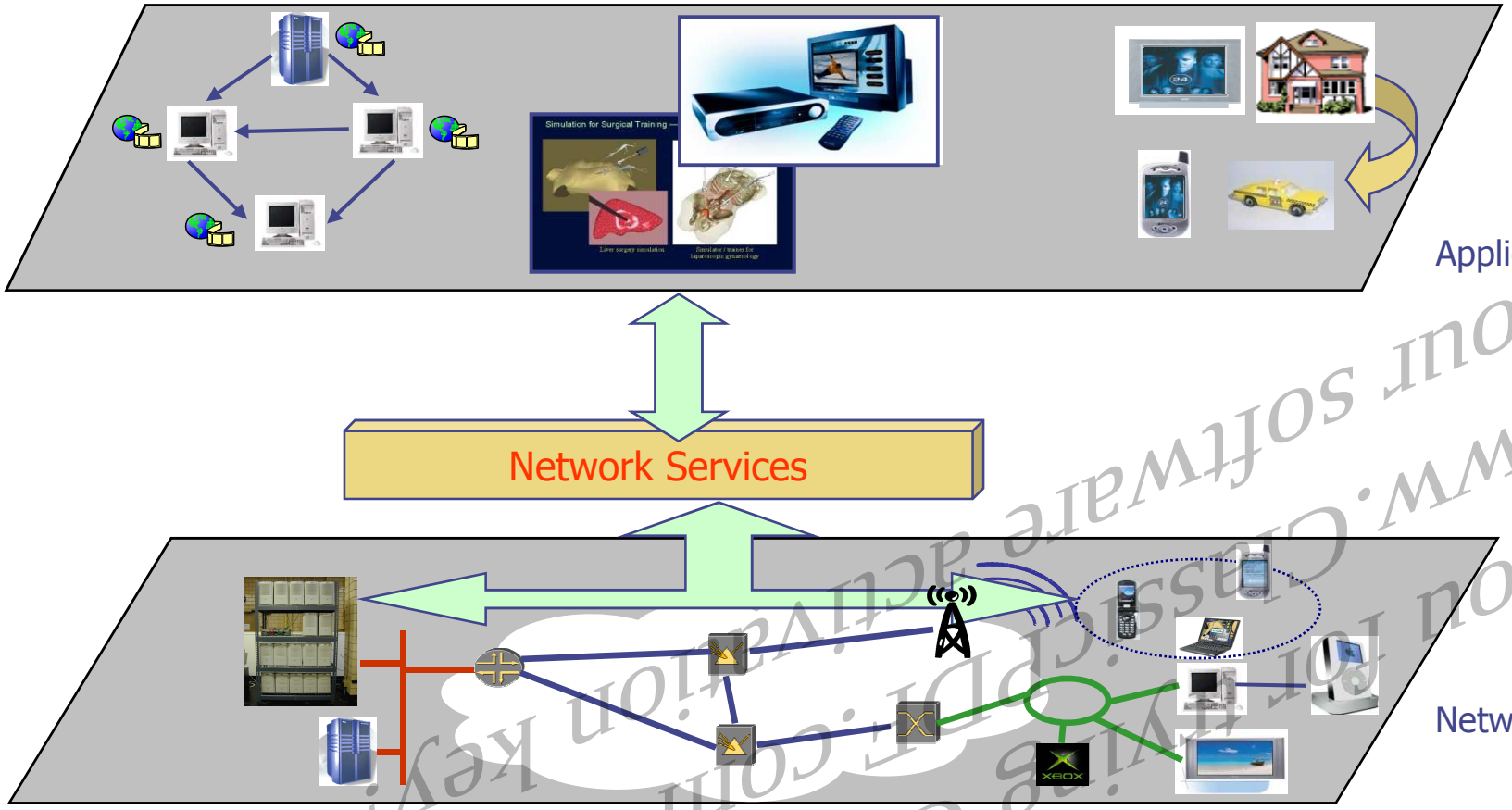
A special acknowledge goes to J.F Kurose and K.W. Ross

Some of the slides used in this lecture are adapted from their original slides that accompany the book “Computer Networking, A Top-Down Approach”



Course Scope

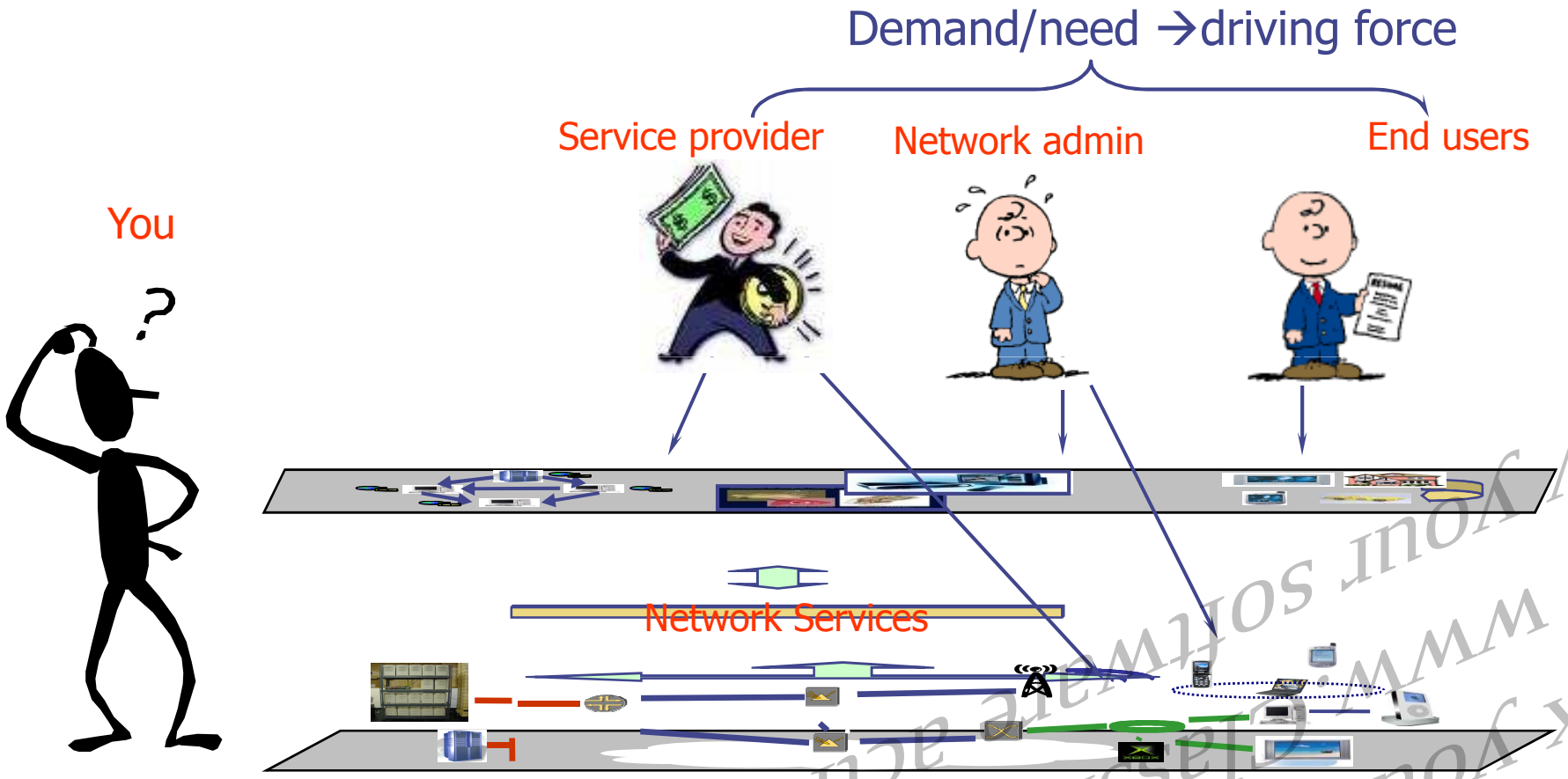
- All about "Network"
  - Network itself: Local Area Network → Internet
  - Networking Applications/Services



Course Focus

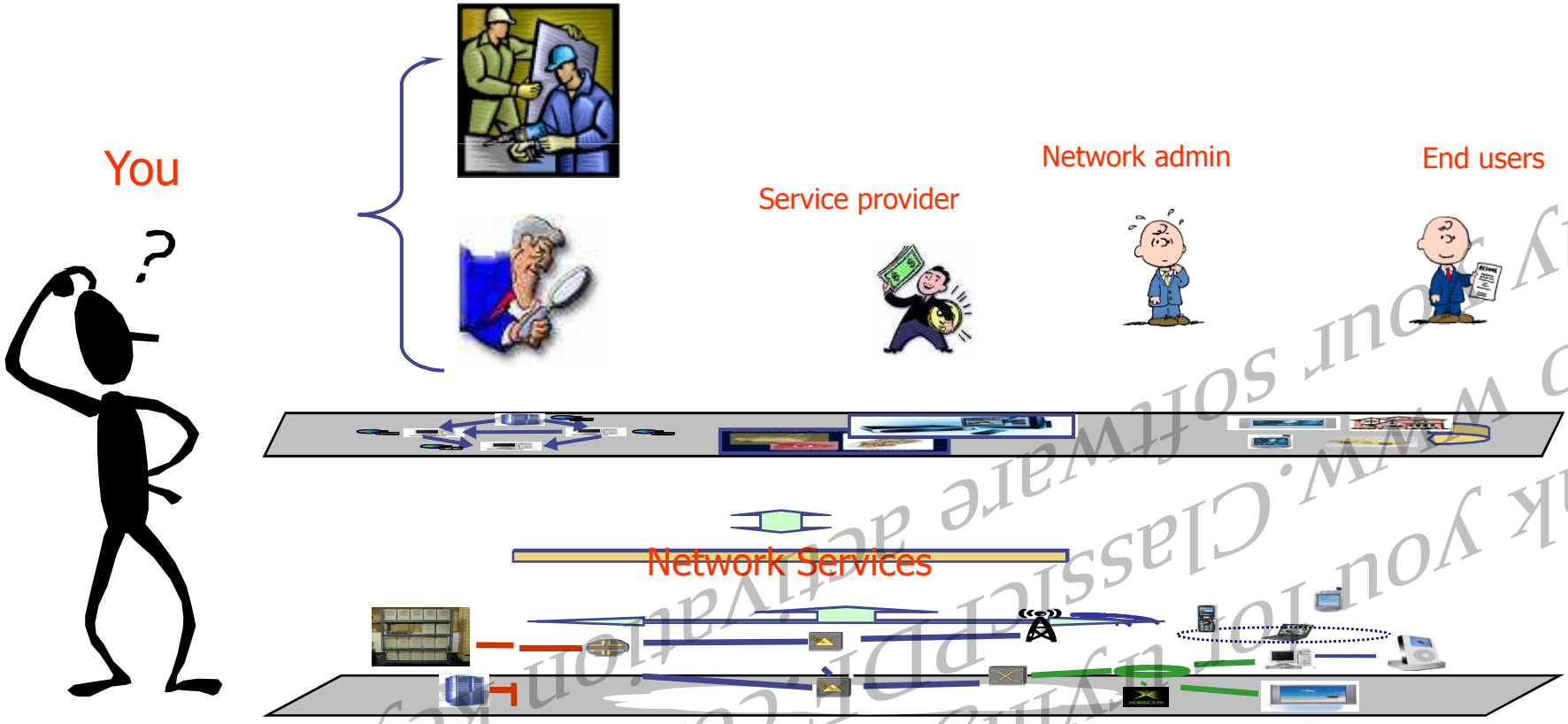
- Basic: first basic class in computer networks
  - network services and overlay topology for Applications
  - Service multiplexing, Traffic Control
  - Addressing, Routing and forwarding
  - Network topology, branching LAN to Internet
  - Network security intro
- Principle + Practice (half and half)
  - Principle – how the network is built and why it is built this way
  - Practice – lots of hands-on experience – what I can do on a network?

Different Perspectives of a Network



Different Perspectives of a Network

- Builder (developer) or Analyst



What you will learn from this course

- Network design/implementation/deployment
- How a network (e.g. the one at Internet Scale) is built
  - Architecture (Layer); protocol; Algorithm
- Why it is built this way? Is it good/the best to build it this way? What if I build it?
- Network Analysis
- How can I understand the behavior of Internet?!
- How to use Network Application?
  - Service (programming interface)/performance

Course Topics

- Overview
  - Network protocol stack (bottom up)
  - Performance measurement
- Application Layer
  - HTTP/DNS
- Transport Layer
  - UDP/TCP
  - Congestion Control
- Network Layer
  - Algorithm + Protocol
  - Multicast
- Link Layer
- Other Topics
  - Wireless Network
  - QoS

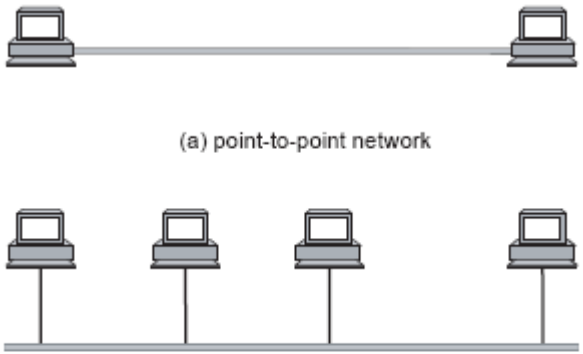
Course Component

- Lecture
  - Slides + white board
  - Take note
  - Online digest/slides
- Participation
  - Discussion
  - Presentation
- Homework
  - many assignments
- Lab/ assignment
  - Work well with your instructors
- Midterm
- Could have a Project !

Grading Policy

- Participation: 10%
- Homework: 10%
- Lab: 20%
- Midterm: 20%
- final:40%

Direct Link Network



(a) point-to-point network

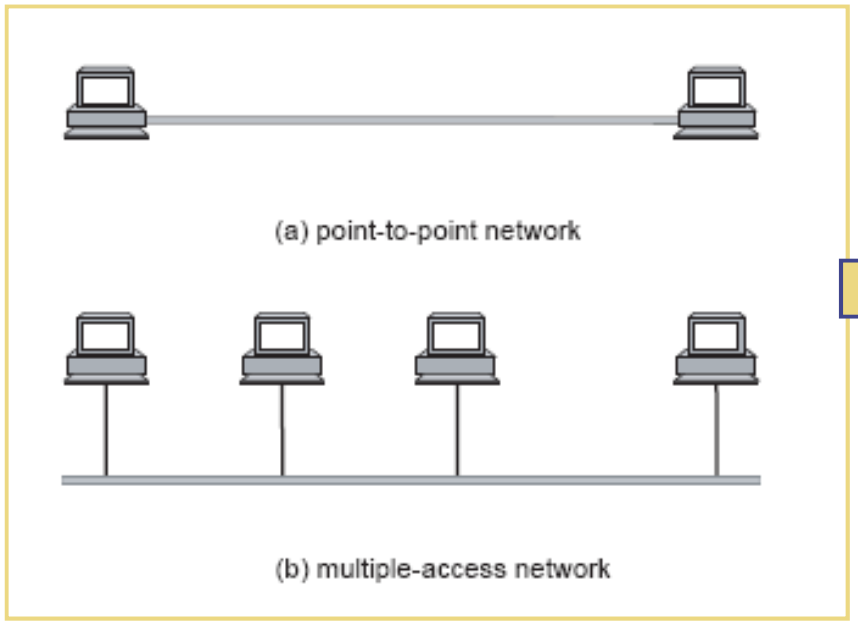
(b) multiple-access network

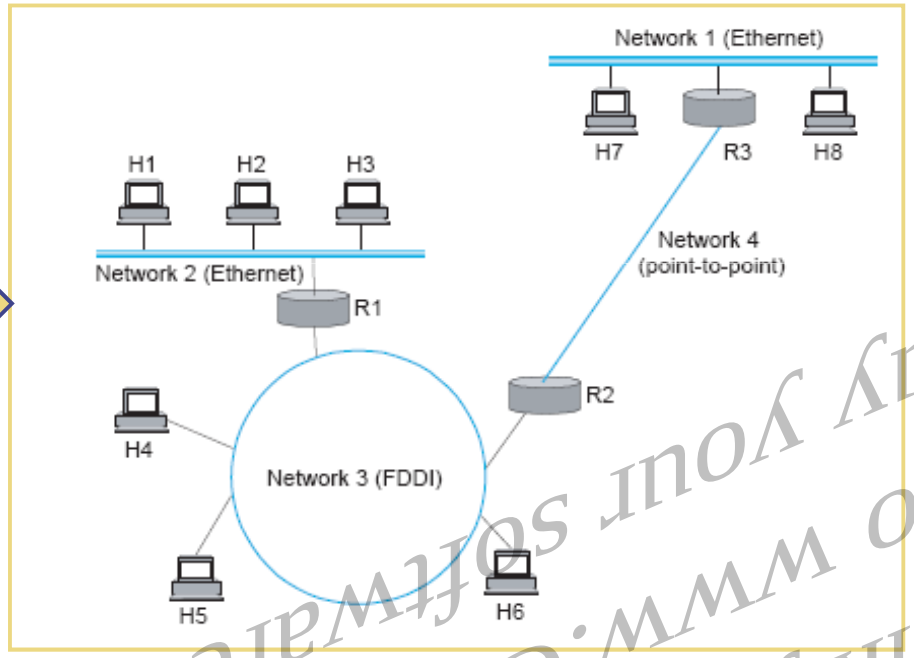
- Point-to-point network
  - Encoding
  - Framing
  - Error detection
  - Reliable delivery

Multiple Access Network

- Problem: When the link is shared by multiple hosts, their accesses to the link need mediation.
- Address
- Media Access Control
  - Ethernet -- CSMA/CD (Carrier Sense Multiple Access / Collision Detection)
  - Wireless LAN – CSMA/CA (Carrier Sense Multiple Access / Collision Avoidance)
    - See file: basic-encoding.pdf

From Direct Link Network To InterNetworking





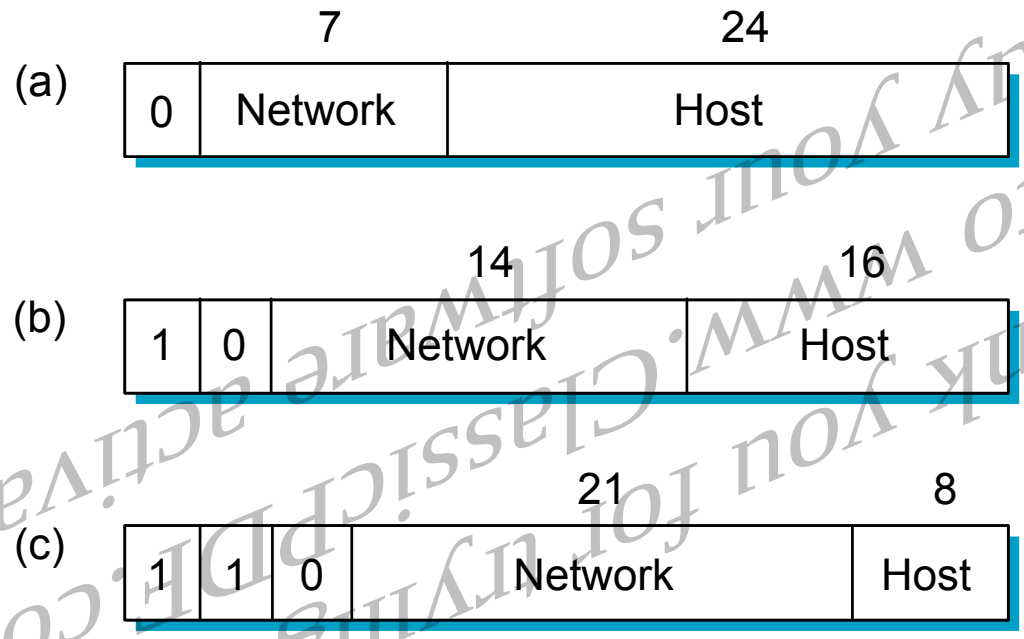
InterNetworking Overview

- Issues
  - Scale
    - the Internet doubled in size each year for 20 years.
  - Heterogeneity
    - internetwork connects networks with different technologies
- Functions
  - Providing suitable names for all hosts → Addressing
    - IP addressing → Hierarchical addressing facilitates scalable networking and routing
  - Building the internetworking infrastructure
    - Nuts and bolts → routers, links, etc.
  - Delivering data → forwarding
    - packet switching → Store and forward → statistical multiplexing enable efficient resource utilization

    [how Internet structure looks like – A reality check]
  - Finding a path → routing
  - Dealing Heterogeneity → Fragmentation and Reassembly

Addressing

- Addressing
  - providing suitable identifiers for all these hosts in internetworks.
- Hierarchical addresses
  - E.g., Surface Mail Address
  - 2301 Vanderbilt Place
    Nashville,
    TN
    USA



What's the Internet: "nuts and bolts" view


PC


server


wireless laptop

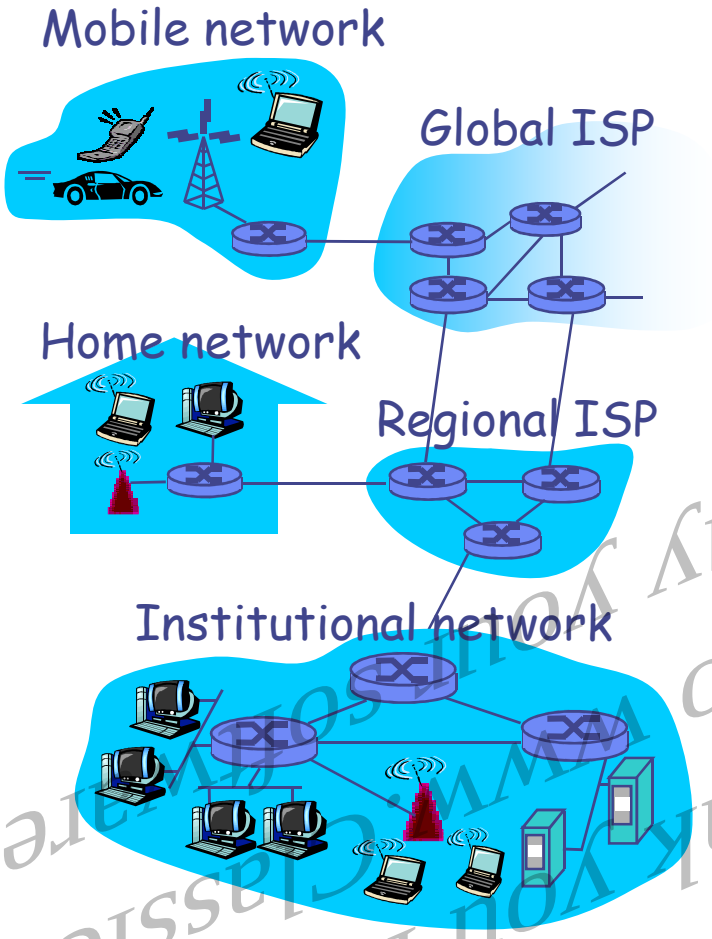

cellular handheld


access points

wired links

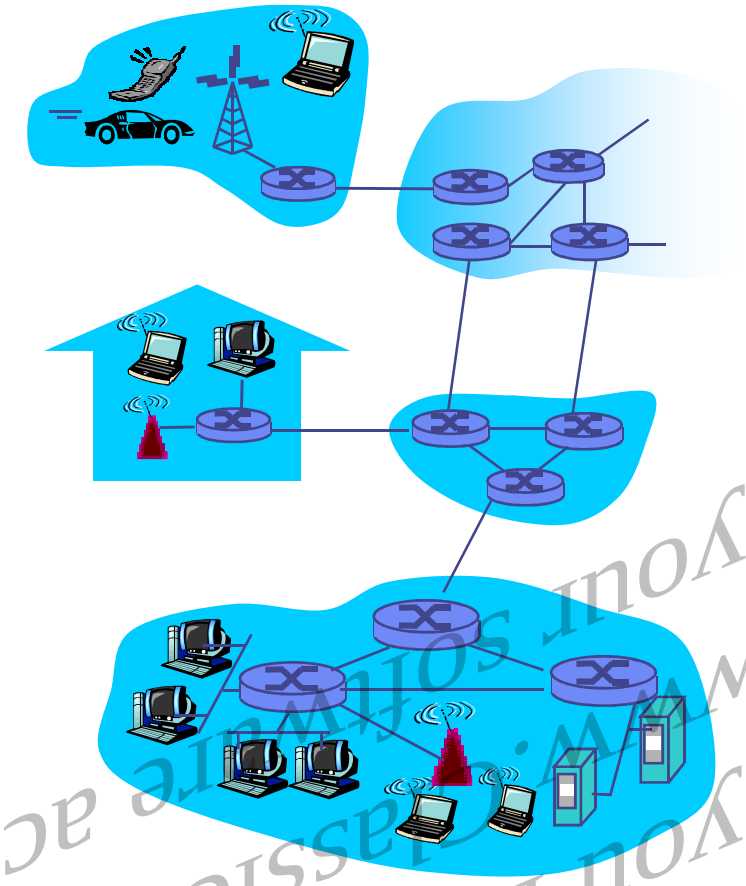

router

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



A closer look at network structure:

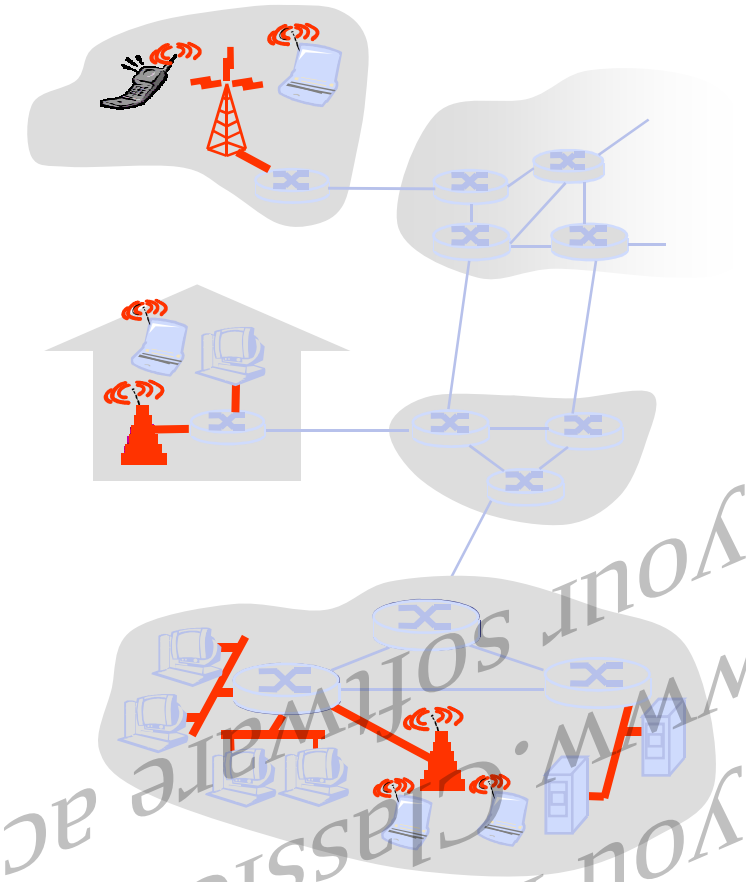
- network edge:
  applications and hosts
  - access networks, physical media: wired, wireless communication links
    - network core:
      - interconnected routers
      - network of networks



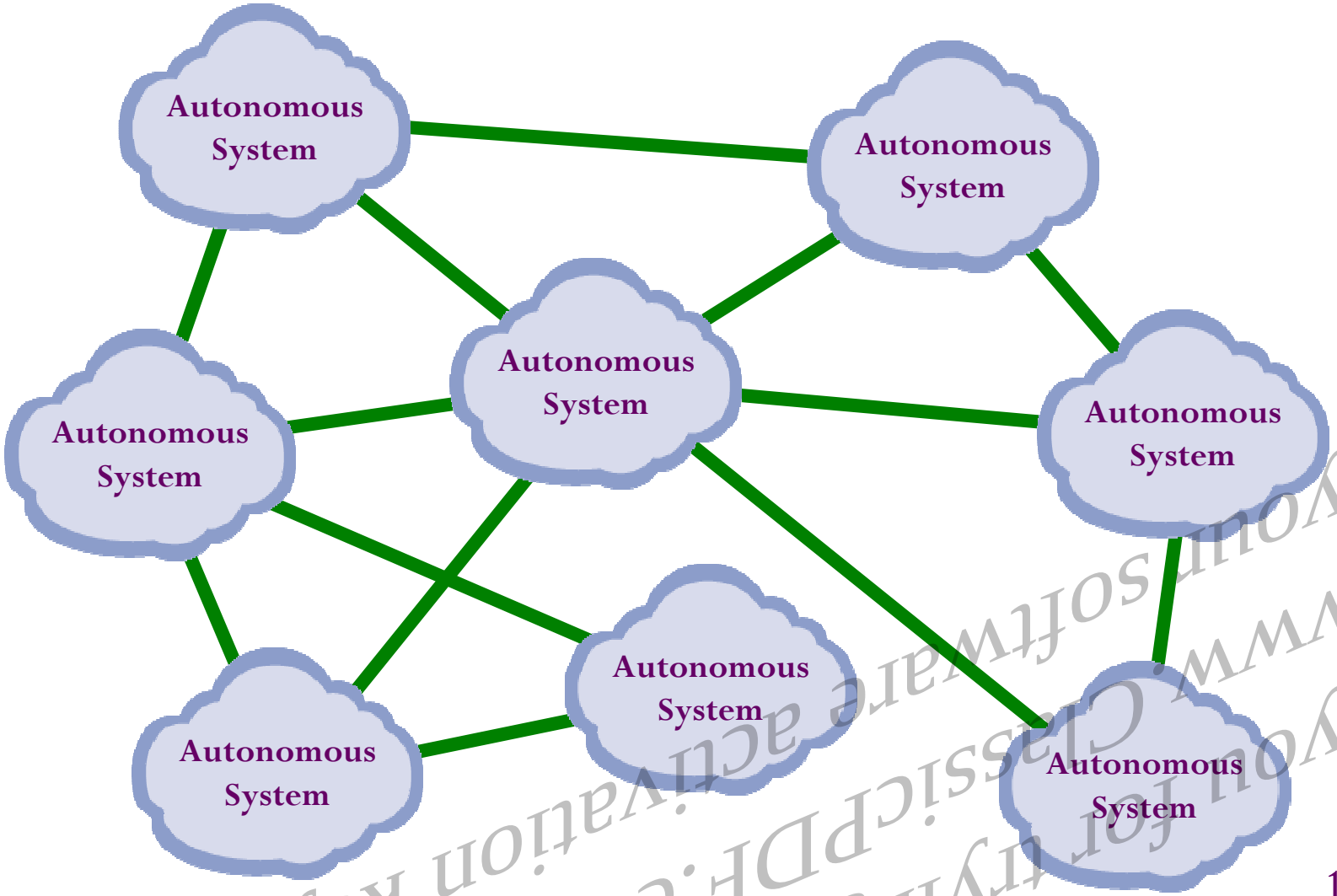
Access networks and physical media

Q: How to connect end systems to edge router?

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

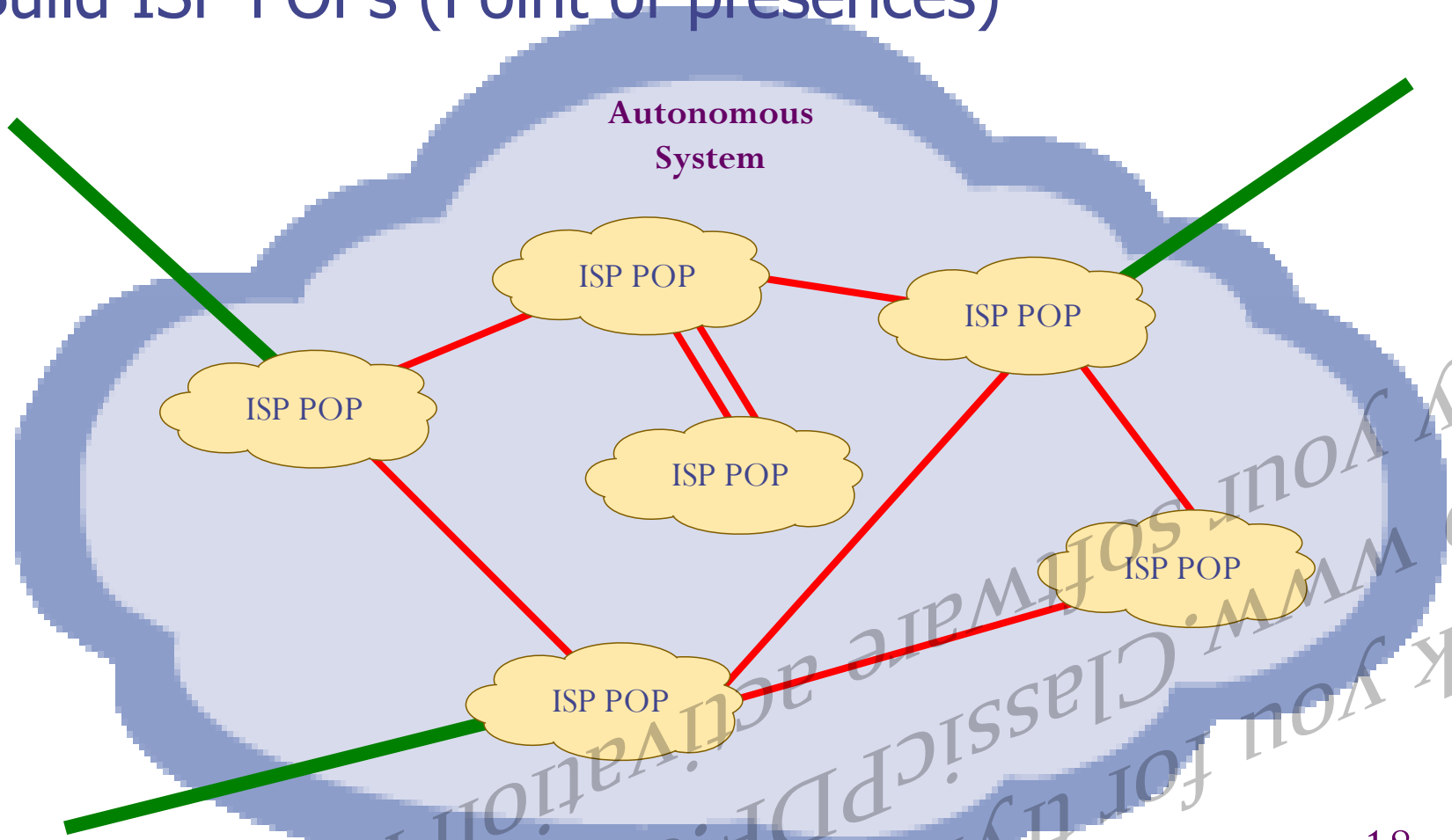


Internet topology



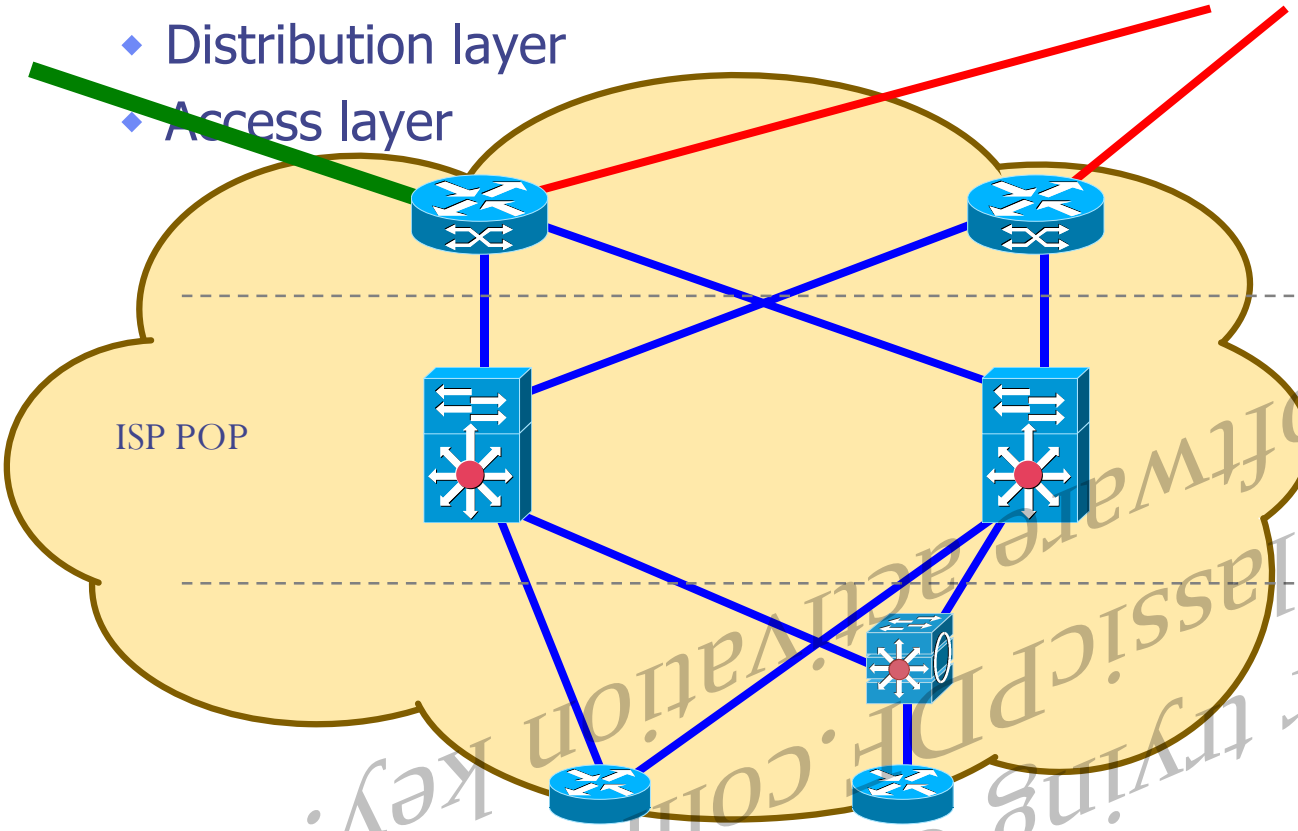
Internet elements

- Build ISP POPs (Point of presences)



Internet elements

- Network topology in ISP POP
  - Three layers architecture
    - Core layer
    - Distribution layer
    - Access layer



Internet elements

- Core router: CSCO XR 12000 series router

Internet elements

- Core router: CSCO CRS-1 Carrier Routing

Internet elements

- Core router: JNPR M-series router

Internet elements

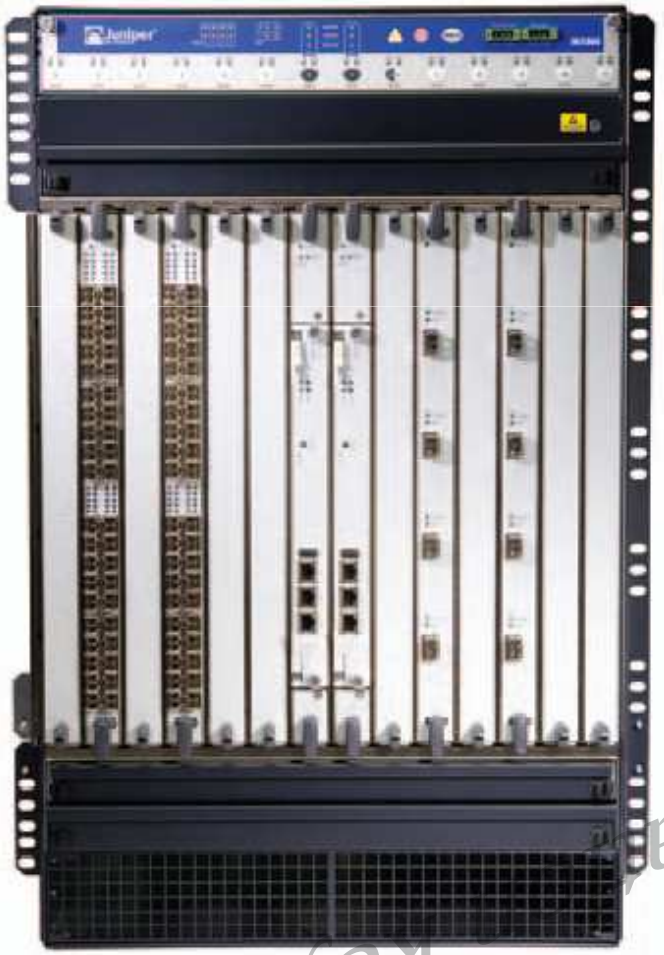
- Core router – JNPR T-series router

Internet elements

- Distribution layer router: CSCO 7600

Internet elements

- Distribution layer router: JNPR MX960



Internet elements

- Distribution layer router: CSCO 6500

Internet elements

- Access layer
  - Face to customers
  - Aggregate many low-speed circuit to one or two high-speed circuit
    - Face to customer: T1, E1, ADSL
    - Connect to distribution layer: FE, GE
  - Use access router or Broadband Remote Access Server (BRAS)
    - Router
      - CSCO 3700, 7200, 7300 series router
      - JNPR M-series router
    - BRAS
      - Redback SmartEdge
      - JNPR E-series BRAS routing platform (ERX)

Internet elements

- Access layer: CSCO 7200 series router

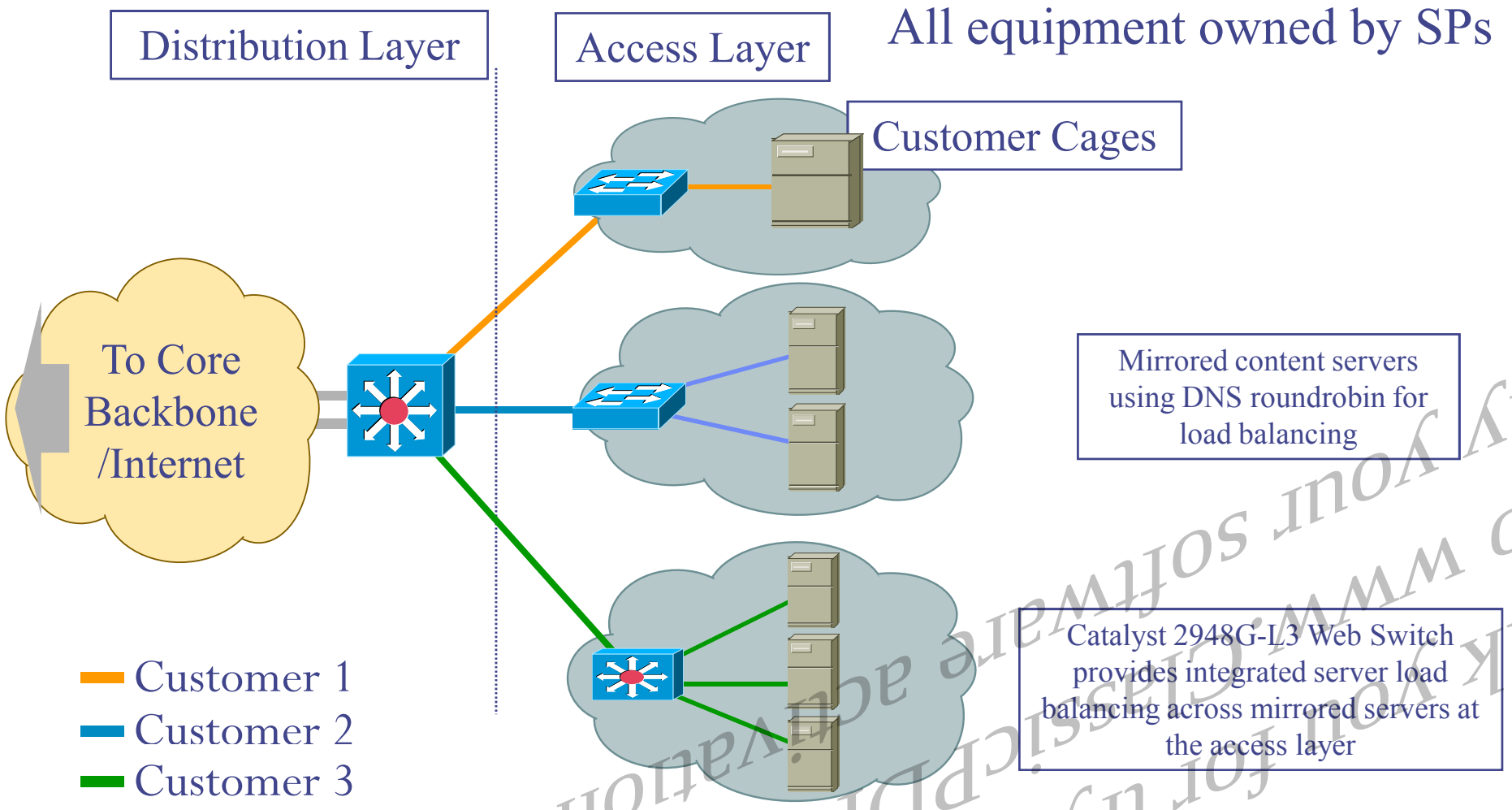
Internet elements

- Access layer: Redback SmartEdge

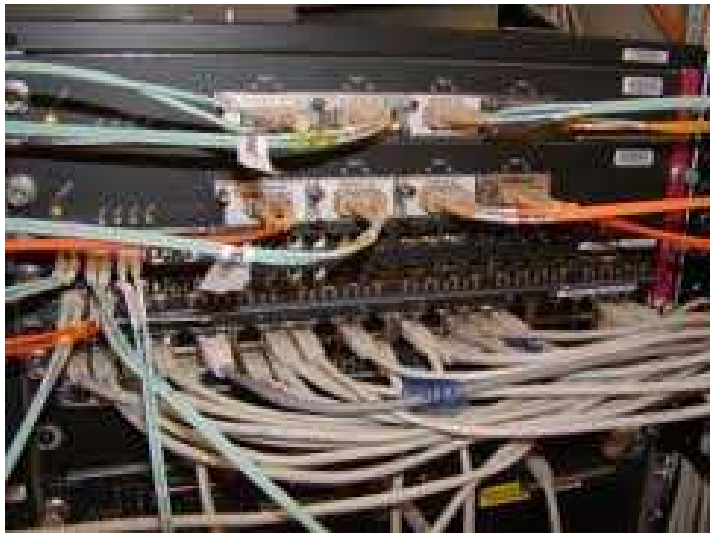
Internet elements

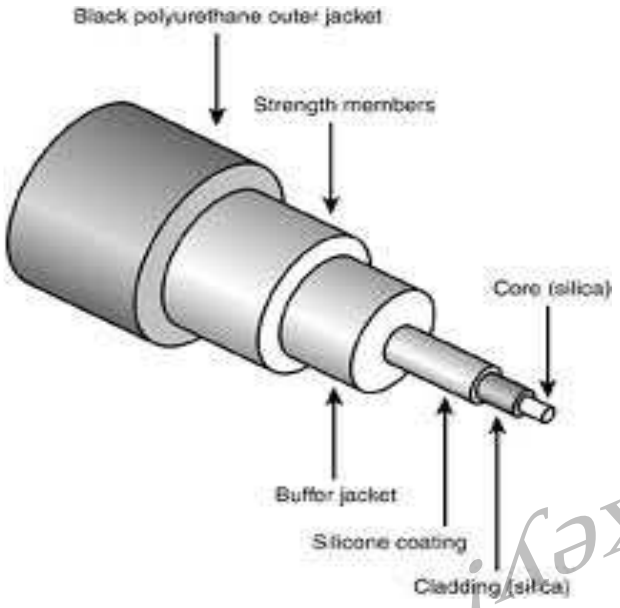
- Access layer: JNPR E-series routing platform

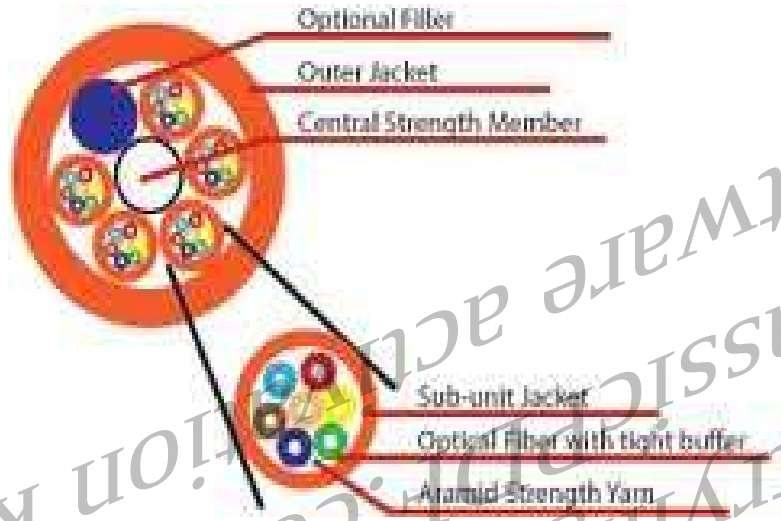
Managed Hosting Model
Basic Web Hosting



Some cables in Core, Dist and Acess layer



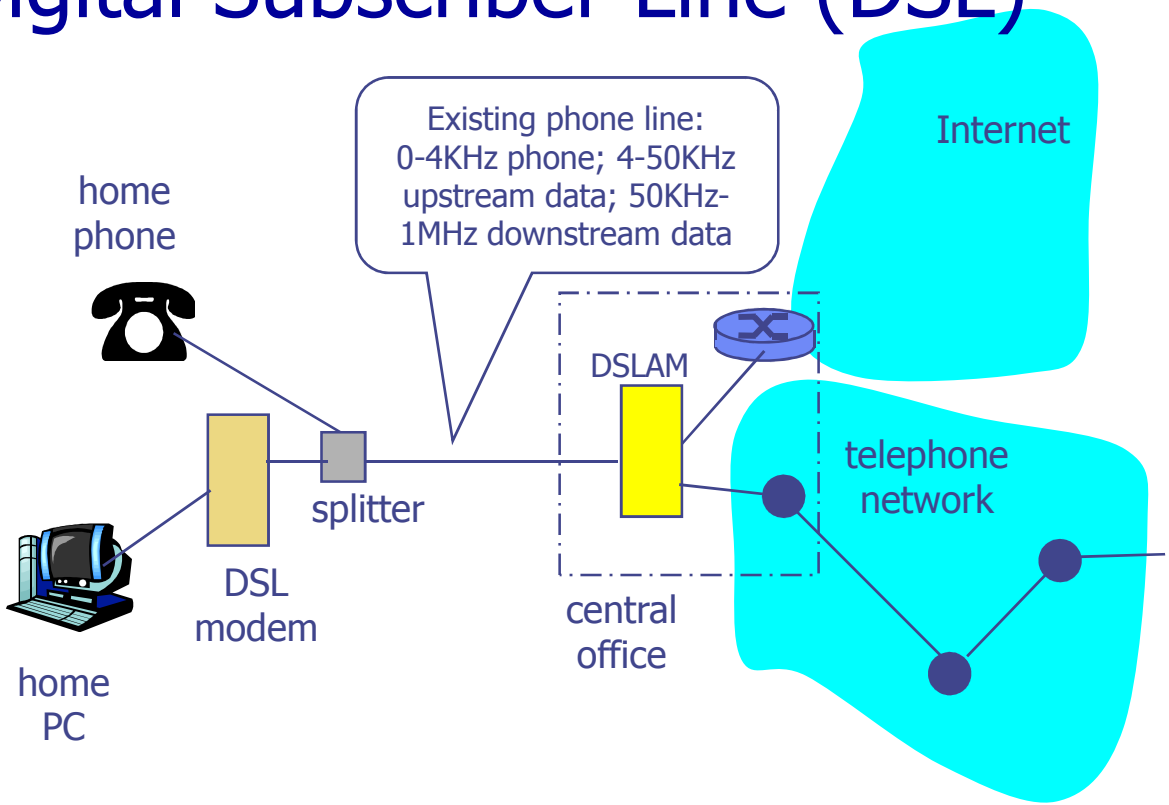




Residential access: cable modems

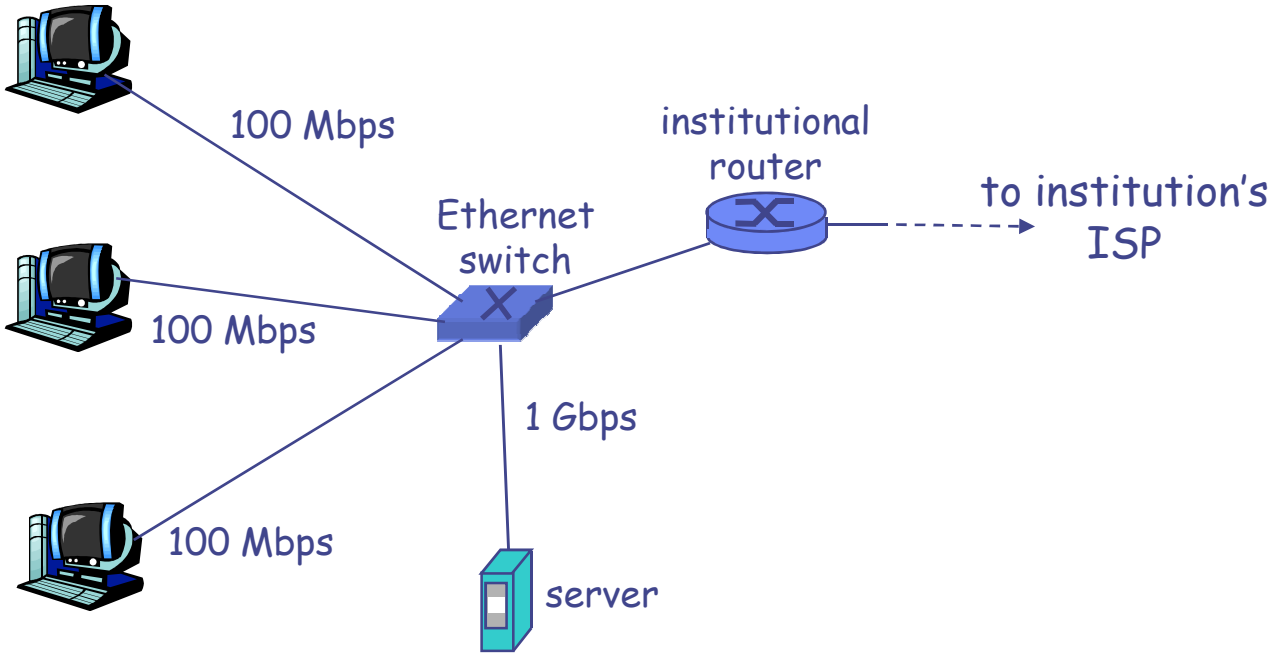
- uses cable TV infrastructure, rather than telephone infrastructure
- HFC: hybrid fiber coax
  - asymmetric: up to 30Mbps downstream, 2 Mbps upstream
- network of cable, fiber attaches homes to ISP router
  - homes share access to router
  - unlike DSL, which has dedicated access

Digital Subscriber Line (DSL)

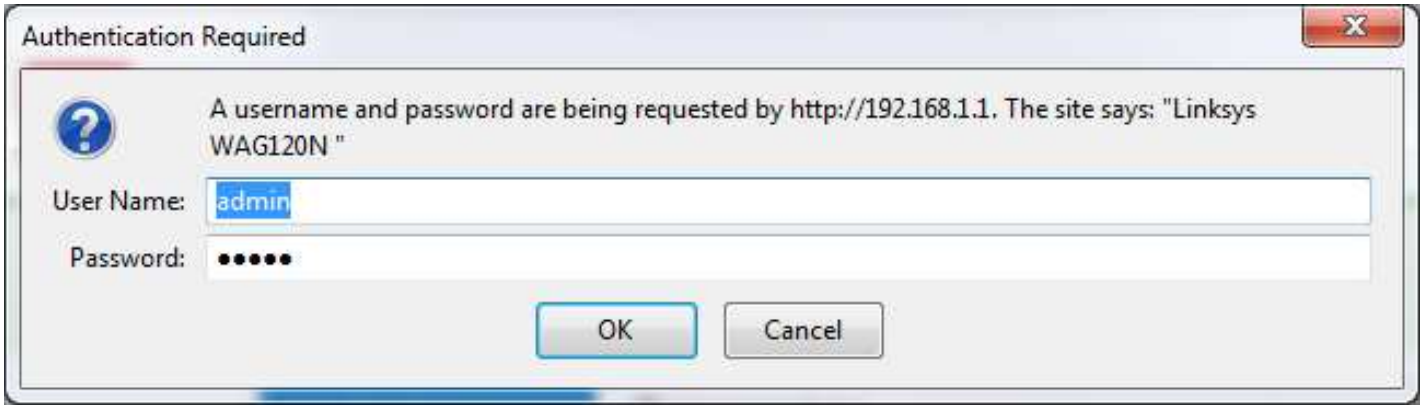


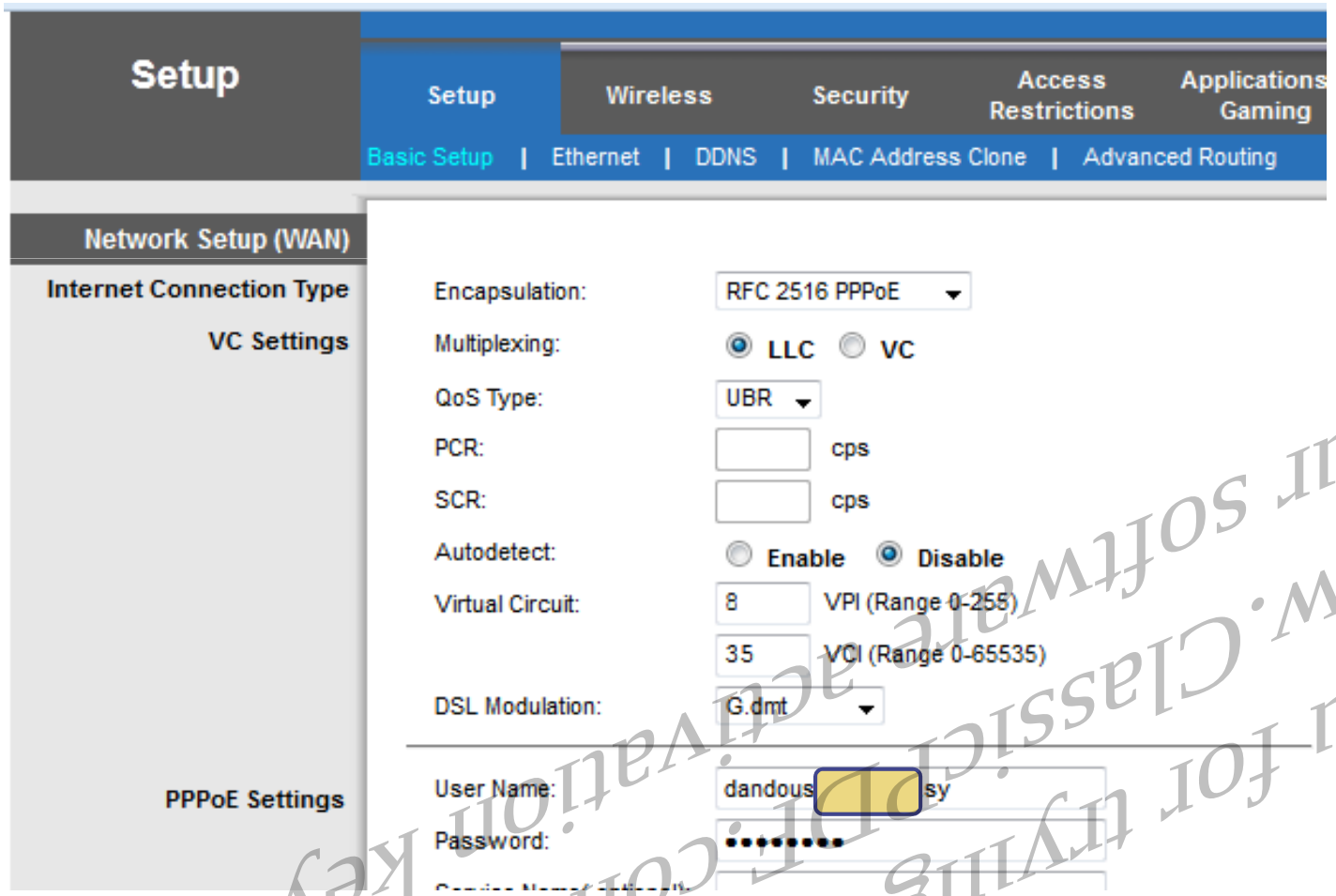
- uses existing telephone infrastructure
- up to 1 Mbps upstream (today typically < 256 kbps)
- up to 8 Mbps downstream (today typically < 1 Mbps)
- dedicated physical line to telephone central office

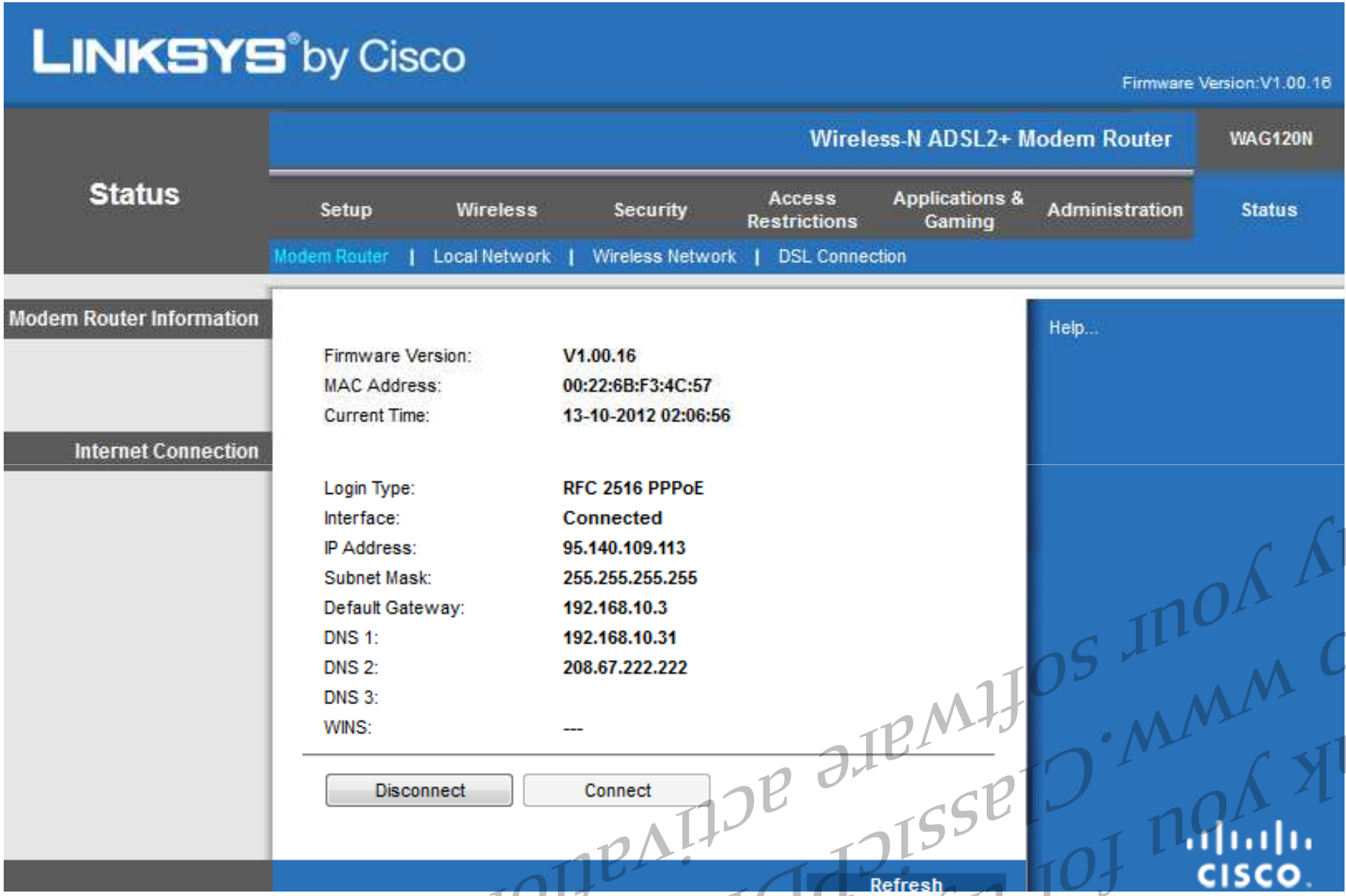
Ethernet Internet access



- typically used in companies, universities, etc
- 10 Mbps, 100Mbps, 1Gbps, 10Gbps Ethernet
- today, end systems typically connect into Ethernet switch

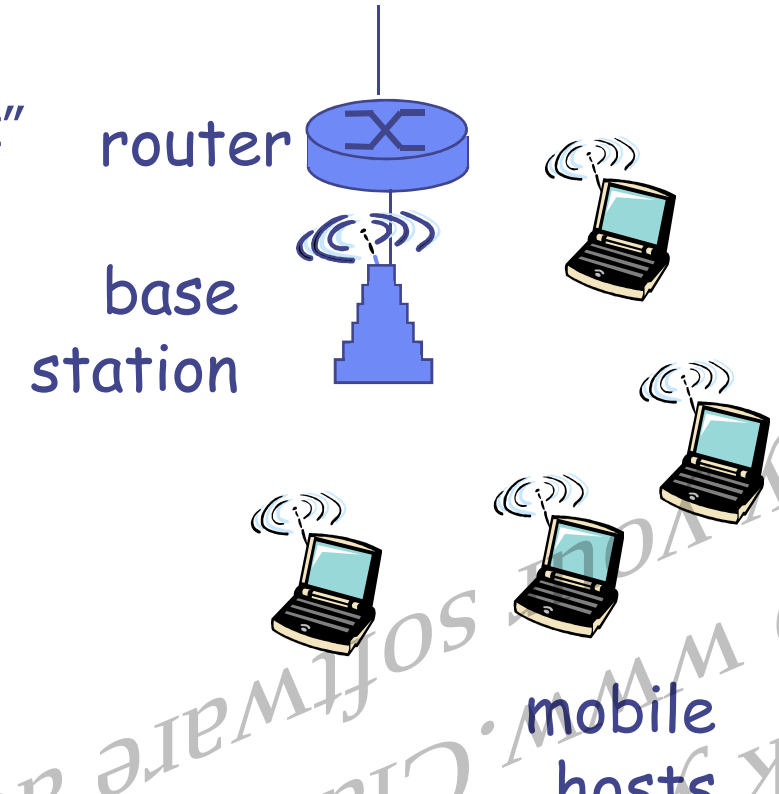


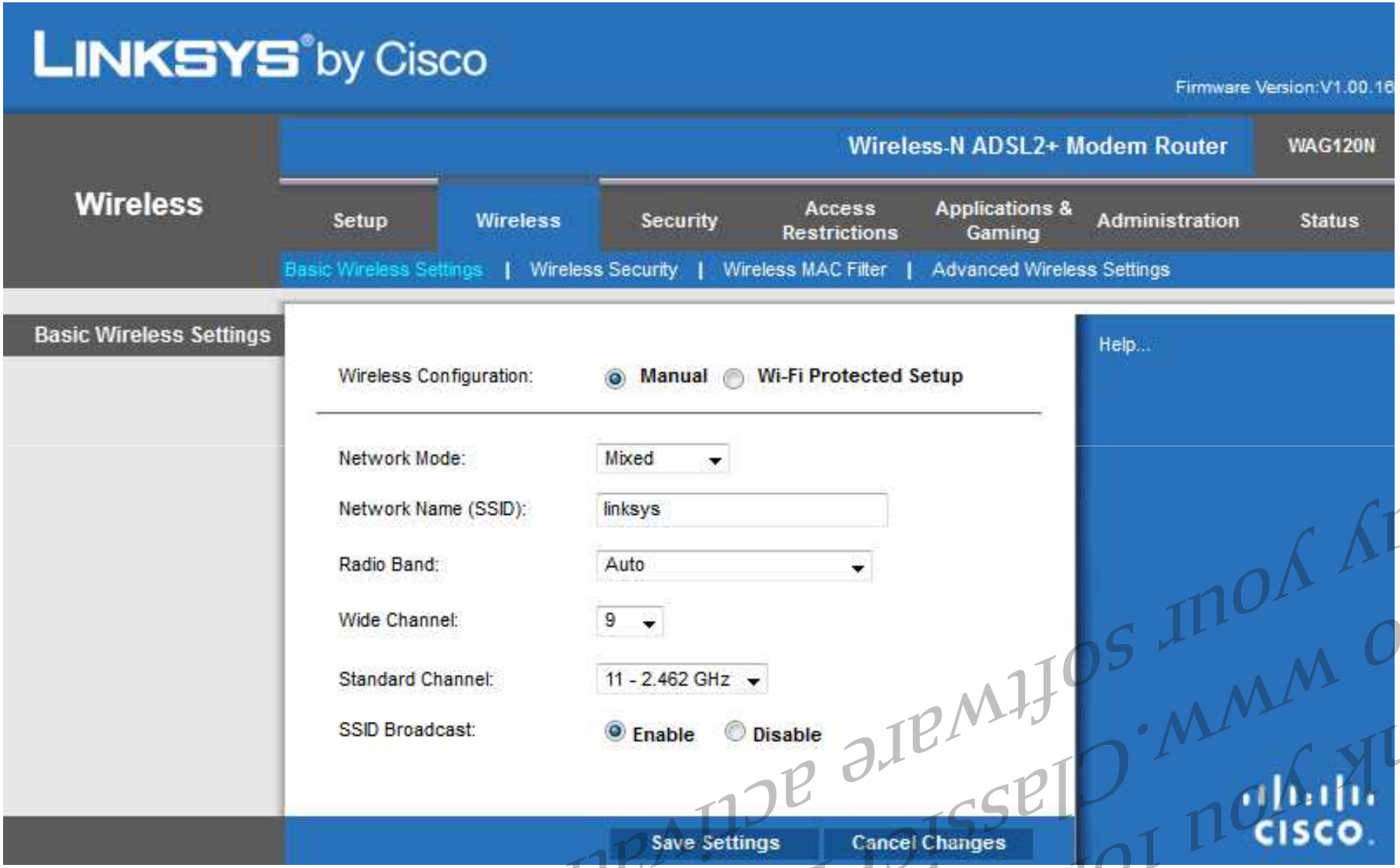


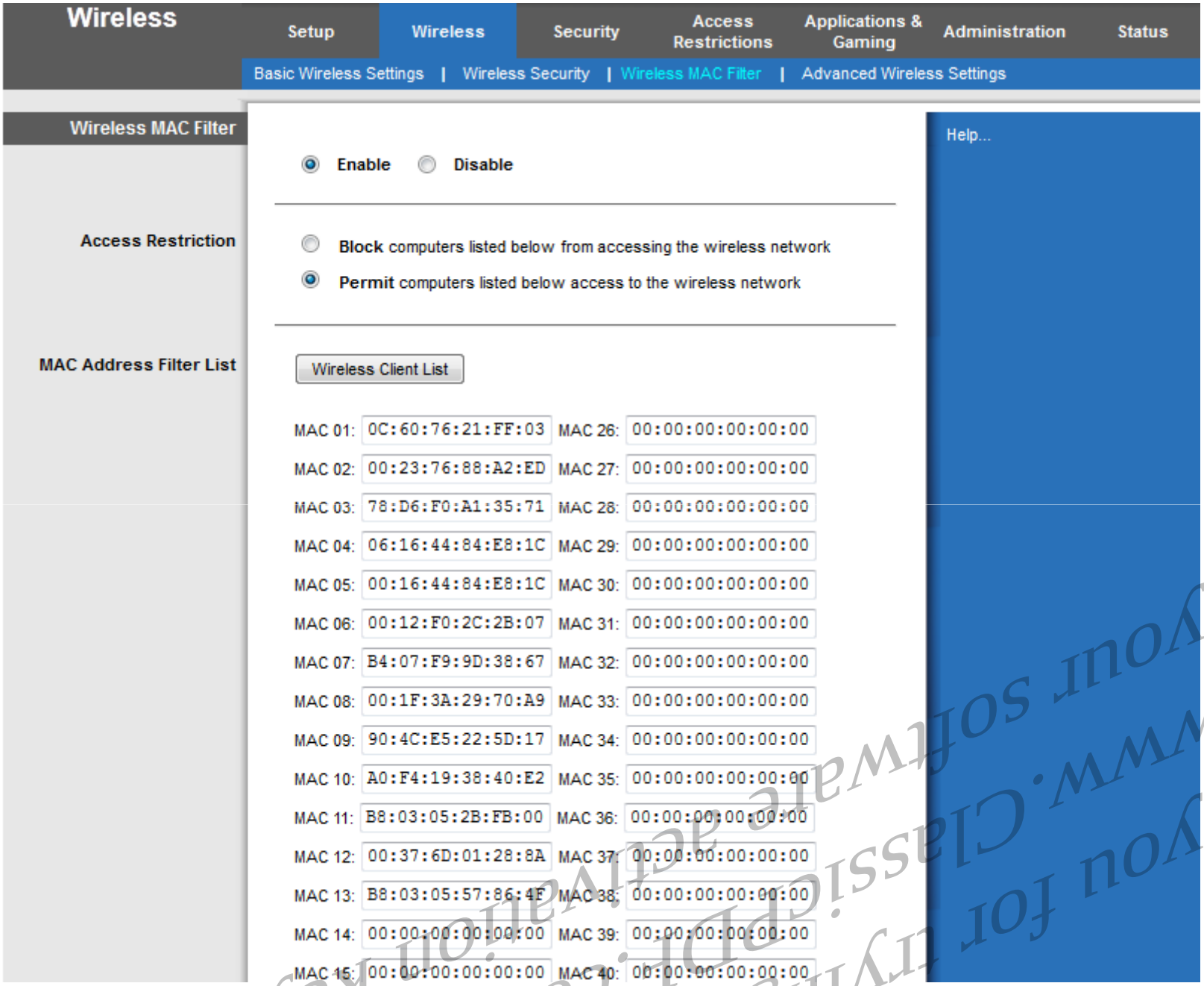


Wireless access networks

- 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

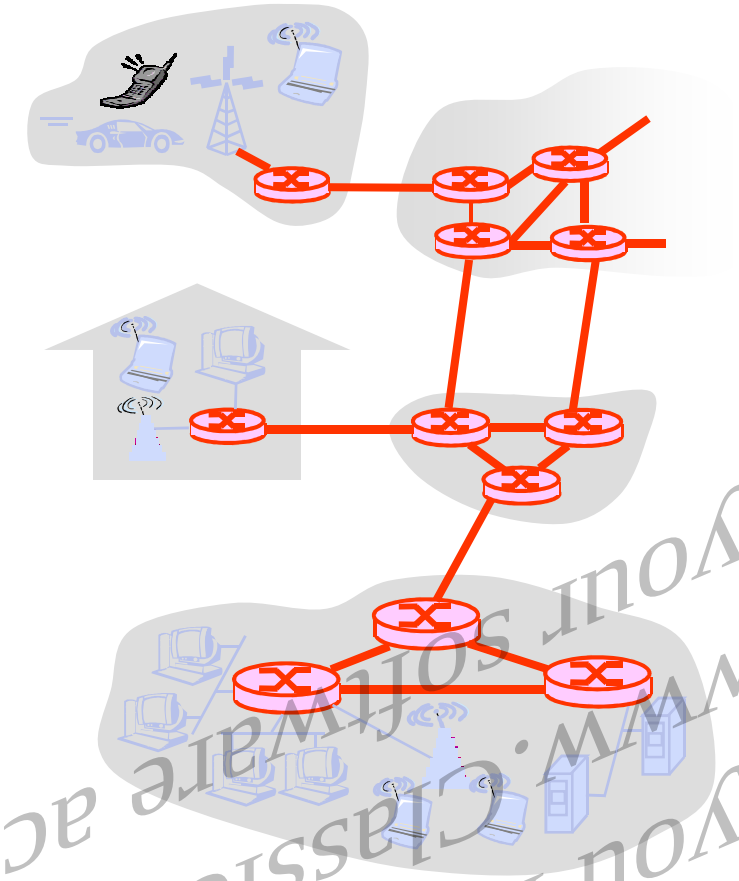






The Network Core

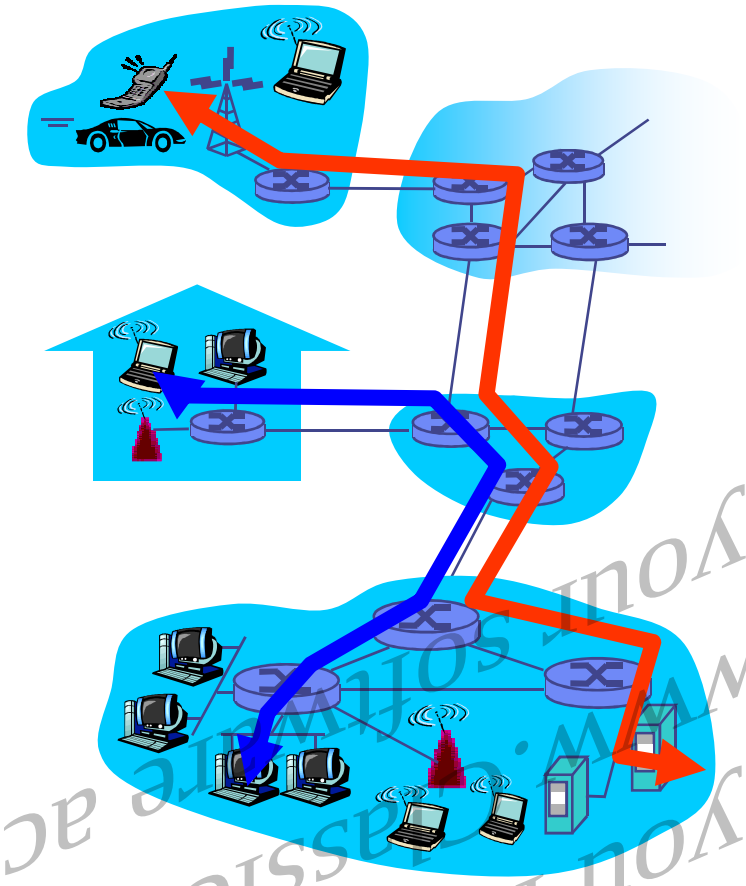
- mesh of interconnected routers
- *the* fundamental question: how is data transferred through net?
  - circuit switching: dedicated circuit per call: telephone net
  - packet-switching: data sent thru net in discrete "chunks"



Network Core: Circuit Switching

end-end resources reserved for "call"

- link bandwidth, switch capacity
- dedicated resources: no sharing
- circuit-like (guaranteed) performance
- call setup required



Network Core: Circuit Switching

network resources (e.g., bandwidth) divided into "pieces"

- pieces allocated to calls
- resource piece *idle* if not used by owning call *(no sharing)*

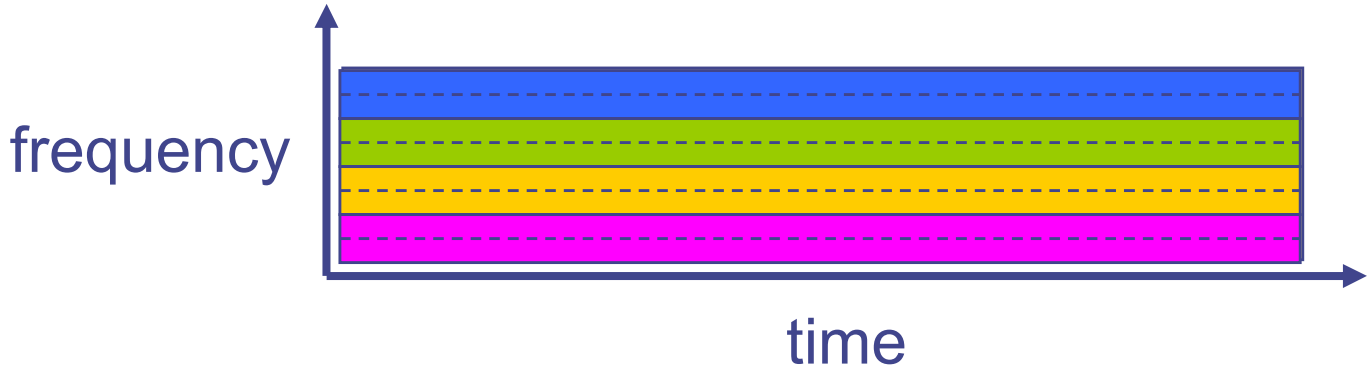
- dividing link bandwidth into "pieces"
  - frequency division
  - time division

Circuit Switching: FDM and TDM

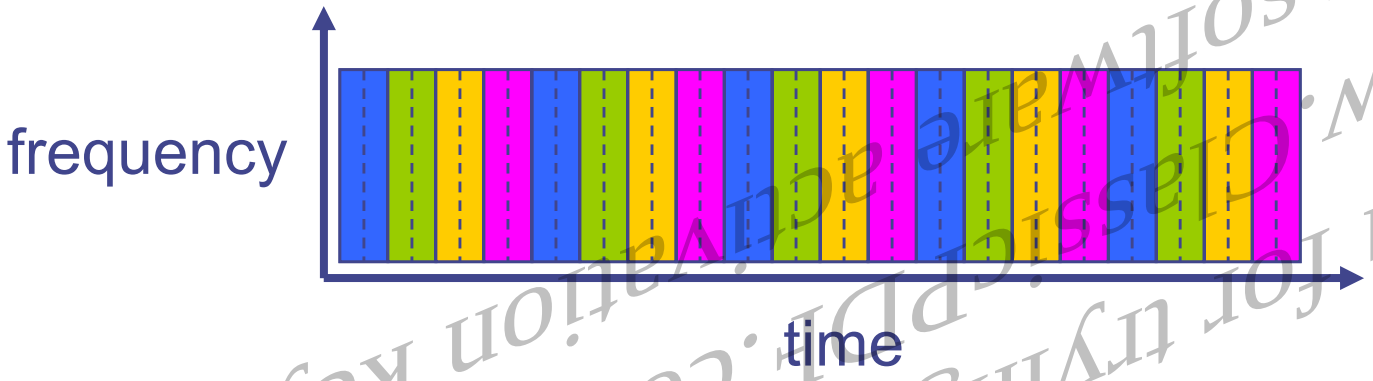
Example:

4 users

FDM



TDM



Network Core: Packet Switching

each end-end data stream divided into *packets*

- user A, B packets *share* network resources
- each packet uses full link bandwidth
- resources used *as needed*

Bandwidth division into "pieces"

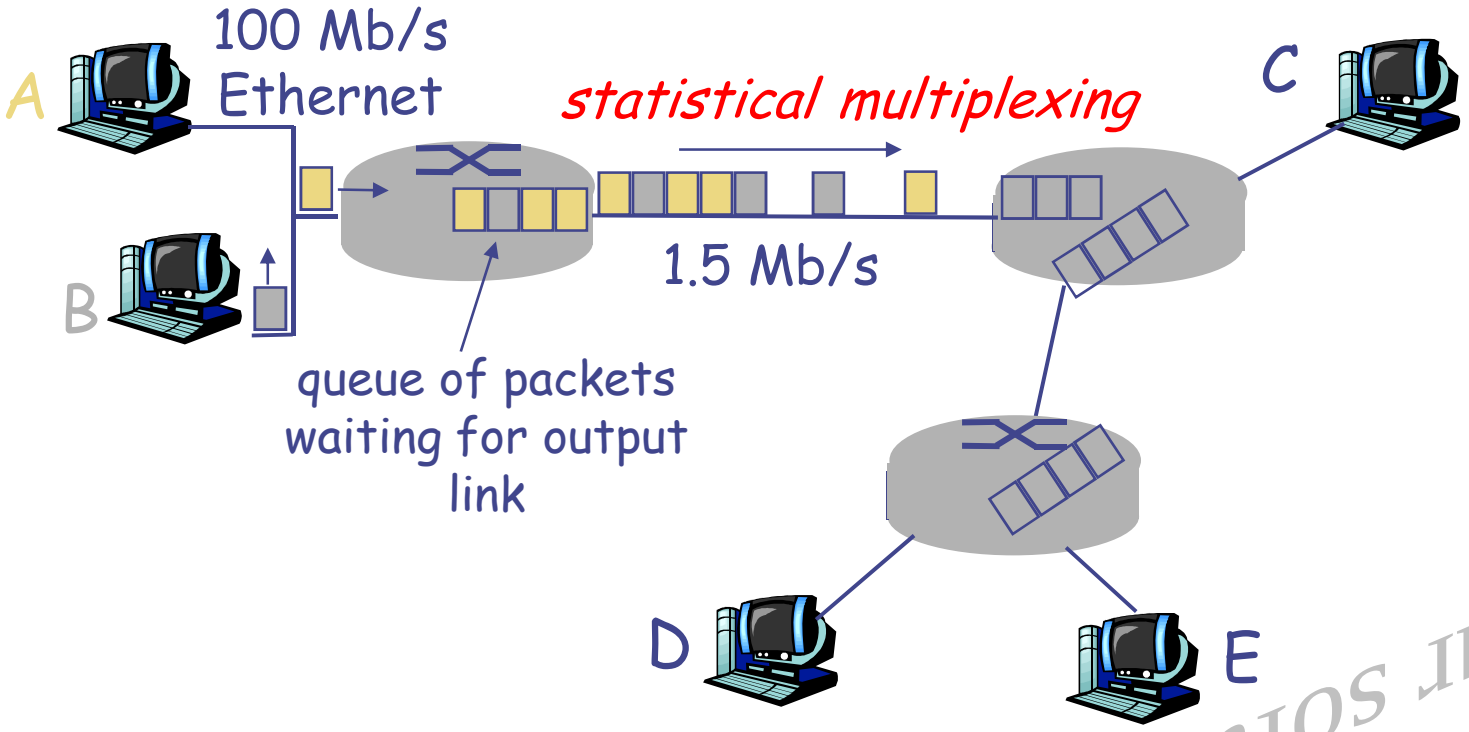
Dedicated allocation

Resource reservation

resource contention:

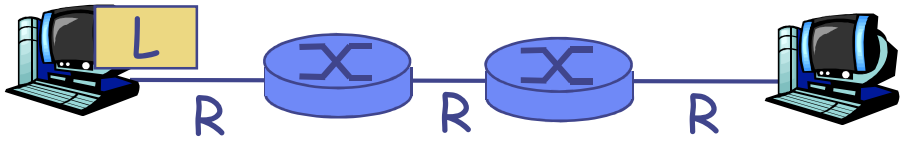
- aggregate resource demand can exceed amount available
- congestion: packets queue, wait for link use
- store and forward: packets move one hop at a time
  - node receives complete packet before forwarding

Packet Switching: Statistical Multiplexing



- sequence of A & B packets has no fixed timing pattern
  - bandwidth shared on demand: *statistical multiplexing*.
- TDM: each host gets same slot in revolving TDM frame.

Packet-switching: store-and-forward



- takes L/R seconds to transmit (push out) packet of L bits on to link at R bps
- *store and forward:* entire packet must arrive at router before it can be transmitted on next link
- delay = 3L/R (assuming zero propagation delay)

more on delay shortly …

Example:

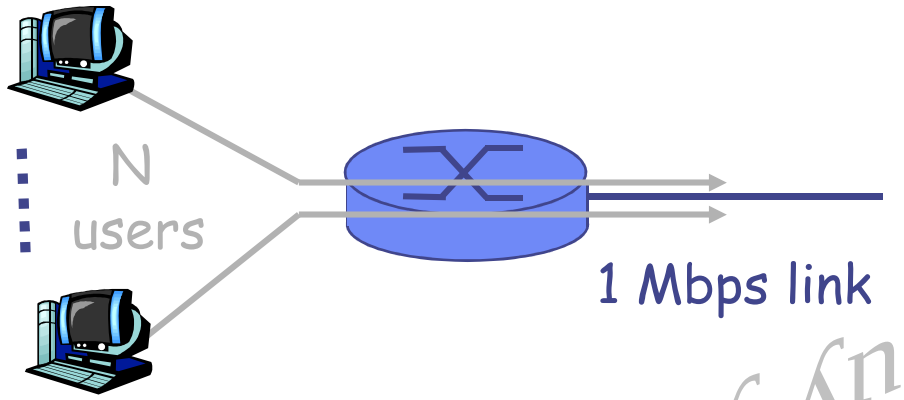
- L = 7.5 Mbits
- R = 1.5 Mbps
- transmission delay = 15 sec

Packet switching versus circuit switching

Packet switching allows more users to use network!

Example:

- 1 Mb/s link
- each user:
  - 100 kb/s when "active"
  - active 10% of time



- *circuit-switching:*
  - 10 users
- *packet switching:*
  - with 35 users, probability > 10 active at same time is less than .0004

Packet switching versus circuit switching

Is packet switching a "slam dunk winner?"

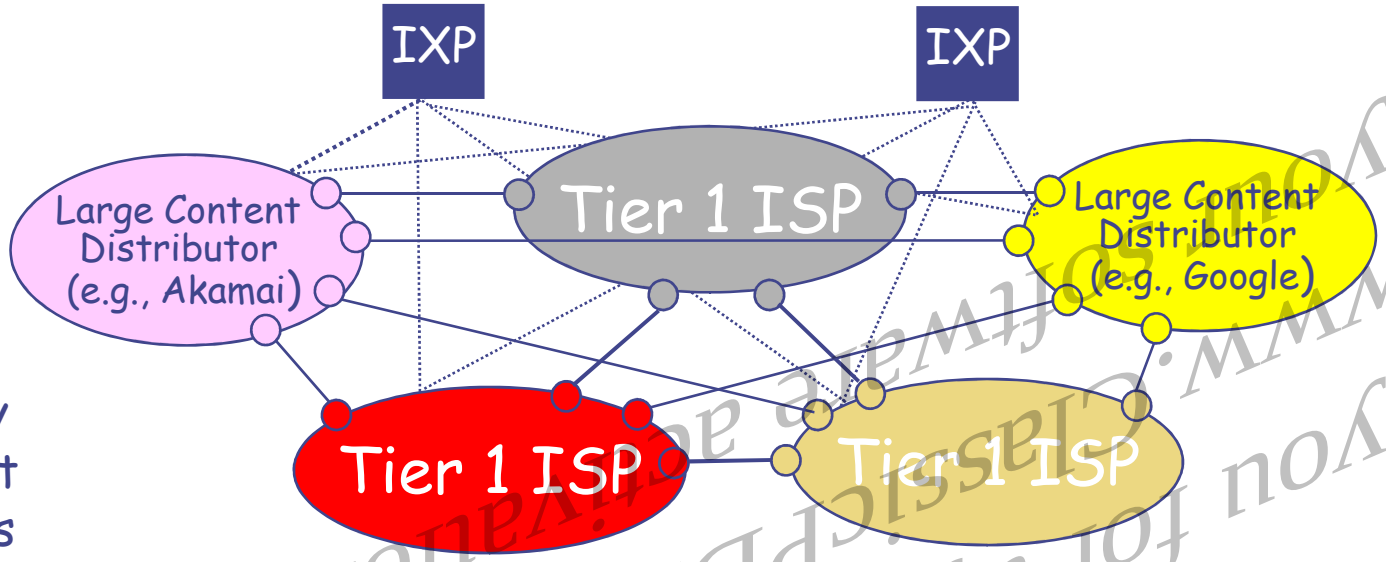
- great for bursty data
  - resource sharing
  - simpler, no call setup
- excessive congestion: packet delay and loss
  - protocols needed for reliable data transfer, congestion control
- Q: How to provide circuit-like behavior?
  - bandwidth guarantees needed for audio/video apps
  - still an unsolved problem (chapter 7)

Q: human analogies of reserved resources (circuit switching) versus on-demand allocation (packet-switching)?

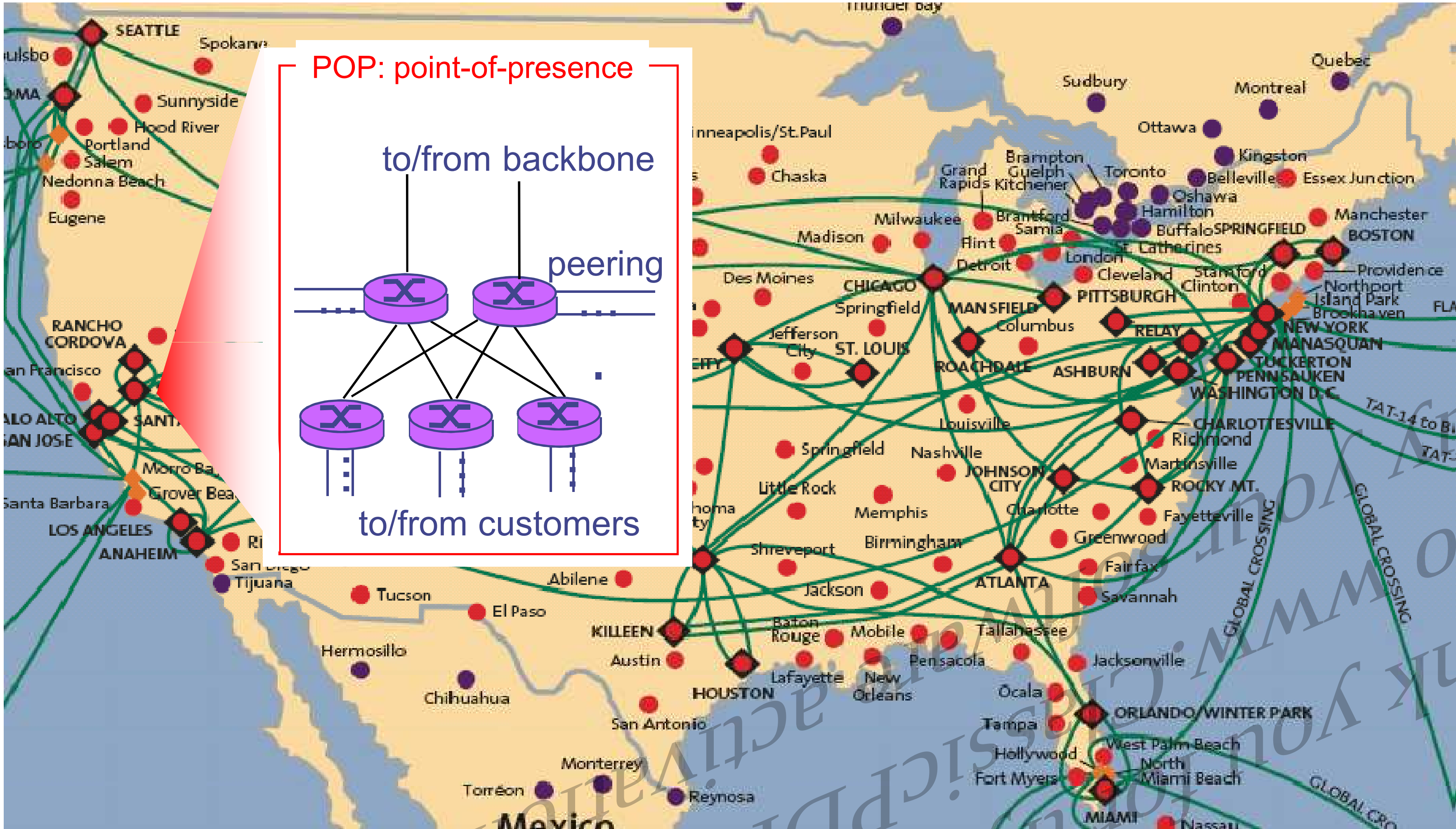
Internet structure: network of networks

- roughly hierarchical
- at center: small # of well-connected large networks
  - "tier-1" commercial ISPs (e.g., Verizon, Sprint, AT&T, Qwest, Level3), national & international coverage
  - large content distributors (Google, Akamai, Microsoft)
  - treat each other as equals (no charges)

Tier-1 ISPs & Content Distributors, interconnect (peer) privately … or at Internet Exchange Points IXPs



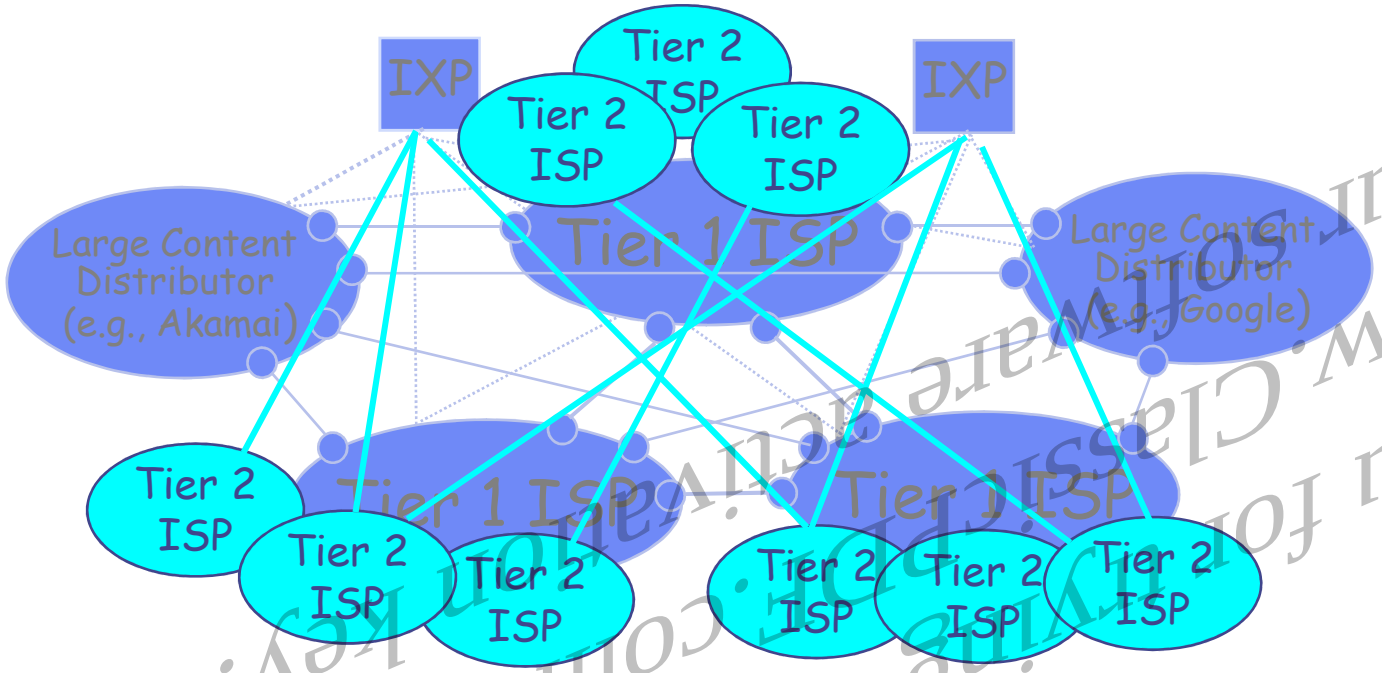
Tier-1 ISP: e.g., Sprint



Internet structure: network of networks

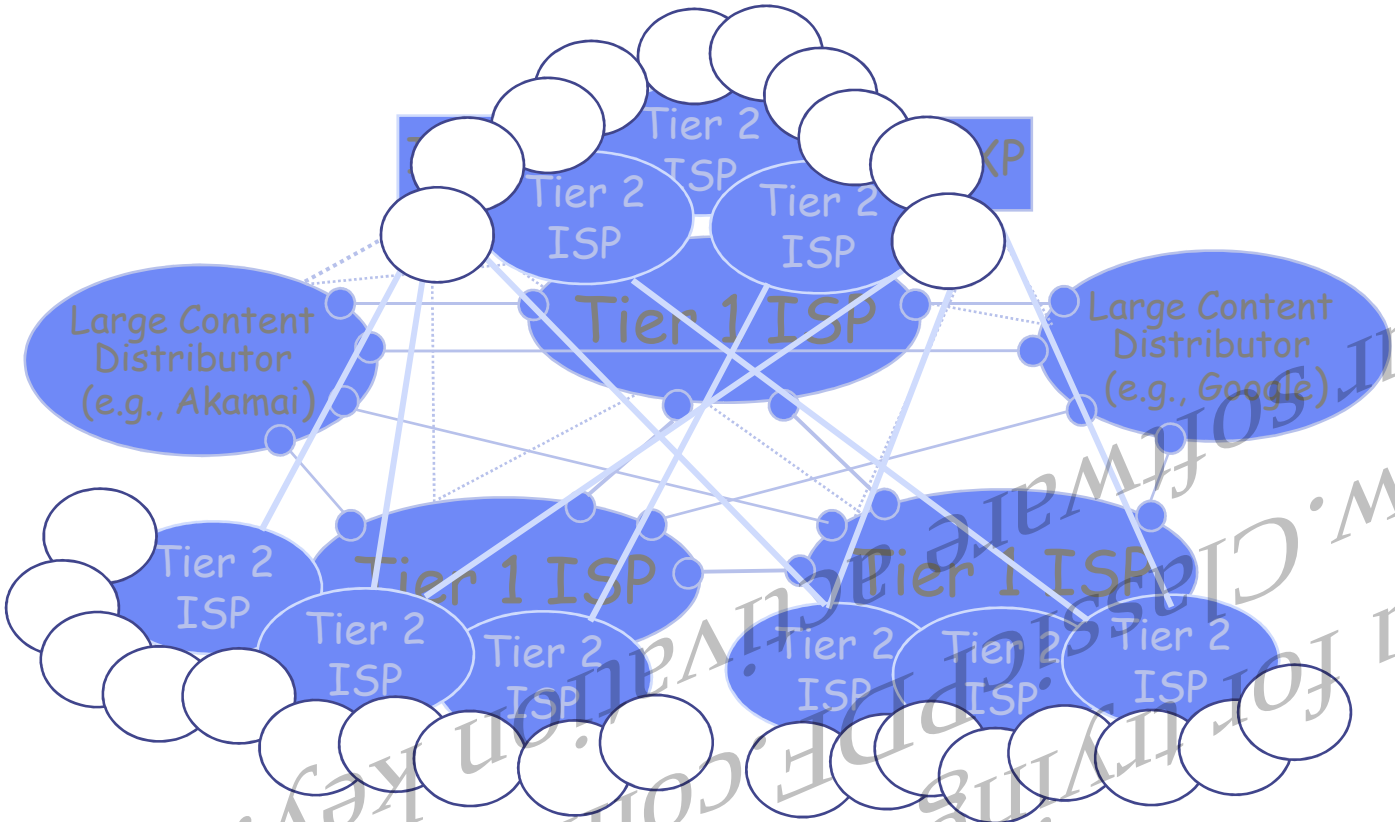
"tier-2" ISPs: smaller (often regional) ISPs

- connect to one or more tier-1 *(provider)* ISPs
  - each tier-1 has many tier-2 *customer nets*
    - tier 2 pays tier 1 provider
- tier-2 nets sometimes peer directly with each other (bypassing tier 1) , or at IXP



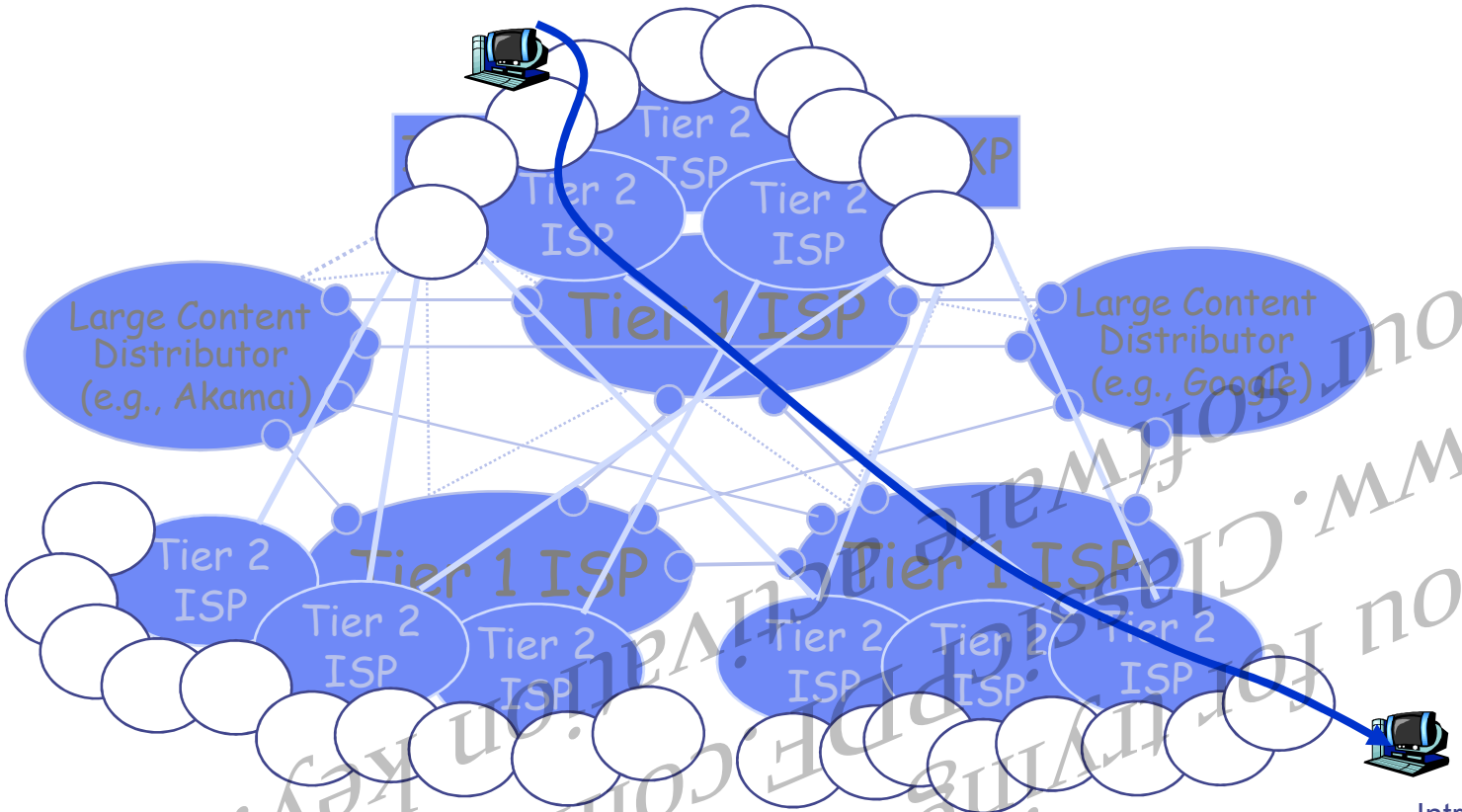
Internet structure: network of networks

- "Tier-3" ISPs, local ISPs
- customer of tier 1 or tier 2 network
  - last hop ("access") network (closest to end systems)



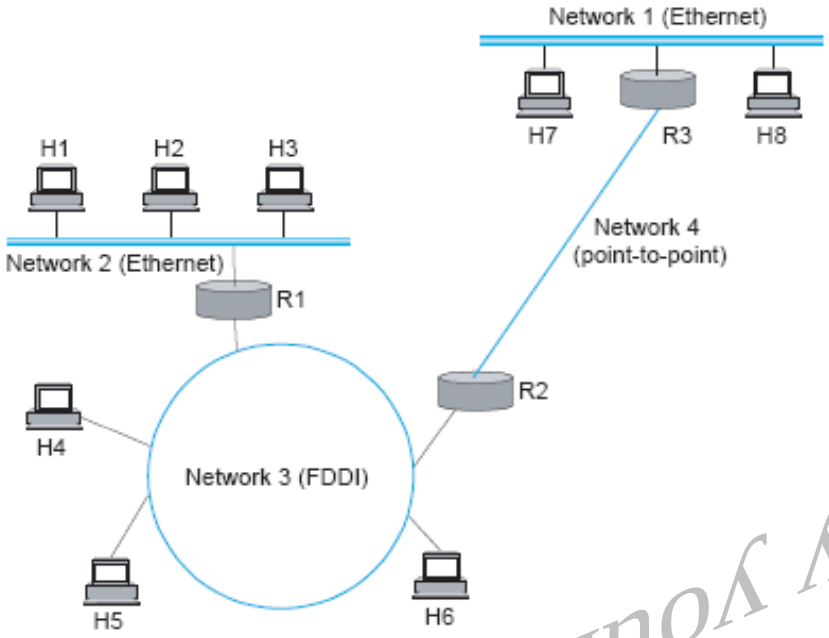
Internet structure: network of networks

- a packet passes through *many* networks from source host to destination host



Routing and Forwarding

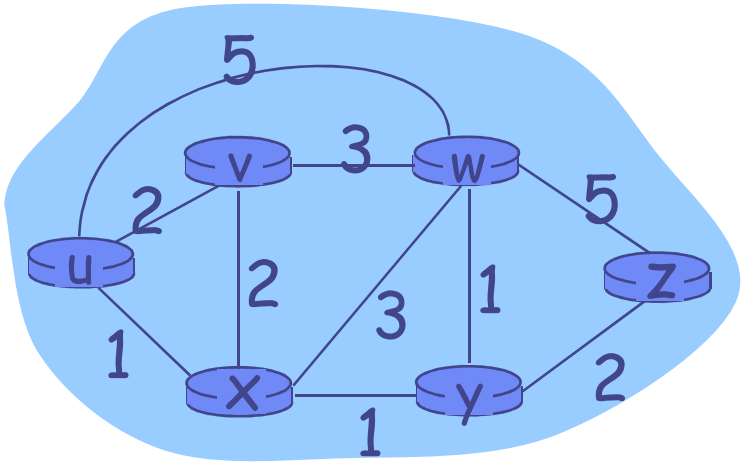
- Forwarding
  - Dispatch the packets based on forwarding table
- Routing
  - Disseminate network topology information
  - Find a path to the destination based on certain criteria (routing algorithm)
  - Identify the immediate neighbor to forward the packet → construct forwarding table



R2

Network#	nexthop
1	R3
2	R1

Routing: A graph problem



- c(x,x') = cost of link (x,x')

  - e.g., c(w,z) = 5

- cost could always be 1, or inversely related to bandwidth, or inversely related to congestion

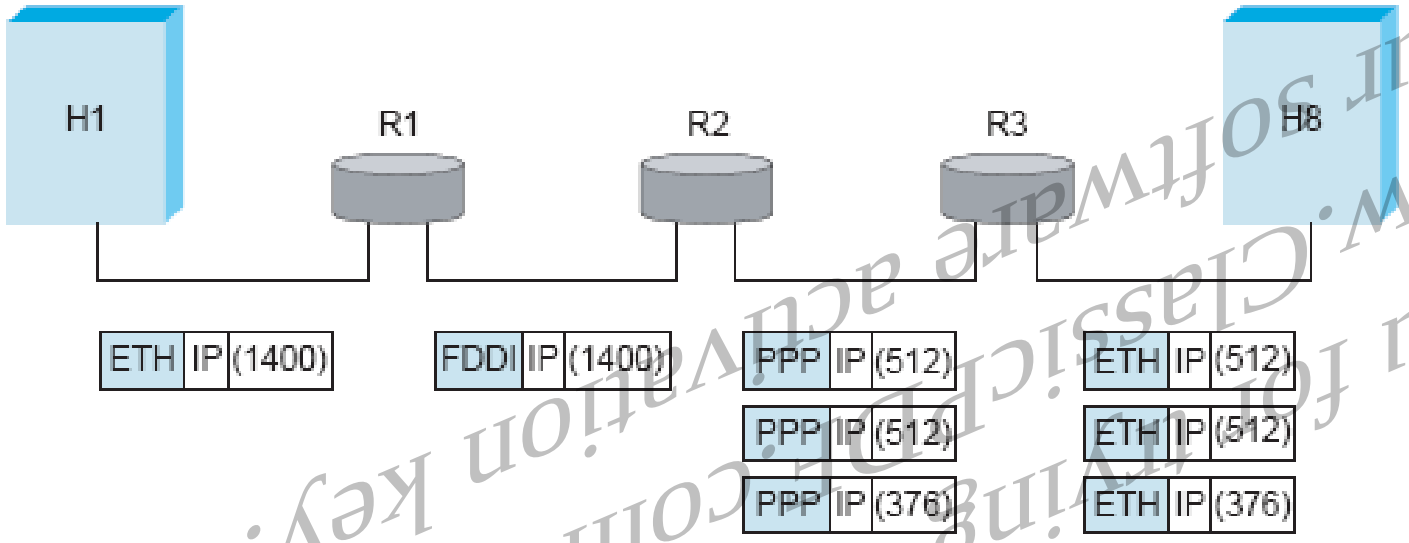
Cost of path $(x_1, x_2, x_3, \ldots, x_p) = c(x_1,x_2) + c(x_2,x_3) + \ldots + c(x_{p-1},x_p)$

Question: What's the least-cost path between u and z ?

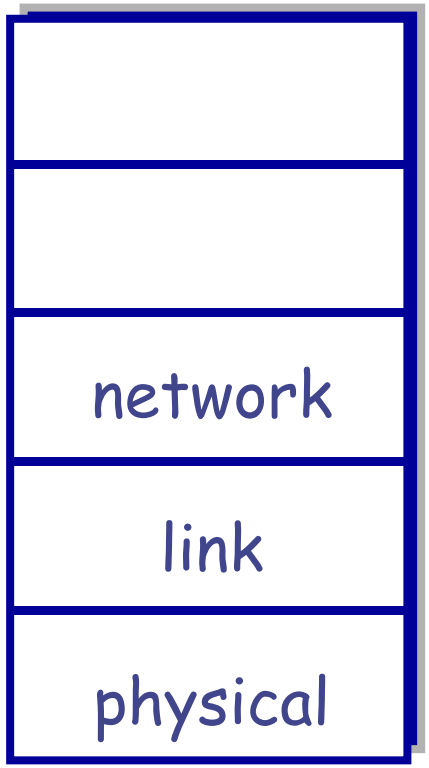
Routing algorithm: algorithm that finds least-cost path

Fragmentation and Reassembly

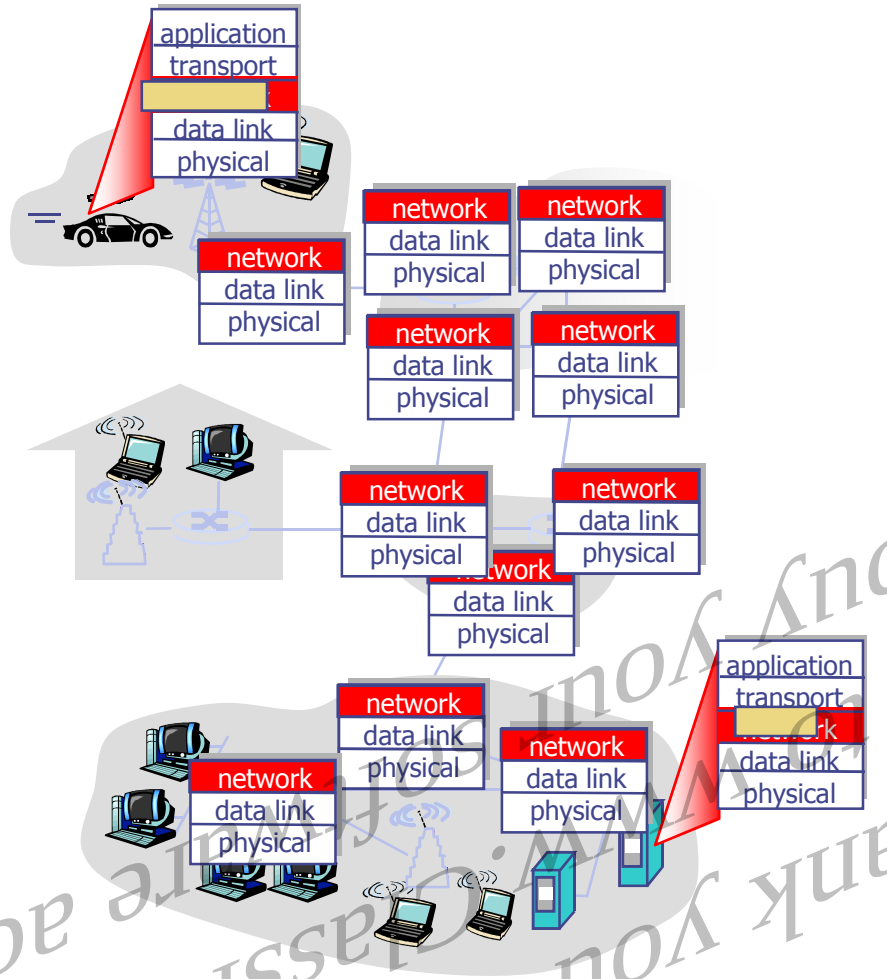
- Problem
  - Each network technology has its own definition of packet size.
- Solution
  - packets can be fragmented and reassembled when they are too big to cover a given network.
  - Each IP datagram is re-encapsulated for each physical network over which it travels.



Internet protocol stack: Network layer



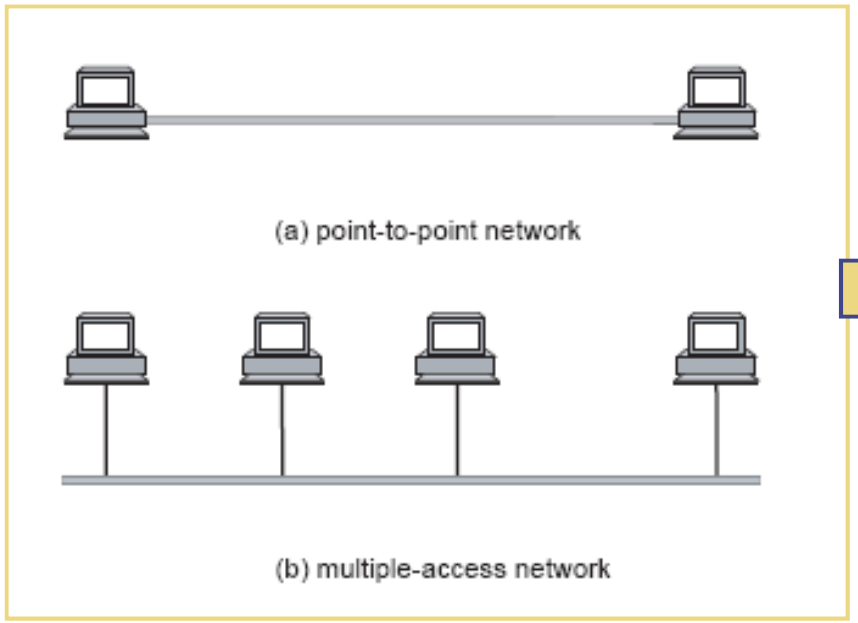
- network: routing of datagrams from source to destination
- link: data transfer between neighboring network elements
- physical: bits "on the wire"

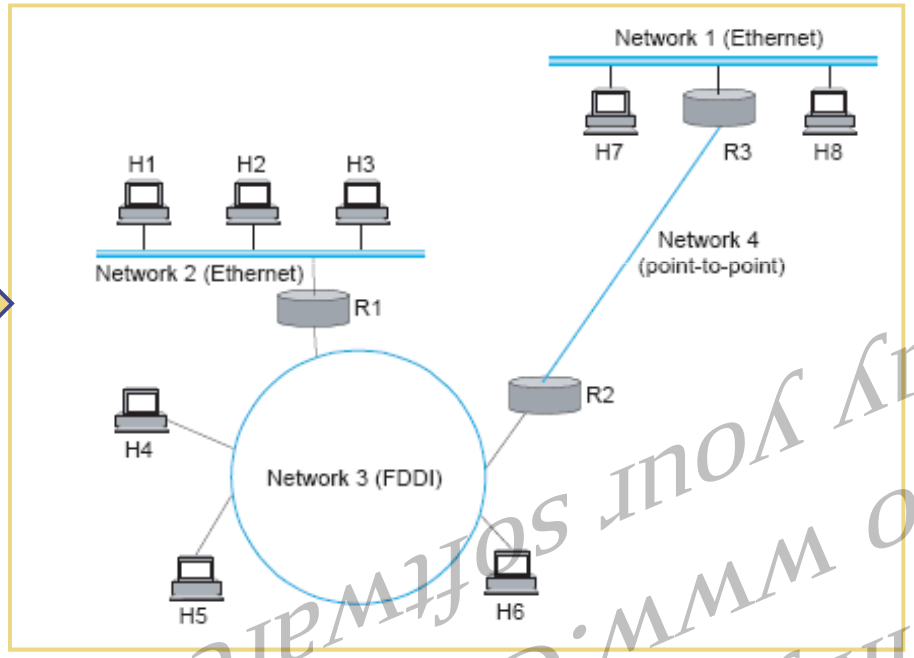


Overview

- Review: networking Layer
- End-to-end communication
- Protocol stack

From Direct Link Network To InterNetworking



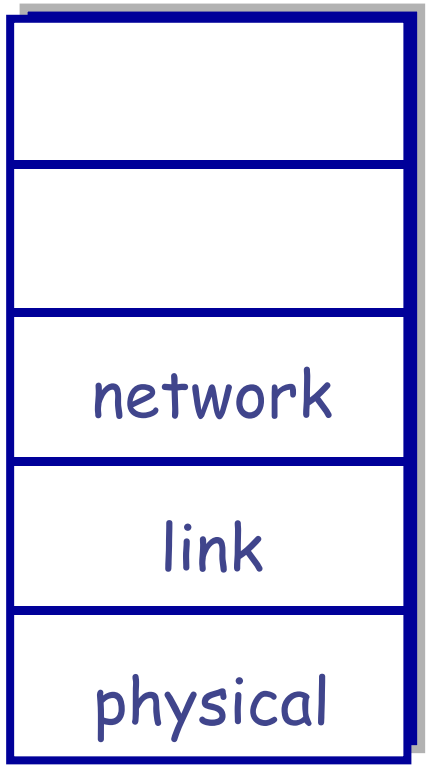


InterNetworking Overview

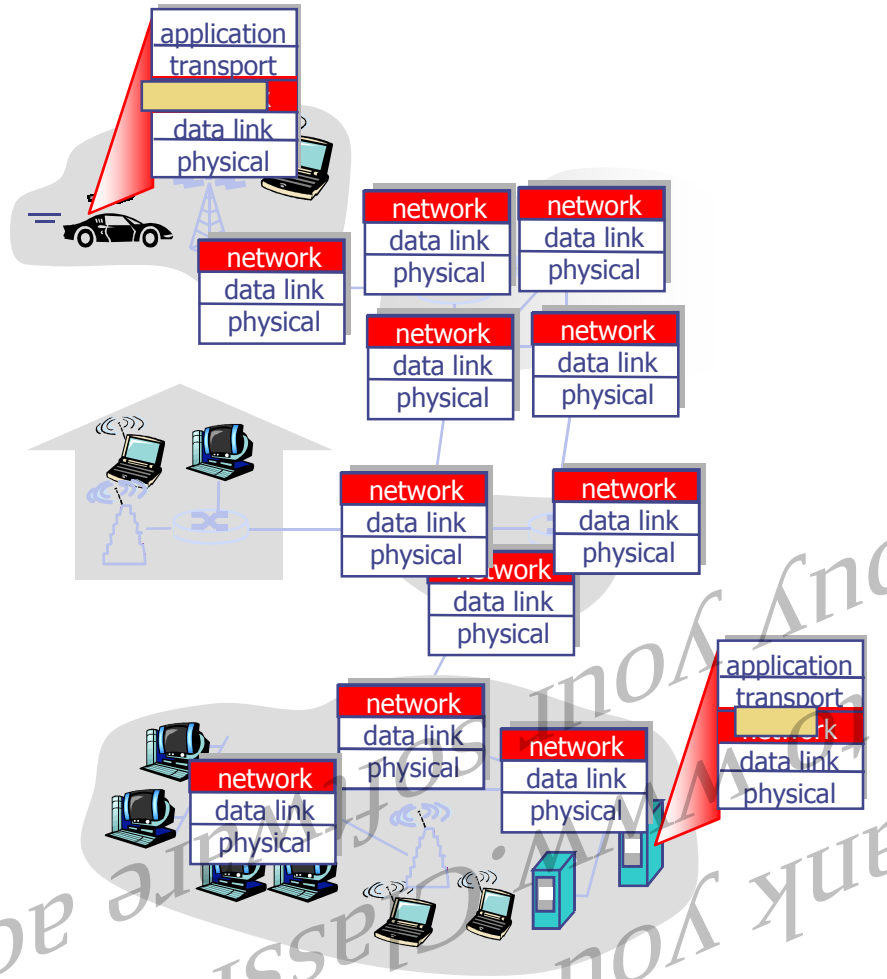
- Issues
  - Scale
    - the Internet doubled in size each year for 20 years.
  - Heterogeneity
    - internetwork connects networks with different technologies
- Functions
  - Providing suitable names for all hosts → Addressing
    - IP addressing → Hierarchical addressing facilitates scalable networking and routing
  - Building the internetworking infrastructure
    - Nuts and bolts → routers, links, etc.
  - Delivering data → forwarding
    - packet switching → Store and forward → statistical multiplexing enable efficient resource utilization

    [how Internet structure looks like – A reality check]
  - Finding a path → routing
  - Dealing Heterogeneity → Fragmentation and Reassembly

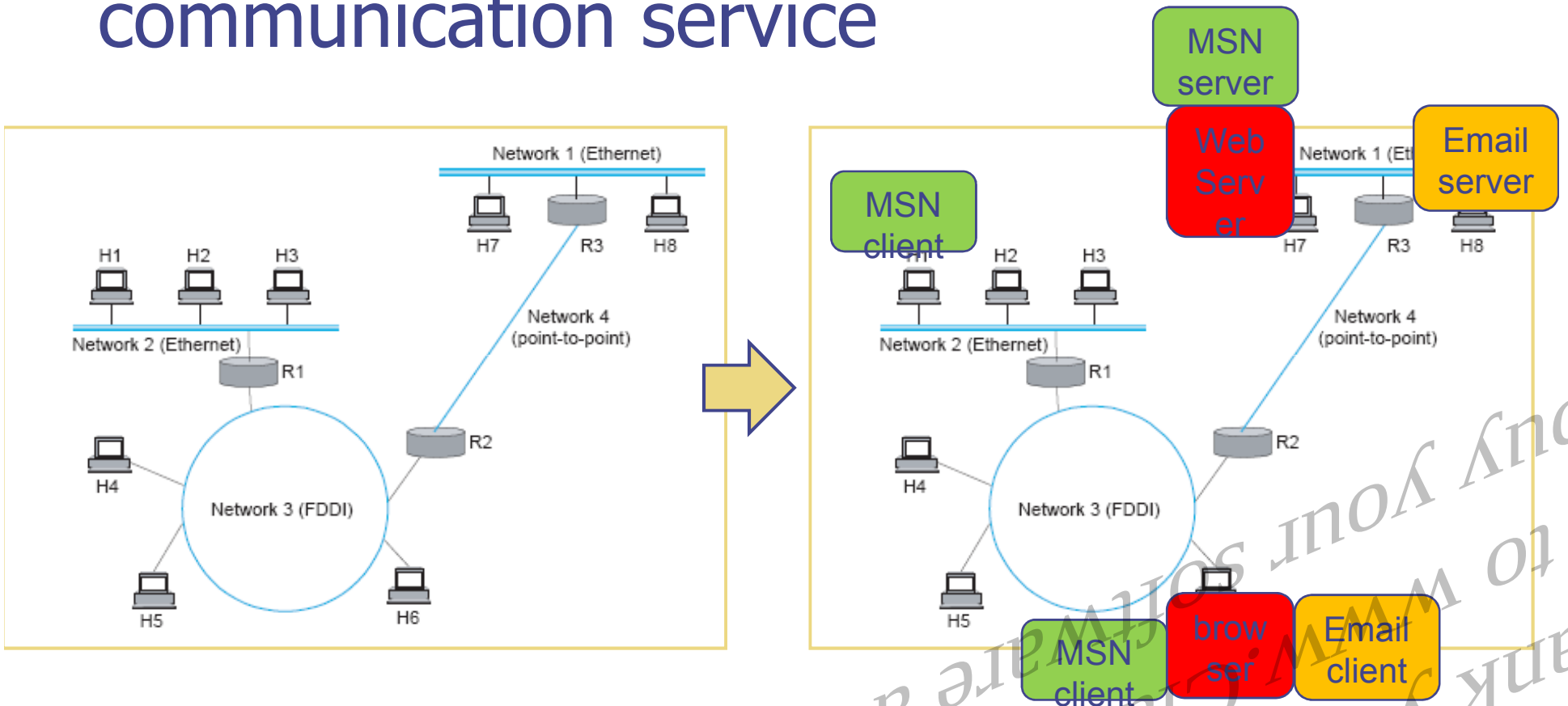
Internet protocol stack: Network layer



- network: routing of datagrams from source to destination
- link: data transfer between neighboring network elements
- physical: bits "on the wire"

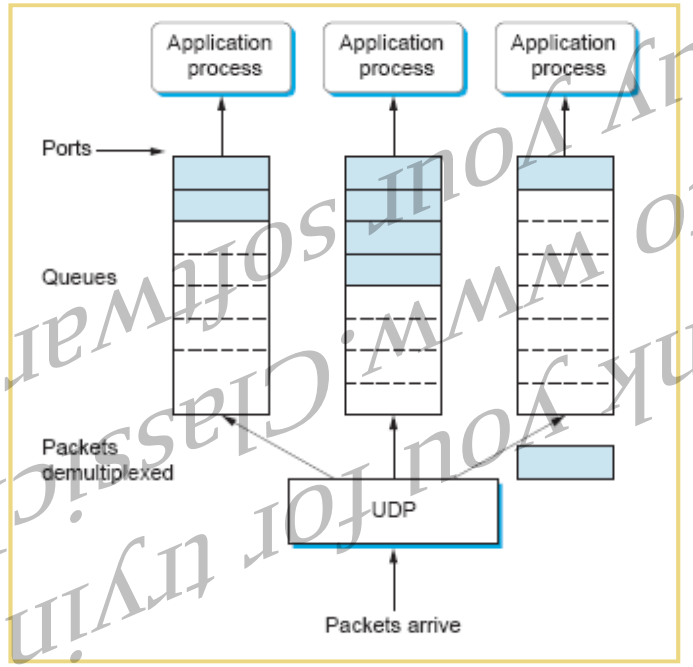


From host-to-host data delivery to application-to-application communication service

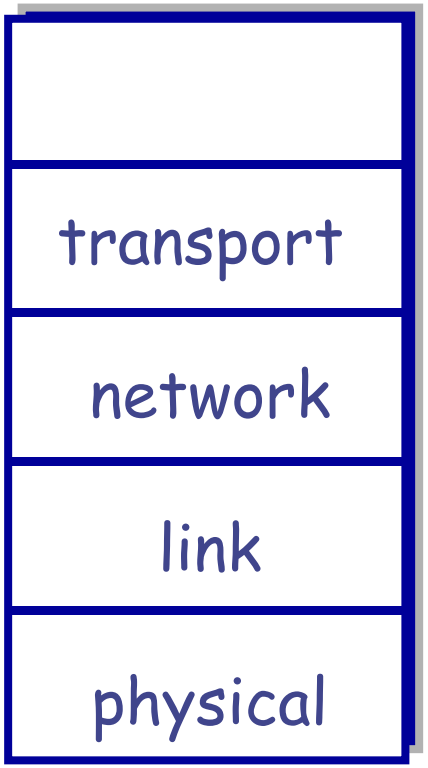


End-to-End Protocols

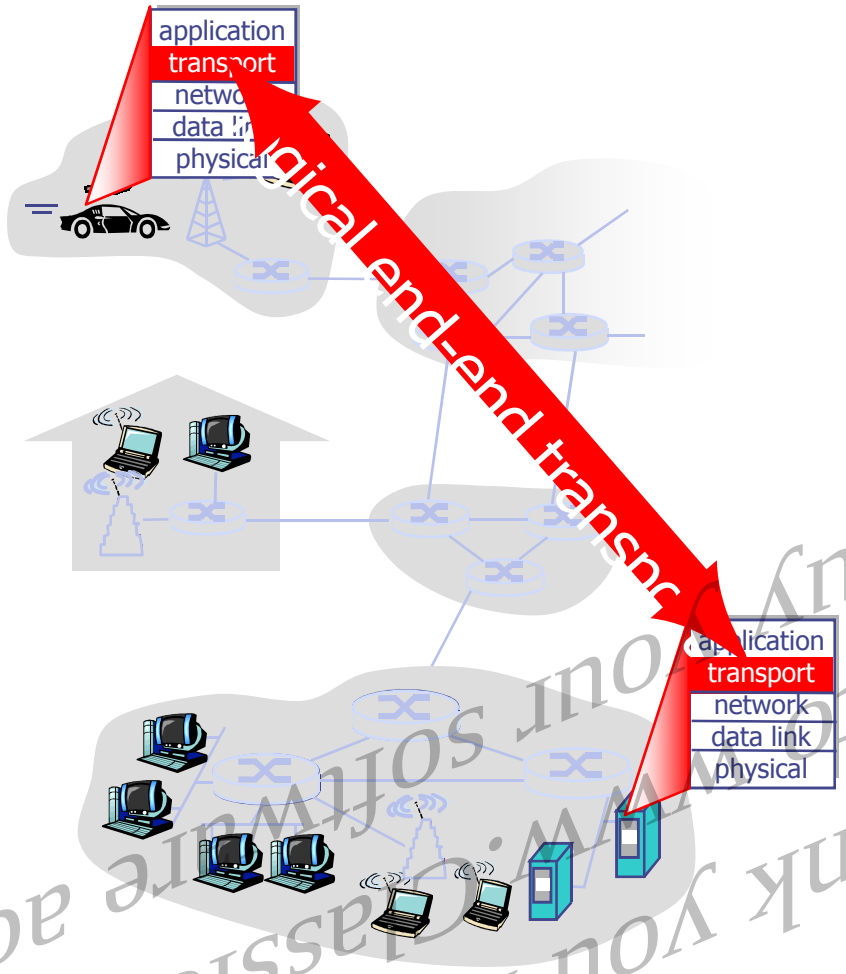
- Problem
  - Turn host-to-host packet delivery service into a logical communication channel between application processes.
- End-to-end protocols of Internet
  - Different services:
    - UDP: Best effort connectionless
    - TCP: Reliable Connection-oriented protocol
      - Connection establishment
      - Reliable transmission
      - Congestion control



Internet protocol stack: Transport layer

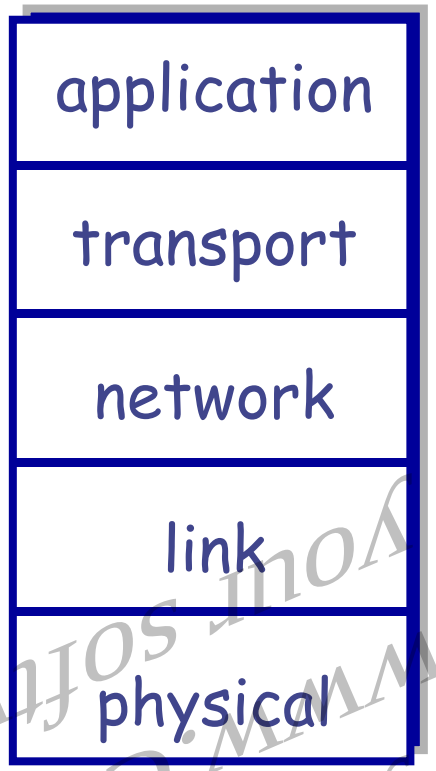


- transport: process-process data transfer
- network: routing of datagrams from source to destination
- link: data transfer between neighboring network elements
- physical: bits "on the wire"



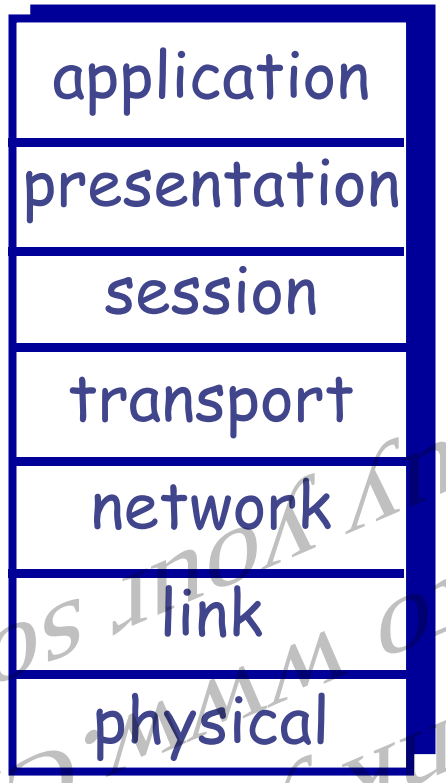
Internet protocol stack

- 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
  - Ethernet, 802.111 (WiFi), PPP
- 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,* may be implemented in application/transport layer



Why layering?

Dealing with complex systems:

- explicit structure allows identification, relationship of complex system's pieces
  - layered reference model for discussion
- modularization eases maintenance, updating of system
  - change of implementation of layer's service transparent to rest of system
  - e.g., change in gate procedure doesn't affect rest of system
- layering considered harmful?

Message Encapsulation