Infrastructure Layers

Infrastructure
Internet perspective

Change in point of view

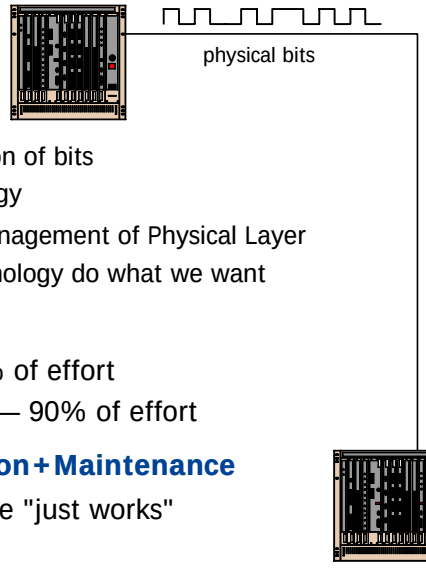
Internet standards do not discuss Data Link + Physical Layers

Hardware developers define standards

Internet Layer Model		
Application	Internet application Expects Internet services from OS	Internet Aware
Transport	Local + remote ports Service requirements	
Network	End-to-end IP routing + forwarding	
Infrastructure	Data Link Layer — hardware management Physical Layer — hardware	Not Internet Aware

Infrastructure
Engineering perspective

Infrastructure layers



Bottom-up design

Physical layer (**PHY**)

Defines physical transmission of bits

Exploits a physical technology

Data Link layer (**DL**) defines management of Physical Layer

How to make physical technology do what we want

Infrastructure management

Delivering data messages — 10% of effort

Making hardware work correctly — 90% of effort

OAM = Operations+Administration+Maintenance

Application assumes infrastructure "just works"

"Just works" ⇒

Reliability, availability, stability, serviceability, growth

Infrastructure
Economic perspective

Enormous investment in existing equipment

Global network of hardware nodes + transmission lines

Developed to provide many services

Internet (IP-based unreliable connectionless) just one service

Most developed before Internet

Telegraph — 1794

Telephone — 1876

Teletype modem — 1943

Digital telephone — 1962

Internet opened to public — 1992

Hardware updates

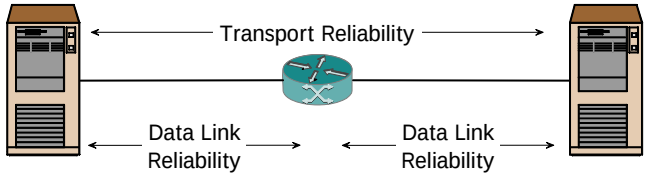
Replacement of manufactured hardware

Slower than software updates

More expensive than software updates

Data Link Functions

Similar to transport layer functions



Framing

- Assemble network PDUs into hardware packets
- Attach header + trailer for Data Link and Physical layers

Medium access + flow control + congestion control

- When / how transmitter sends data onto link
- Transmitter avoids overflow of receiver buffer
- Transmitters avoid interfering with other transmitters

Error control

- Detect / correct transmission bit errors

Local addressing

- Convert network addresses to hardware local addresses

Data Link Sublayers

Logical Link Control (LLC) sublayer

- Multiplexing of data sources / destinations
- Packet type identification
- Error correction
- Flow control

Medium Access (MAC) sublayer

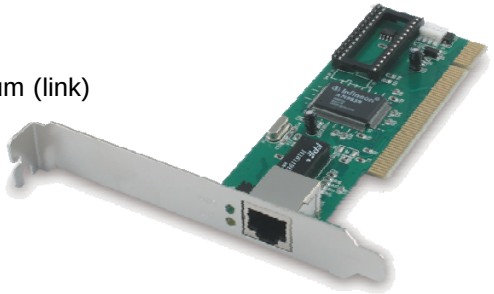
- Network topology
- Medium access management
  - Sharing medium among nodes
  - Permission to transmit
- Data frame structure
- Hardware (MAC) addressing
- Error detection

LLC Sublayer	Data Link Layer	2
MAC Sublayer		
Physical Layer		1

Interface to Data Link Layer

Infrastructure layers

- Typically implemented in hardware
  - PHY — physical circuits for transmitter / link / receiver
  - DL — embedded program in firmware (ROM) + controller
- Host node
  - Network interface Card (NIC)
    - Connection port to medium (link)
    - Controller
    - Transceiver



- Switching node
  - Switching fabric
  - Controller + multiple transceivers + connection ports to medium (link)

DL layer interface

- Interface to OS level hardware driver
- Network PDU ↔ OS driver ↔ NIC / switch controller ↔ transceiver

Framing

Data frame format

Header	Data	Trailer

Data link protocol management

- Header / Trailer format
  - Similar to headers at network and transport layers
- Addressing, error control, flow control, ...

Physical layer hardware management

- Transmission parameters
  - Bit rate, Baud rate, modulation method, ...
- Transmitter / receiver synchronization
  - Clock training bits
  - 1010101010 ... allows receiver clock to sync
- Frame marking
  - Start / Stop Fields
  - Start field / byte count
  - Lose sync ⇒ drop frame + wait for new Start Field

Frame Marking Methods

STX / ETX

ASCII control codes

STX	Start of Text	0x02
ETX	End of Text	0x03
DLE	Data Link Escape	0x10

DLE STX	Rest of Header	Data	Rest of Trailer	DLE ETX

DLE in data stream

Byte stuffing

Transmitter sends **DLE** as **DLE DLE**

Receiver removes extra **DLE**

Frame Marking Methods

Flags

Start = Stop = 01111110 = 0x7E

01111110	Rest of Header	Data	Rest of Trailer	01111110

7E in data stream

Byte / bit stuffing

Byte stuffing

Send **7E** as **7E 7E** — receiver removes extra **7E**

Bit stuffing

Send **11111** as **111110** — receiver removes extra **0**

Error Control

Check sequence

Transmitter

Calculates hash of data

Includes sequence in transmitted header / trailer

Receiver

Calculates hash of data

Compares received sequence with calculated sequence

Error Control

Parity (even parity)

Parity bit = XOR of data bits

Data + parity = even number of 1 bits

Cyclic Redundancy Code (CRC)

D = Data field

G = Generator

Predetermined pattern of r+1 bits

R = Remainder of $(D \times 2^r) / G$ (modulo 2 division) = CRC field

Generally $(D \times 2^r + R) / G = 0$

D = data	$R = (D \times 2^r) / G$

Modulo 2 Polynomial Arithmetic

Represent data bits as coefficients of polynomial

$$D = a_{n-1}a_{n-2}\ldots a_0$$

$$D(x) = a_{n-1}\times x^{n-1} + a_{n-2}\times x^{n-2} + \ldots a_0 \times x^0$$

Arithmetic modulo 2 in each order (XOR)

$$0+0=1+1=0 \qquad 0+1=1+0=1$$

Polynomial addition = subtraction

$$A(x)\pm B(x) = a_{n-1}\times x^{n-1} + \ldots + a_0\times x^0 \pm \left(b_{n-1}\times x^{n-1} + \ldots + b_0 \times x^0\right)$$
$$= \left(a_{n-1}\oplus b_{n-1}\right)\times x^{n-1} + \ldots + \left(a_0 \oplus b_0\right)\times x^0$$

Polynomial multiplication / division

Multiply / divide as usual

Modulo 2 arithmetic in each order

Examples $\left(x^3+x\right)\left(x^2+1\right) = x^5 + x^3 + x^3 + x = x^5 + (1+1)x^3 + x = x^5 + x$

$\left(x^5+x\right)\div\left(x^2+1\right) = x^3 + x$

Modulo 2 Long Division

$$\frac{x^5+x}{x^2+1}$$

$$x^2+1 \;\overline{)\; x^5 + x \;}\; x^3 + x$$

$$x^5 + x^3$$

$$x^3 + x$$

$$x^3 + x$$

$$0$$

Cyclic Redundancy Code (CRC)

Why it works

Data

D = data

Shift left D r bits ⇒ D → D × 2^r

D = data	0 … 0

Divide by G ⇒ D × 2^r → D × 2^r / G = Q + **R** / G

Quotient Q

Remainder R

Transmit T = D × 2^r + **R**

D = data	R

Receiver calculates T / G

T / G = D × 2^r / G + **R** / G = (Q + **R** / G) + **R** / G = Q + (**R** + **R**) / G

= Q + **0** / G = Q

CRC Example

Data stream

D = 1101011011

Generator

G = 10011

```
                1 1 0 0 0 0 1 0 1 0
          ------------------------
1 0 0 1 1 | 1 1 0 1 0 1 1 0 1 1 0 0 0 0
            1 0 0 1 1
          ---------
              1 0 0 1 1
              1 0 0 1 1
          ---------
                  0 1 0 1 1 0
                  1 0 0 1 1
                ---------
                  0 0 1 0 1 0 0
                    1 0 0 1 1
                  -----------
                      1 1 1 0
```

Remainder

R = 1110

Transmit

T = 1101011011 1110

CRC check at receiver

```
                1 1 0 0 0 0 1 0 1 0
          ------------------------
1 0 0 1 1 | 1 1 0 1 0 1 1 0 1 1 1 1 1 0
            1 0 0 1 1
          ---------
              1 0 0 1 1
              1 0 0 1 1
          ---------
                  0 1 0 1 1 1
                  1 0 0 1 1
                ---------
                  0 0 1 0 0 1 1
                    1 0 0 1 1
                  -----------
                      0 0
```

Zero remainder ⇒ no error

CRC Standards

CRC-12

$$G(x) = x^{12} + x^{11} + x^3 + x^2 + x^1 + 1$$

CRC-16

$$G(x) = x^{16} + x^{15} + x^2 + 1$$

CRC-CCITT

$$G(x) = x^{16} + x^{15} + x^5 + 1$$

CRC-32

$$G(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$$

Flow Control

Go Back N (GBN)

- Transmit N frames
- Wait for ACK
- Piggybacking — transmit ACK signals in data frame

Sliding Window

- N-bit SEQ number
- Window size — number of unACKed frames before stopping
- Source window
  - SEQ numbers of unACKed frames
  - Frames buffered at transmitter until ACKed
- Destination window
  - SEQ numbers of frames to be accepted
  - Frames passed to network layer in SEQ order
    - Out-of-order frames buffered
    - Frames too far ahead of window rejected
- Timeout — retransmit if no ACK after fixed time

High-Level Data Link Control (HDLC)

Family of data link protocols

Based on IBM SDLC

- Layer 2 protocol in mainframe SNA
- Originally for communication between CPUs and peripherals

Link Access Protocol (LAP)

Versions of HDLC used in public network architectures

X.25	**LAPB**
Frame Relay	**LAPF**
ISDN	**LAPD**
Ethernet Logical Link Control (LLC)	**IEEE 802.2**
Internet point-to-point	**SLIP, PPP**

High-Level Data Link Control (HDLC)

Data link attributes in HDLC

Framing	0x7E flag with byte / bit stuffing
Addressing	hardware level addressing possible
Error control	16-bit CRC-CCITT or 32-bit CRC-32
Flow control	3-bit SEQ number
Frame types	Information, Supervisory, Unnumbered

HDLC Frame Structure

General HDLC frame

01111110	Address	Control	data	CRC	01111110
8	8	8	≥ 0	16 / 32	8

Address

8 bit address ⇒ 256 hardware addresses

Control field

Specifies frame type / control

Information (data)

7	6 5 4	3	2 1 0
0	SEQ N(S)	p/f	NEXT N(R)

Supervisory (flow control)

7	6	5 4	3	2 1 0
1	0	type	p/f	NEXT N(R)

Unnumbered (management / connectionless)

7	6	5 4	3	2 1 0
1	1	type	p/f	subtype

HDLC Control Fields

Flow control

SEQ — sequence number of data frame

NEXT — next expected SEQ (ACK all previous frames)

Type

00 — ACK + Receiver Ready (RR)

01 — Reject (REJ): retransmit all frames from N(R)

10 — Receiver Not Ready (RNR): ACK N(R) but stop sending

11 — Selective Reject (SREJ): retransmit N(R)

Information (data)

7	6 5 4	3	2 1 0
0	SEQ N(S)	p/f	NEXT N(R)

Supervisory (flow control)

7	6	5 4	3	2 1 0
1	0	type	p/f	NEXT N(R)

Unnumbered (management / connectionless)

7	6	5 4	3	2 1 0
1	1	type	p/f	subtype

HDLC Polling

Polling

Primary host initiates communication

Secondary host responds

Poll / Final (p/f) bit

Invitation — primary to secondary with p = 1

Response

Secondary sends I-frames to primary with f = 0

Secondary sets f = 1 on last response frame

Information (data)

7	6 5 4	3	2 1 0
0	SEQ N(S)	p/f	NEXT N(R)

Supervisory (flow control)

7	6	5 4	3	2 1 0
1	0	type	p/f	NEXT N(R)

Unnumbered (management / connectionless)

7	6	5 4	3	2 1 0
1	1	type	p/f	subtype

HDLC Internet Dial-Up Protocols

Serial Line Internet Protocol (SLIP)

RFC 1055

0xC0	IP datagram with byte stuffing (C0 → DB DC, DB→ DB DB)	0xC0

Point-to-Point Protocol (PPP)

Layer 2 protocol used between

Internet routers

Host and Internet service provider (ISP)

01111110	11111111	11000000	Protocol	Data	CRC	01111110
8	8	8	8 or 16	≥ 0	16 / 32	8

Address = 11111111 = broadcast

HDLC control = 11000000 = Unnumbered (connectionless data)

Protocol

Protocol in data field

Network protocol or link negotiation protocol (upper layer 2 sublayer)

PPP Protocol Options

01111110	11111111	11000000	Protocol	Data	CRC	01111110
8	8	8	8 or 16	≥ 0	16 / 32	8

Standard network protocols

- IP, IPX, AppleTalk, ...
- Datagram in data field

Control Protocols

- Link Control Protocol (LCP)
  - PPP options
    - Header compression (remove control / address fields)
    - Size of protocol / CRC fields and data
  - Test
  - Terminate
- Network Control Protocol (NCP)
  - Network layer options
    - Protocol
    - Address
    - Header compression (encode header fields)
  - Authentication (ISP user / password exchange)

Shared Medium Networks

Physical layer

- Multiple nodes transmit on single medium
  - Time division
  - Frequency division
  - Code division
- Shared physical medium ⇒ local area network (LAN)

Data link layer

- Medium access (MAC) sublayer
  - Allocates medium capacity among nodes
  - Error detection
  - Network topology
- Logical link control (LLC) sublayer
  - Frame types
  - Flow control
  - Error correction
  - Protocol negotiation

LLC Sublayer	Data Link Layer	2
MAC Sublayer		
Physical Layer		1

Medium Access Sharing

Time division

- Each host granted full bandwidth in allocated time slot
  - Time slot allocated statistically or deterministically
- Examples
  - Deterministic — telephone switching
  - Statistical — Ethernet, WiFi, ...

Frequency division

- Each host granted partial bandwidth in all time slots
- Examples
  - Commercial radio / TV
  - Bluetooth

Code division

- Each host granted full bandwidth in all time slots
- Each host transmits using different coding scheme
- Example
  - Cellular CDMA

Common Shared Medium Networks

Ethernet	IEEE 802.3 local area network Wired time-division access
WiFi	IEEE 802.11 local area network Wireless time-division access
Bluetooth	IEEE 802.15 personal area network Wireless frequency-division access
WiMAX	IEEE 802.16 metropolitan area network Wireless time/frequency-division access
GSM / UTMS	ITU 2G / 3G cellular network Wireless time/frequency-division access
CDMA / CDMA2000	ITU 2G / 3G cellular network Wireless code-division access

What are IEEE 802 and ITU?

Institute of Electrical and Electronics Engineers (IEEE)

- Professional organization
- Coordinates technical standards for electronic equipment

IEEE 802 Committee

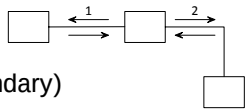
- Standards committee for Data Link and Physical Layer
- OEMs (original equipment manufacturers)
  - Develop hardware / software systems at infrastructure layers
  - Request standardization (recognition) from 802 committee
- Other 802 standards
  - 802.1 — bridging (interconnecting different 802 LANs)
  - 802.2 — LLC sublayer for 802 LANs
  - 802.4 — Token Bus (LAN for manufacturing environments)
  - 802.5 — Token Ring (ring topology LAN)

International Telecommunication Union (ITU)

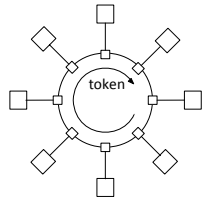
- UN standards committee
- Sets telephone and (non-Internet) WAN standards

Time Division Medium Access

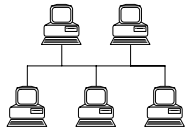
Deterministic



- Polling
  - Primary node initiates session (sends data to secondary)
  - Secondary node responds (sends data to primary)
  - Optional mode in WiFi
- Token passing
  - Token message passes from host to host
  - Host with token may transmit
  - Token Ring, Token Bus, FDDI



Statistical



- Aloha
  - Hosts transmit at random / hope to avoid collisions
- Carrier sense
  - Hosts listen for other transmissions / try to avoid collisions
- Arbitration
  - Deterministic procedure chooses among random group of hosts

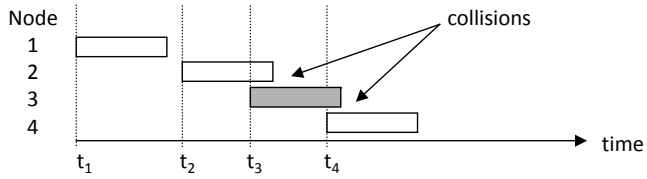
ALOHAnet

Background

- First wireless packet data network (1971)
- Low data volume
- Connected University of Hawaii campuses (separate islands)

Protocol

- Host transmits when ready
- Two frames overlap in time
  - Collision ⇒ both frames corrupted
  - Retransmit after random wait



Utilization and Throughput

Capacity

τ = time (seconds) to transmit 1 packet

R = maximum packets/second on medium = $\frac{1}{\tau}$

Traffic

λ = actual packets/second transmitted by hosts

Utilization

$$G = \frac{\lambda}{R} = \lambda\tau$$

Collisions

P_0 = probability success (packet transmitted without collision)

At receiver

$\lambda' = \lambda P_0$ = uncorrupted packets/second

Throughput

$$S = \frac{\lambda'}{R} = \frac{\lambda P_0}{R} = GP_0$$

ALOHA Throughput

Success = no collisions

Previous packet starts at $t_2 < t_3 - \tau$

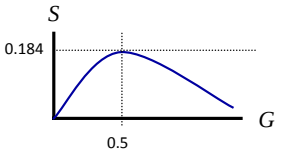
Next packet starts at $t_4 > t_3 + \tau$

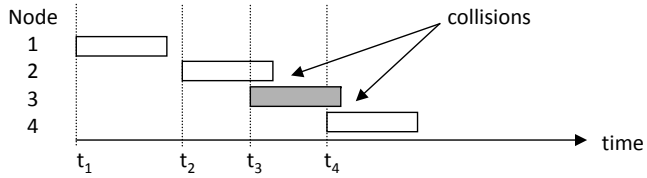
Interval of no transmissions = $t_4 - t_2 > 2\tau$

Packets obey Poisson statistics

$$P(k \text{ packets in } T \text{ seconds}) = \frac{(\lambda T)^k}{k!} e^{-\lambda T} \Rightarrow P_0 = \frac{(2\tau\lambda)^0}{0!} e^{-2\tau\lambda} = e^{-2\tau\lambda} = e^{-2G}$$

$$\boxed{S = Ge^{-2G}}$$





Slotted ALOHA

Central clock

Synchronize packet transmissions

Transmit new packet constructed between t_k and $t_k + \tau$

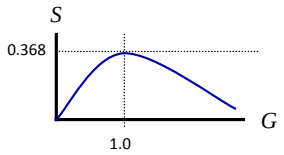
Collision

Two nodes construct packets in same interval

Probability of success (of my packet)

No other packets constructed during interval τ

$$P_0 = \frac{(\tau\lambda)^0}{0!} e^{-\tau\lambda} = e^{-\tau\lambda} = e^{-G} \Rightarrow \boxed{S = Ge^{-G}}$$



Carrier Sense

Carrier Sense Multiple Access (CSMA)

Nodes listen for transmissions before transmitting

No transmission — node can transmit

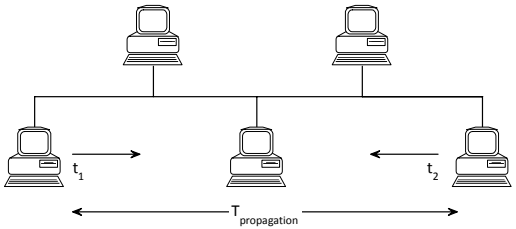
Transmission — node waits until end of transmission

Collision

Multiple nodes transmit "at same time"

"Same time"

$|t_1 - t_2| < T_{propagation}$



Collision Detection (CD)

Nodes listen for collision

Corrupted data

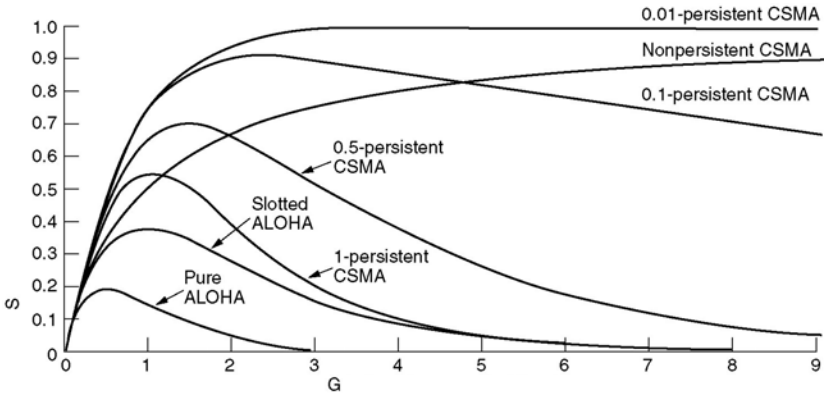
On collision

All nodes stop transmitting

Nodes jam transmission

Nodes waits random backoff before retransmitting

CSMA/CD Throughput



Tanenbaum, Fig. 4-4

Persistent CSMA

No carrier detected ⇒ node with data transmits

q-persistence

Node transmits with probability $0 < q < 1$

$q < 1$ ⇒ fewer collisions but longer latency

Arbitration

Deterministic medium access

- Random group of nodes request access
- One node chosen by deterministic algorithm
- No collisions
- Efficient throughput

Used within computer

- Peripheral Component Interconnect (PCI)
  - Multiple CPUs and peripherals compete for access to memory
  - PCI bridge allocates memory access efficiently
- Intel Multibus II
  - Multiple nodes request bus access using pseudo-randomized ID
  - Highest ID proceeds

Binary countdown switch

- Multiple hosts begin transmitting onto bus
  - Bus output = logical OR of all inputs
- Host sends 0 but sees 1 on bus ⇒ host stops

Ethernet

Family of wired LAN systems

- Defined at physical and data link layers
- Dominant / generic LAN technology

Background

- Developed 1974 at Xerox PARC
- Commercialized by Xerox / Intel / Digital in 1980
- Standardized as IEEE 802.3 in 1982
  - 10 Mb/s baseband transmission
  - Bus topology — single coaxial cable < 2.5 km
  - CSMA/CD
    - Shared bus topology → CSMA
    - Long propagation delay on coaxial cable → CD

Developments

- Bit rates: 10 Mbps → 100 Mbps → 1 Gbps → 10 Gbps → 100 Gbps
- Media: coaxial cable → hub (virtual bus on star) → switch
  - Ethernet switch — non-blocking N × N switch with no collisions

Ethernet Topologies

Coaxial cable

Distributed CSMA/CD

- 1980 – 1990
- Original Ethernet design at 10 Mbps

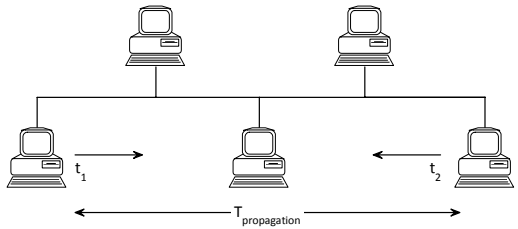
Shared physical bus

- Coaxial cable < 2.5 km

$$\text{End-to-end propagation delay} = \frac{2.5 \text{ km}}{3\times10^{5} \text{ km/sec}} = 8\times10^{-6} \text{ sec}$$

$$\text{Bits transmitted before carrier detect} = (10 \text{ Mb/s})\times(8\times10^{-6} \text{ sec}) = 80 \text{ bits}$$



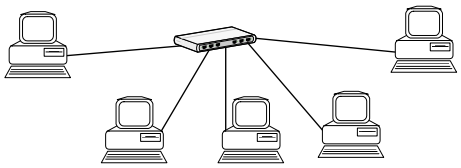


Ethernet Topologies

Passive hub

Centralized CSMA/CD

- 1990 – present
- Fast Ethernet
- 100 Mbps



Logical bus on physical star

- Central hub
- Multiple cables < 100 m
- Each station (STA = node) receives logical OR of all inputs
- Multiple frames ⇒ collision

$$\text{End-to-end propagation delay} = \frac{200 \text{ m}}{3\times10^{8} \text{ m/sec}} \approx 7\times10^{-7} \text{ sec}$$

$$\text{Bits transmitted before carrier detect} = (100 \text{ Mb/s})\times(7\times10^{-7} \text{ sec}) = 70 \text{ bits}$$

Ethernet Topologies

Active hub

Ethernet switch

1995 – present

Physical star

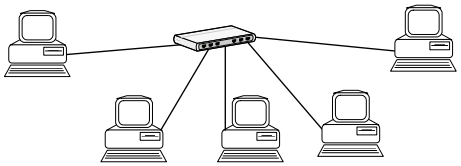
100 Mbps → 1 Gbps → 10 Gbps → 100 Gbps

Fast N × N non-blocking switch

- Hub learns MAC addresses at each switch port
- Each frame directed to port by destination address in frame
- Large output buffer at each port

All stations can send at same time

No collisions



802.3 Ethernet Standards

10 Base 5	10 Mb/s on thick coaxial cable
10 Base 2	10 Mb/s on thin coaxial cable
10 Base T	10 Mb/s on 1 twisted pair
100 Base T	100 Mb/s on 1 twisted pair
100 Base TX	100 Mb/s full duplex on 2 twisted pairs
100 Base FX 100 Base SX 100 Base BX 100 Base LX	100 Mb/s full duplex on optical fibers
1000 Base TX	1 Gb/s full duplex on 2 twisted pairs
1000 Base SX 1000 Base LX 1000 Base BX 1000 Base ZX	1 Gb/s full duplex on 2 optical fibers

Ethernet Frame

Preamble	Start	Dest Address	Src Address	Type or Length	Data	CRC
7 bytes	1 byte	6 bytes	6 bytes	2 bytes	46 – 1500 bytes	4 bytes

Preamble	7 bytes of 10101010 for sync of receivers	
Start	10101011	
Address	Hardware (MAC) address of node 48-bit MAC addresses assigned by OEM and fixed in hardware Broadcast address FF:FF:FF:FF:FF:FF (frame read by all STAs)	
Length	Length of data field (<1500)	
Type	Code identifying protocol in data field Used in most Ethernet systems Type codes > 1536 =0x600	IP = 0x0800 AppleTalk = 0x809B ARP = 0x0806
CRC	CRC-32	

IEEE 802.2

LLC sublayer for 802.3 Ethernet

Based on HDLC

Permits connection oriented services at data link layer

802.2 I/S

DSAP	SSAP	control	data + pad
1 byte	1 byte	2 bytes	42 - 1496 bytes

802.2 U

DSAP	SSAP	control	data + pad
1 byte	1 byte	1 bytes	43 - 1497 bytes

DSAP	Destination service access point (protocol / service at destination)
SSAP	Source service access point (protocol / service at source)
Control	Frame type + SEQ + ACK (I / S frames) Frame type (U frame)

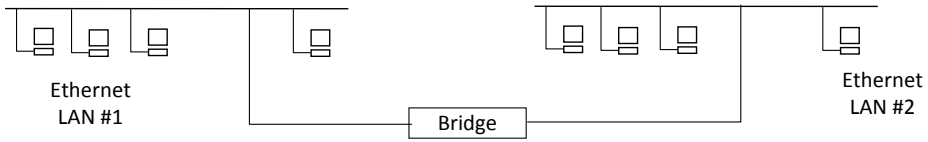
Segmentation

Ethernet bridge

2-port switch

Connects 2 Ethernet segments

Reduces traffic in each segment



Initialization — promiscuous mode

Bridge passes every Ethernet frame

- Listens as destination STA
- Repeats Ethernet frame as source STA

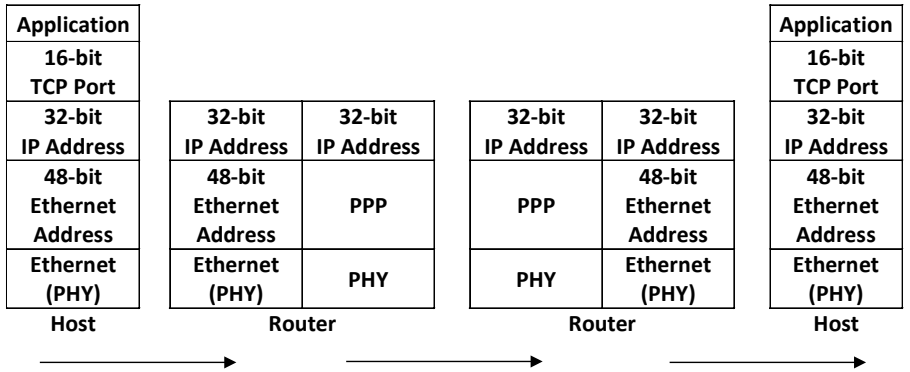
Bridge learns network topology

Builds table of source MAC addresses

Forwards only inter-segment frames

Relationship of Protocol Layers

Typical network



Locate router by IP address (uses default gateway)

Send to router by MAC address

Point-to-point

Locate host by IP address

Send to host by MAC address

Ethernet always uses source / destination Ethernet addresses — not IP addresses

How does host find MAC address for router?

Address Resolution Protocol (ARP)

Look-up MAC address by IP address (RFC 826)

Q: Who has IP = a.b.c.d ? (MAC layer broadcast)

A: I am IP = a.b.c.d with MAC = u:v:w:x:y:z

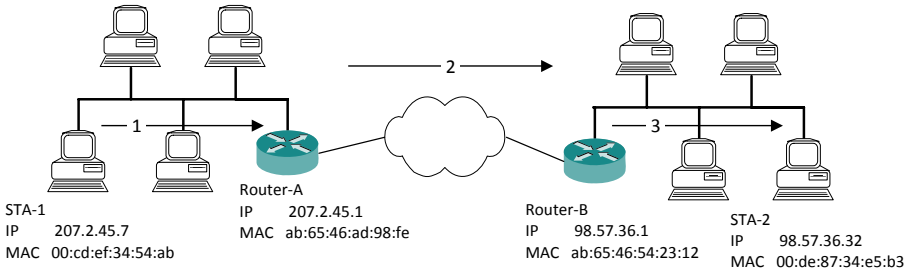
STAs store mappings in arp table

Windows / Linux **`arp -a`** prints arp table

ARP packet fields

HTYPE	Hardware type — MAC protocol
PTYPE	Protocol type — network protocol
HLEN	Hardware length — length in octets of MAC address
PLEN	Protocol length — length in octets of network address
Operation	1= request / 2 = reply
SHA	Sender hardware address
SPA	Sender protocol address
THA	Target hardware address (ignored in requests)
TPA	Target protocol address

Example



Frames between STA-1 and STA-2

	MAC Header	IP Header	IP data	MAC Trailer
1	SRC: 00:cd:ef:34:54:ab DST: ab:65:46:ad:98:fe	SRC: 207.2.45.7 DST: 98.75.36.32	TCP segment	CRC
2	PPP	SRC: 207.2.45.7 DST: 98.75.36.32	TCP segment	CRC
3	SRC: ab:65:46:54:23:12 DST: 00:de:87:34:e5:b3	SRC: 207.2.45.7 DST: 98.75.36.32	TCP segment	CRC

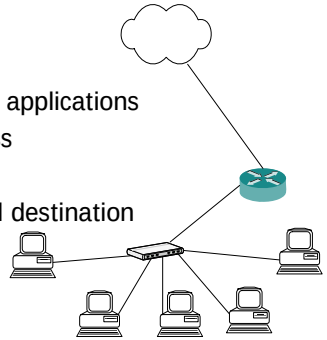
Advanced Switch / Router Interactions

Standard model

Switch organizes STAs into LAN

Intra-LAN traffic

- STAs use IP addresses as names for TCP/IP applications
- STAs use ARP to translate IP to MAC address
- STAs send frames on LAN by MAC address
- Packets contain MAC and IP address of local destination



Router organizes LAN into Internet AS

Inter-LAN traffic

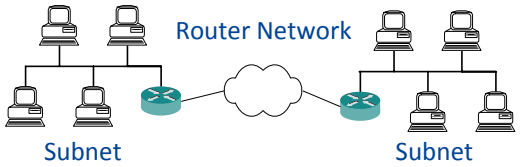
- STAs use IP addresses as names for TCP/IP applications
  - Local MAC addresses not available for remote STAs
- STAs send frames via router
- Packets contain
  - MAC address of router
  - IP address of remote destination

Advanced Switch / Router Interactions

Virtual LAN

Standard subnet

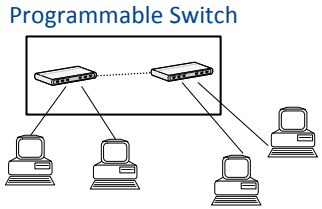
subnet-1 and subnet-2 are LAN broadcast domains



Virtual LAN (VLAN)

LAN switch configured to partition nodes into subnets

No router needed for subnetting



Advanced Switch / Router Interactions

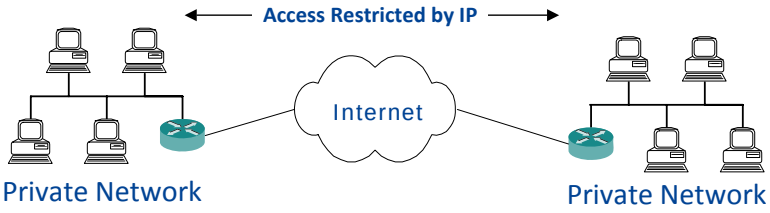
Virtual Private Network

Virtual Private Network (VPN)

Private network implemented on public infrastructure

Access to private networks restricted by IP

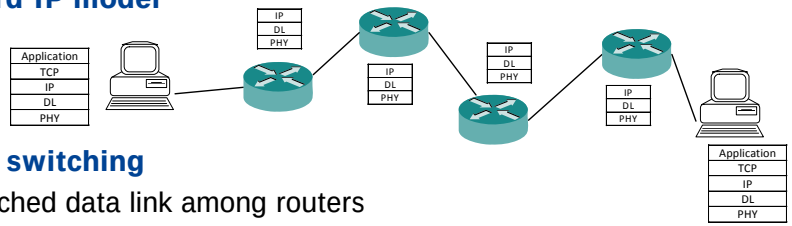
Possible encryption of data over public infrastructure



Advanced Switch / Router Interactions

Layer 3 Switching

Standard IP model



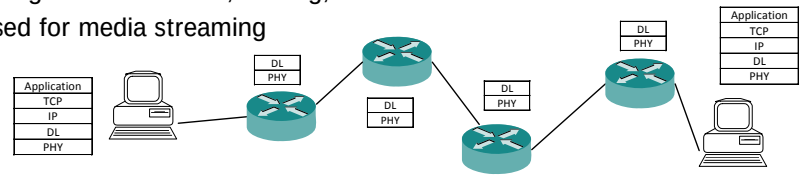
Layer 3 switching

Switched data link among routers

- Connection-oriented virtual circuit network
- Frame Relay, ATM, label switching, ...

Traffic crosses router network at layer 2

- Saves time of layer 3 processing
- Datagram read / write, routing, TTL
- Used for media streaming



Advanced Switch / Router Interactions

Label switching

Multiprotocol Label Switching (MPLS)

MAC Header	MPLS	IP Header	TCP Header	Data	MAC trailer

Header fields

One or more headers per frame — "stack" of labels

Label	20-bit ID
Traffic Class	3-bit QoS (quality of service) field
Stack flag	1-bit — if set, current label is last of "stack" of labels for frame
TTL	8-bit time to live field

Connection-oriented

Set (reserve) router path before data traffic begins

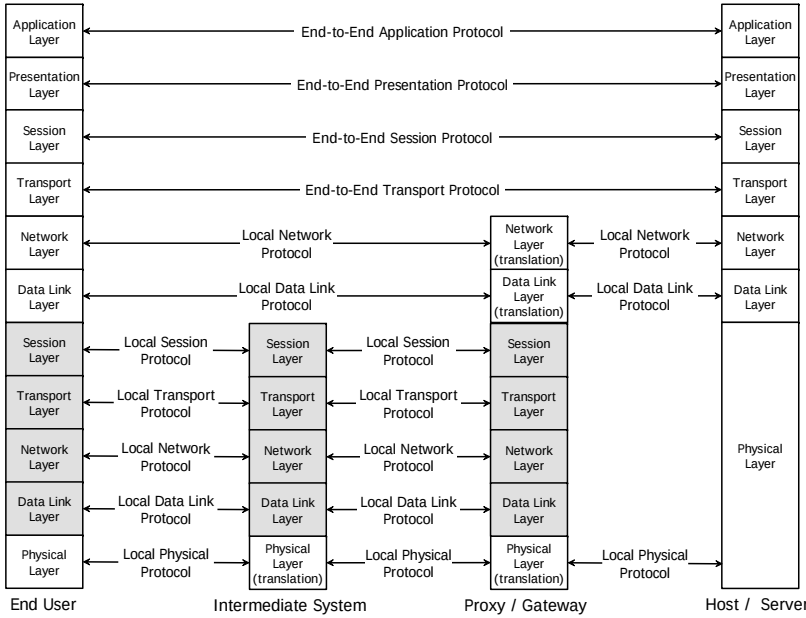
Label Distribution Protocol (LDP)

RSVP-TE — extension of Resource Reservation Protocol (RSVP)

MPLS-aware routers

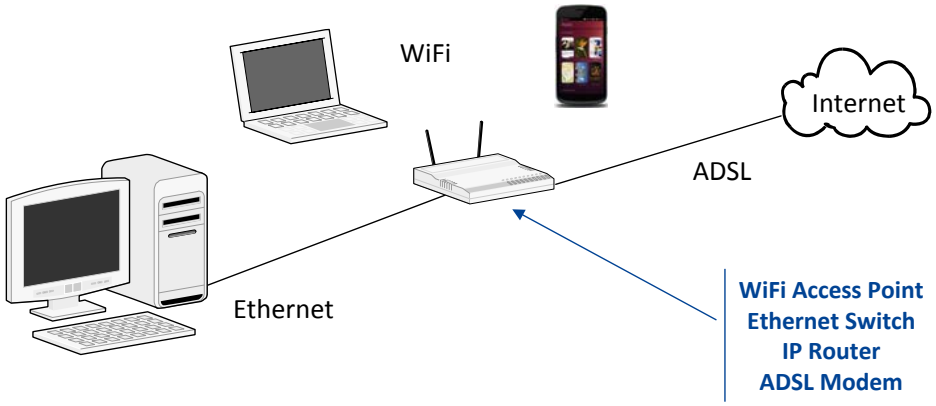
Forward frames on preset route by label ID

Tunneling in the OSI Model



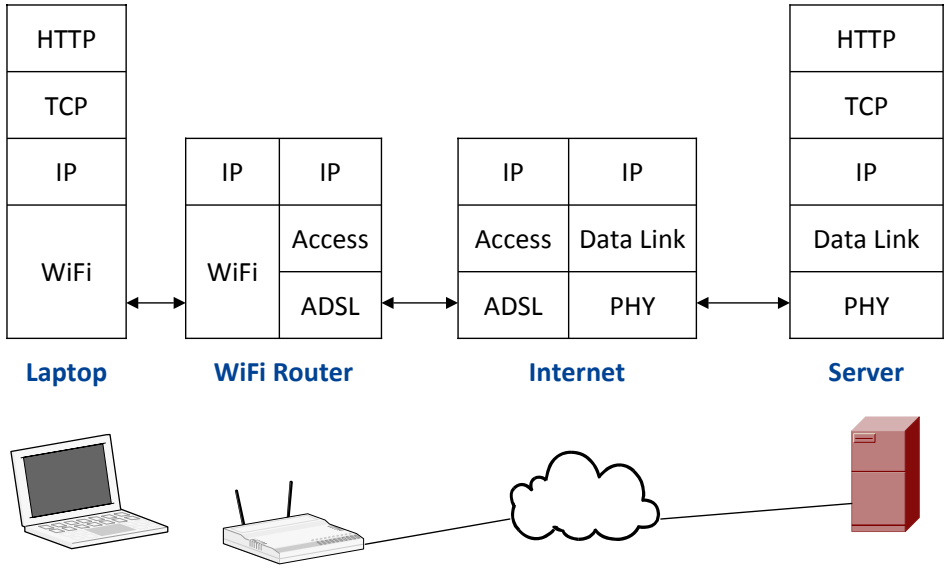
Small Office / Home Office (SOHO)

LAN (Local Area Network) to WAN (Wide Area Network)



Ethernet	802.3	Cable-based **LAN** protocol defined at **DATA LINK** and **PHY** layers
WiFi	802.11	Wireless **LAN** protocol defined at **DATA LINK** and **PHY** layers
ADSL	G.992.5	Cable-based transmission protocol defined at **PHY** layer

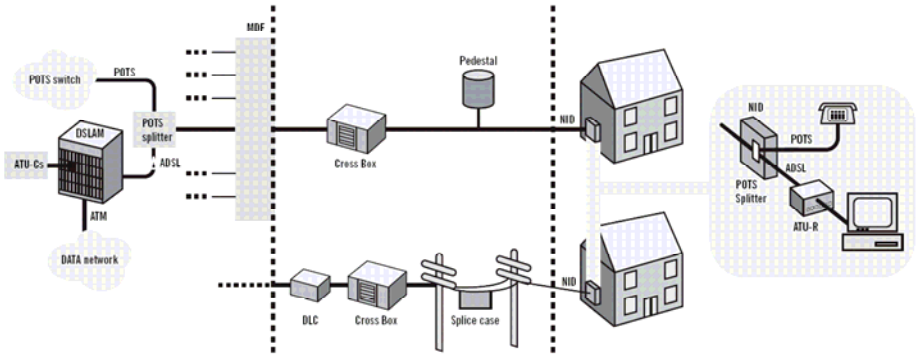
Laptop Browser to Web Server — Simplified View



ADSL — Asymmetric Digital Subscriber Line

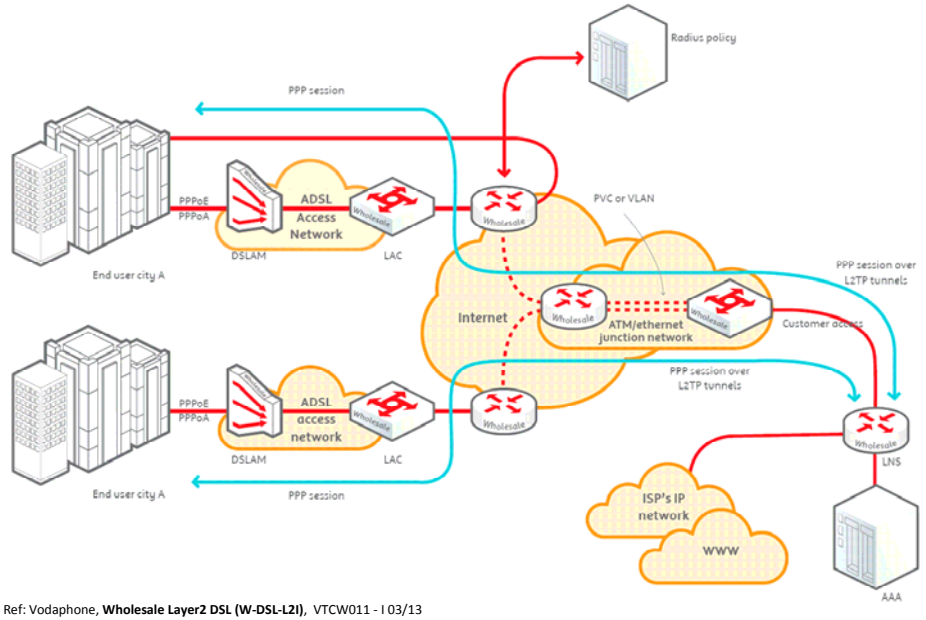
High speed transmission on standard voice line

- POTS — plain old telephone service
- 24 Mbps downstream
- 3.3 Mbps upstream



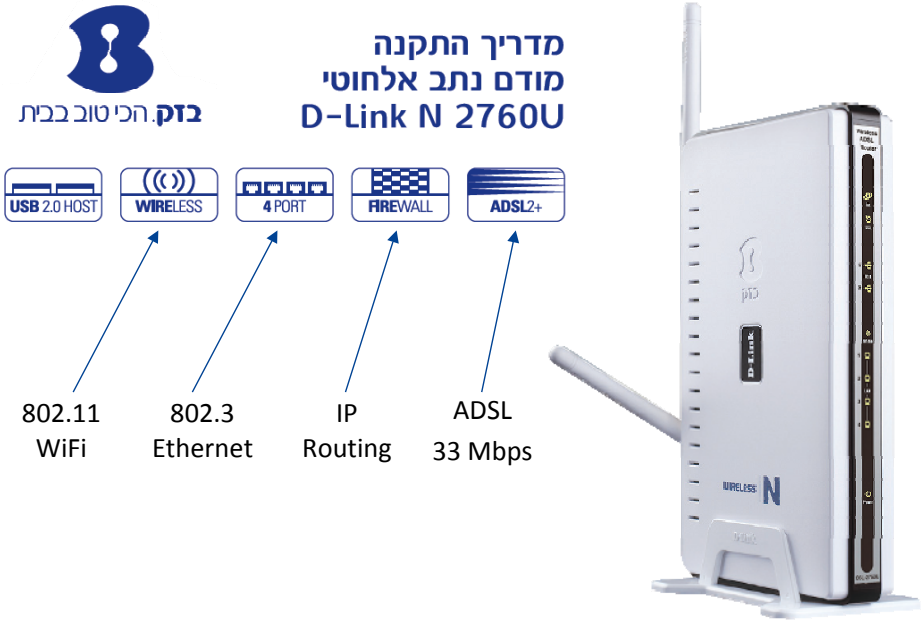
Ref: JDSU, **ADSL Technology**, JDS Uniphase Corporation, 2005

ADSL Access Network

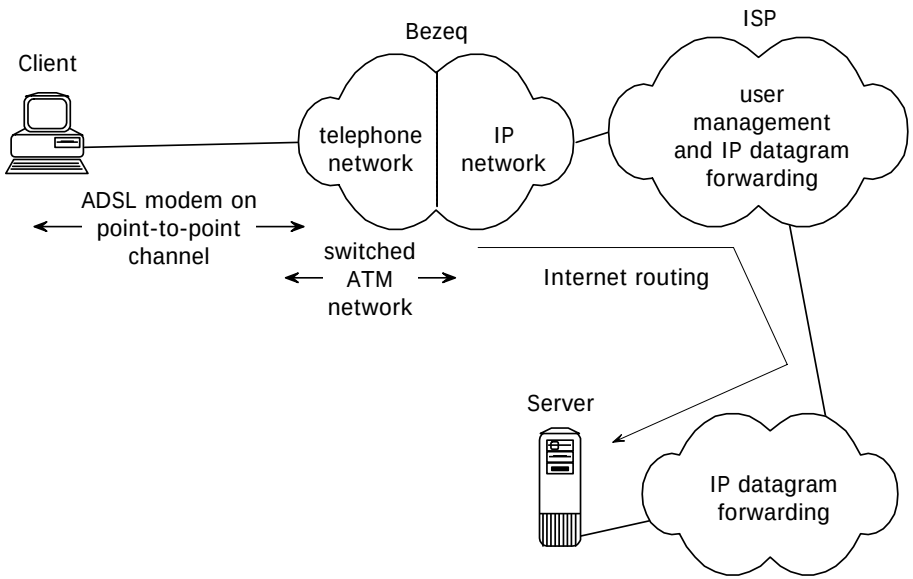


Ref: Vodaphone, **Wholesale Layer2 DSL (W-DSL-L2I)**, VTCW011 - I 03/13

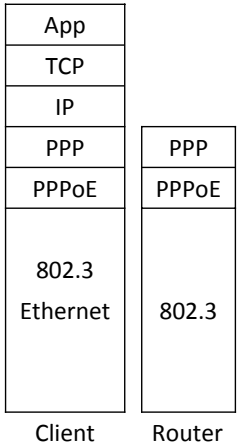
Typical Bezeq ATU-R



Fast Internet Access



Fast Internet Protocols — Typical Campus Case



PPP

Point to Point Protocol

Logon + connection management

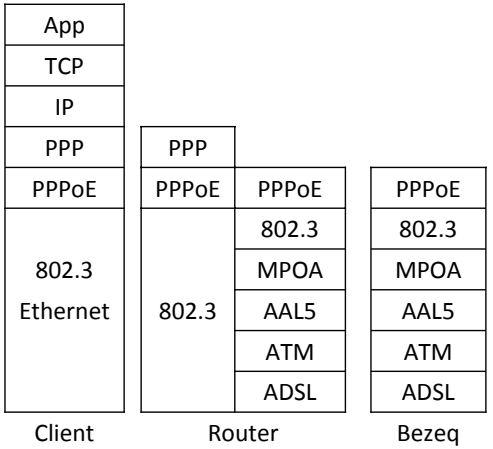
PPPoE

PPP over Ethernet

Virtual point-to-point connection over shared LAN

Client opens private session with ISP

Fast Internet Protocols — Typical Campus Case



ATM

Asynchronous Transfer Mode

Data Link protocol for broadband telephone services

Permits real time QoS

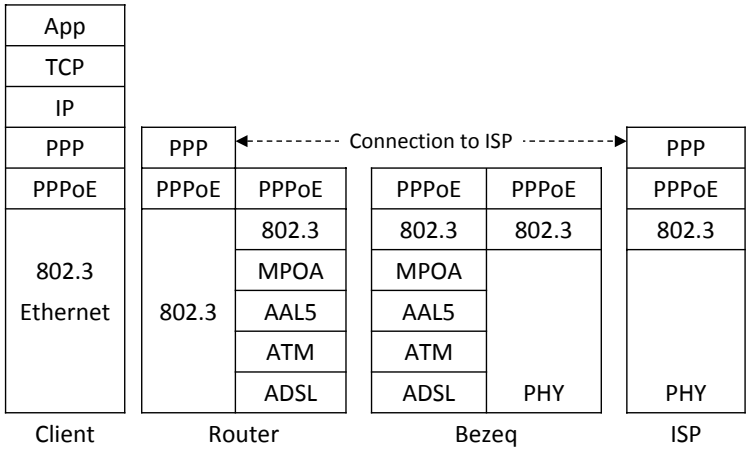
MPOA + AAL5

Adaptation protocols for ATM

ADSL

Physical bit transmission

Fast Internet Protocols — Typical Campus Case



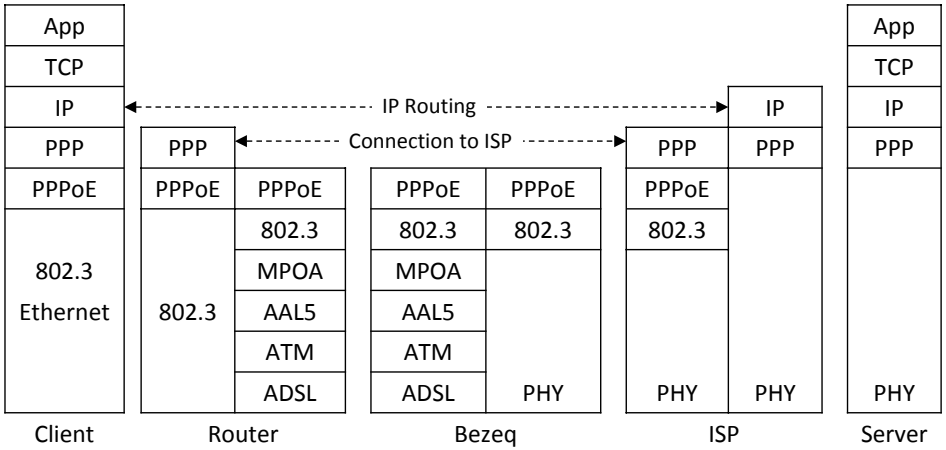
Connection to ISP

Client runs Network Control Protocol (NCP) over PPP

CHAP (challenge handshake authentication protocol) — User Name + Password

ISP authorizes user and engages IP forwarding

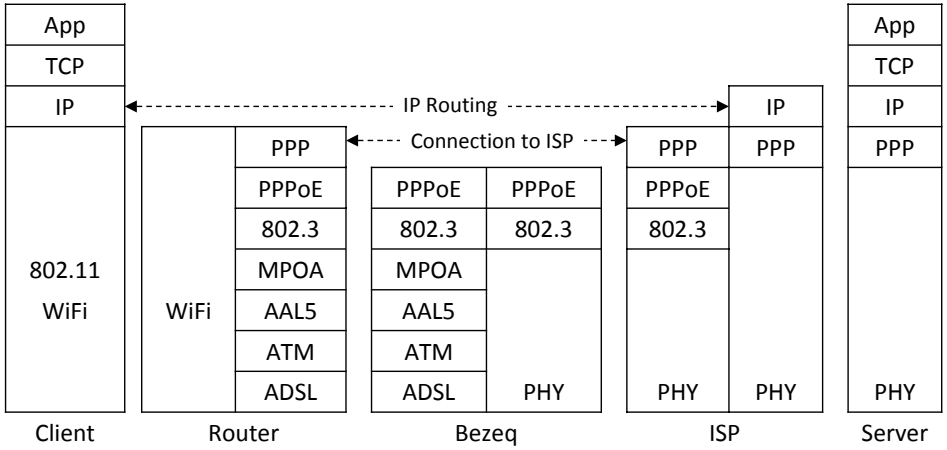
Fast Internet Protocols — Typical Campus Case



IP forwarding

ISP forwards IP datagrams to server via Internet backbone

Fast Internet Protocols — Typical SOHO Case



Router/modem initiates connection to ISP

- Runs NCP over PPP over PPPoE over Ethernet
- Router provides always-on Internet access over WiFi + Ethernet