Computer Networking

Introduction

Fabrice Theoleyre
Fabrice.Theoleyre@imag.fr

http://drakkar.imag.fr/users/Fabrice.Theoleyre/
From the slides of A.Duda

Objective

- Fundamentals in Networking
  - TCP / IP
  - LAN
  - Routing
  - WLAN
  - Network Programming
  - DNS /SNMP
- What is networking for you?

Organization

- Organization
  - 40h
  - 15h Lectures
  - 25h Labs / exercices
- Exam
  - No document: why?
- Lectures
  - http://duda.imag.fr/m1/
- Labs
  - D200
  - Give a pratical knowledge
  - Every lab report will receive a grade
- Prof
  - Fabrice Theoleyre & Andrzej Duda
  - M. Nassiri for Tutorials and lab sessions

References

- Tanenbaum, Computer Networks
- Keshav, An Engineering approach to computer networking
- Kurose,Computer Networking: a top down approach

What kind of network?

- Voice (telecommunications)
  - GSM, UMTS, GPRS
  - POTS
- Data networks
  - IP
  - ATM
  - Fiber Channel
  - Ethernet
  - SDH / SONET
- What are the differences?

Data networks

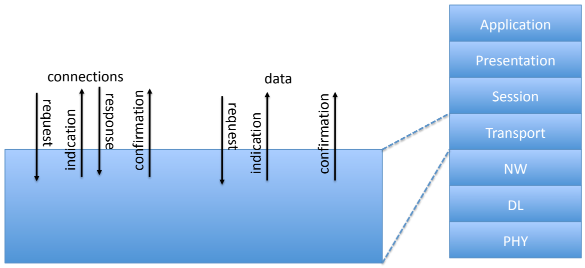
- Digital?
  - A large volume of data to transfer
  - P2P, backups, ERP, etc.
- Challenges
  - Delay
  - Bandwidth
  - Heterogeneity

OSI

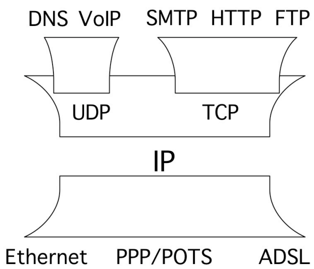
Application	HTTP; DNS; FTP; X11; telnet
Presentation	Xdr; MIME
Session	Http session cookies
Transport	TCP; UDP; SCTP
Network	IP
Link layer	Ethernet 802.11; PPP, SLIP
Physical layer	RS-232

OSI

- Inter-layers communication



IP is not OSI-compliant



An example: http

- Directory service
- PDUs / Headers / Payload description
- URL
  - ✓ service://site[:port#]/path/to/file?request+args
- Addresses conversion
- Multiplexing
- Routing

Take a look inside

- Ethereal (wireshark)

Brief history

- 1970
  - ✓ Deployment of ALOHA
- 1974
  - ✓ Specifications of TCP (Vinton Cerf)
  - ✓ Bob Metcalfe and David Boggs present Ethernet
- 1978
  - ✓ TCP is split into TCP and IP ⇒ TCP/IP
- 1980
  - ✓ ARPANET connects 500 stations FTP RFC
- 1981
  - ✓ ARPANET switches to TCP/IP
- 1982
  - ✓ SMTP RFC (by Jonathan B. Postel)

History

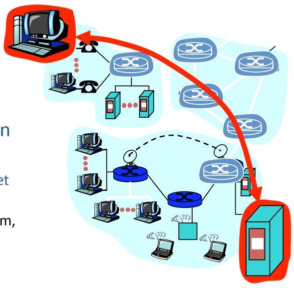
- 1983
  - ✓ Paul Mockapetris invents the DNS
- 1990
  - ✓ Most computers are TCP/IP capable
- 1991
  - ✓ Tim Berners-Lee at CERN develops the WWW
- 1993
  - ✓ Mosaic v 1.0 (web browser)
- 1995
  - ✓ Introduction of Fast Ethernet (100Mb/s)
- 2004
  - ✓ 950 000 000 Internet users
  - ✓ IP telephony available in France
- 2005
  - ✓ IPTv to the home in France
- ?: Telcos switch to IPv4 , v6?

Overview

- Network architectures
  - ✓ recall on the Internet
  - ✓ protocol architectures
    - how entities cooperate?
  - ✓ interconnection structure
    - which entities are connected?
  - ✓ related protocols
    - how and where different functionalities are implemented?
- Performance
  - ✓ transmission
  - ✓ propagation
  - ✓ bandwidth-delay product
  - ✓ queueing delay

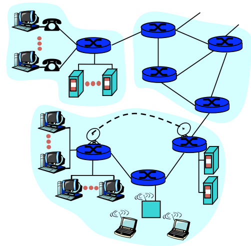
Inside the Internet

- Between end systems
  - ✓ TCP protocol for reliable transmission
- Inside the network core
  - ✓ IP protocol: forwarding packets between routers
- Between routers or between end system and router
  - ✓ high speed link: ATM, POS (Packet over SONET), satellite links
  - ✓ access network: Ethernet, modem, xDSL, HFC



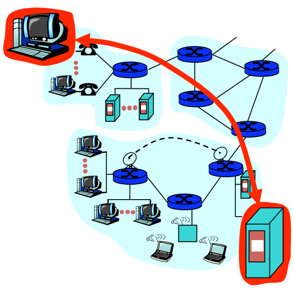
Network structure

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



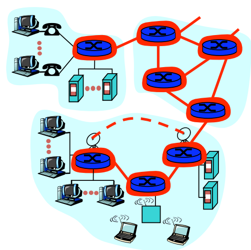
The network edge:

- end systems (hosts):
  - ✓ run application programs
  - ✓ e.g., WWW, email
  - ✓ at "edge of network"
- client/server model
  - ✓ client host requests, receives service from server
  - ✓ e.g., WWW client (browser)/ server, email client/server
- peer-peer model:
  - ✓ symmetric host interaction
  - ✓ e.g. teleconferencing



The Network Core

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



Circuit switching

- Reservation
  - ✓ Call setup / tear down
  - ✓ Alone for this segment
  - ✓ Supervised
  - ✓ Connection oriented
    - Ordered, reserved bandwidth

Packet switching

- No more circuit but packets
  - ✓ Voice → data networks
  - ✓ A space is reserved in the data to specify the destination
  - ✓ Packets travel independently from each other
  - ✓ Different connections share the same logical & physical links
- Fundamental difference!

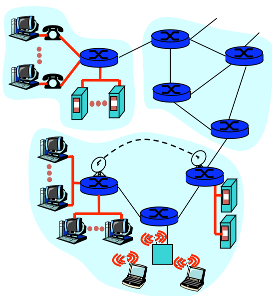
Access networks and physical media

How to connect end systems to edge router?

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

Characteristics:

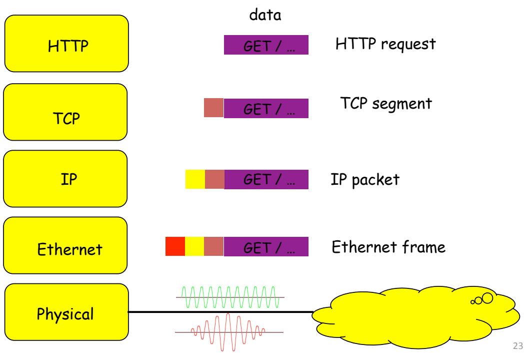
- bandwidth (bits per second) of access network
- shared or dedicated



Internet design principles

- Cerf and Kahn's internetworking principles:
  - ✓ minimalism, autonomy - no internal changes required to interconnect networks
  - ✓ best effort service model
  - ✓ stateless routers
  - ✓ decentralized control
- Small number of layers
  - ✓ compromise between performance and flexibility
    - thin layers encourage flexibility, but increases overhead
- Define today's Internet architecture

TCP/IP Architecture



Application Layer

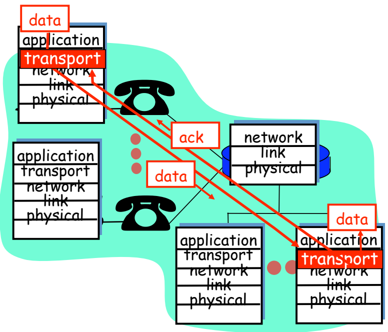
- Application layer supports network application
  - applications that are distributed over the network
  - applications that communicates through the network
- Many known protocols
  - FTP: file transfer
  - SMTP: email protocol
  - HTTP:web protocol
- An application uses UDP or TCP, it is a designer's choice
- Interface with the transport layer
  - use for example the socket API: a library of C functions
  - socket also means (IP address, port number)

Transport Layer

- Why a transport layer ?
  - transport layer = makes network service available to programs
  - is end-to-end only, not in routers
  - Ports for application differentiation
- In TCP/IP there are two transport protocols
  - UDP (user datagram protocol)
    - unreliable
    - offers a datagram service to the application (unit of information is a message)
  - TCP (transmisssion control protocol)
    - reliable
    - offers a stream service (unit of information is a byte)

Layering: logical

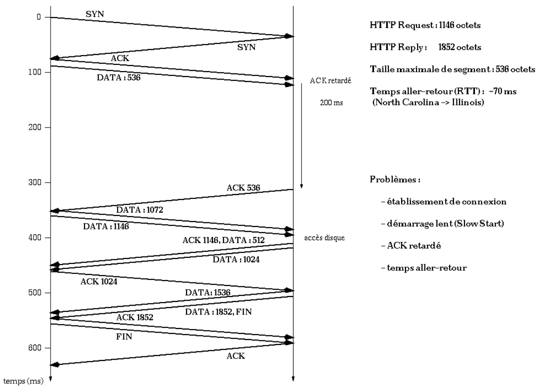
- E.g.: transport
- take data from app
- add addressing, reliability check info to form "datagram"
- send datagram to peer
- wait for peer to ack receipt
- analogy: post office



Layering

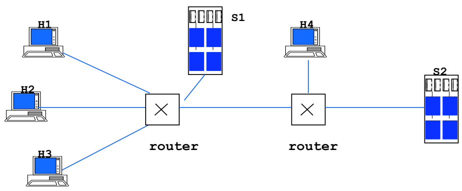
- A tunnel
  - One start point / One end point
  - A virtual pipe
- Transportation analogy
  - I could create a tunnel from me to the destination
  - I just put my object into the tunnel, and the destination gets it
  - This tunnel is only for me and the destination
    - Not *shared*

TCP

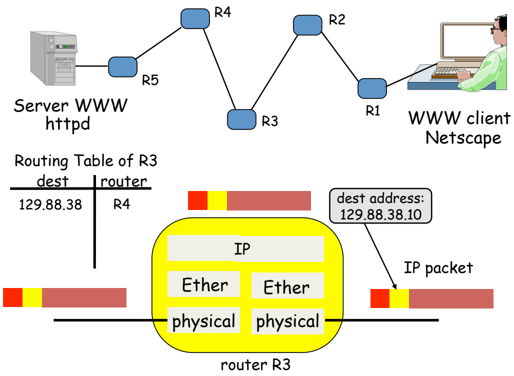


Network Layer

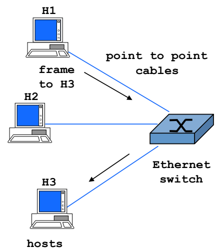
- Set of functions required to transfer packets end-to-end (from host to host)
  - hosts are not directly connected - need for intermediate systems
  - examples: IP, Appletalk, IPX
- Intermediate systems
  - routers: forward packets to the final destination
  - interconnection devices



IP



Physical Layer
Data Link Layer

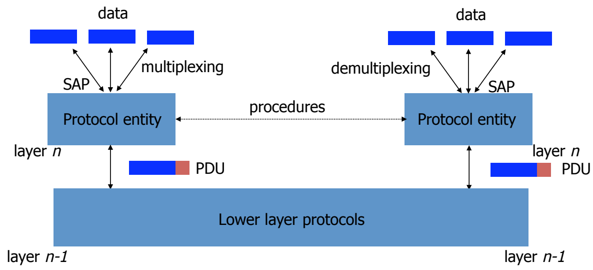


- Physical transmission = Physical function
  - ✓ bits <-> electrical / optical signals
  - ✓ transmit individual bits over the cable: modulation, encoding
- Frame transmission = Data Link function
  - ✓ bits <-> frames
  - ✓ bit error detection
  - ✓ packet boundaries
  - ✓ in some cases: error correction by retransmission (802.11)
- Modems, xDSL, LANs, FR (below IP)

Protocol architectures

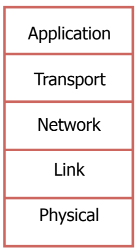
- Protocol entity
  - ✓ provides a set of services, eg.
    - connect, send
  - ✓ data multiplexing/demultiplexing
  - ✓ construction/analysis of PDUs
  - ✓ execution of procedures
- Protocol unit (PDU)
  - ✓ header: control functions
  - ✓ opaque data
- Procedures
  - ✓ actions to perform protocol functions: e.g. lost packet retransmission

Protocol architecture

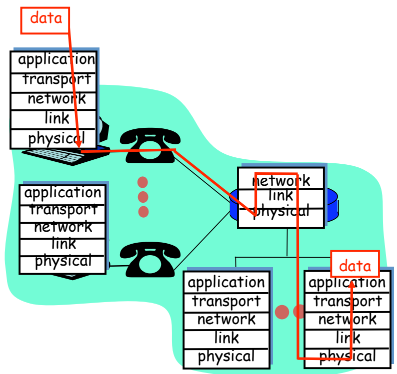


Internet protocol stack

- **Application**: supporting network applications
  - ✓ FTP, SMTP, HTTP, OSPF, RIP
- **Transport**: host-host data transfer
  - ✓ TCP, UDP
- **Network**: routing of datagrams from source to destination
  - ✓ IP
- **Link**: data transfer between neighboring network elements
  - ✓ PPP, Ethernet
- **Physical**: bits "on the wire"

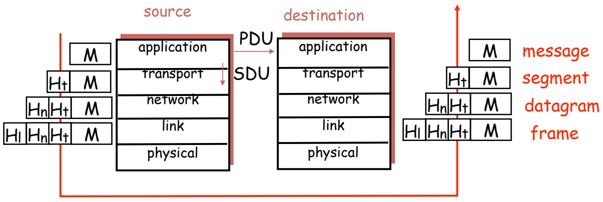


Layering: physical communication

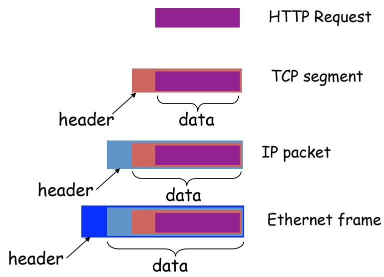


Protocol layering and data

Each layer takes data from above
- adds header information to create new data unit
- passes new data unit to layer below



Encapsulation

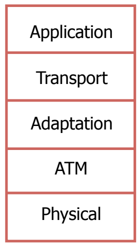


ISO Model

Application	Common functions
Presentation	Interchangable formats
Session	Organizing dialog
Transport	Reliable transmission
Network	Forwarding in the network
Data link	Transmission between two nodes
Physical	Signal transmission

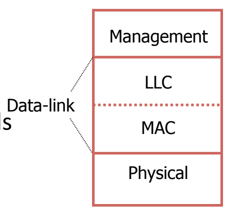
ATM protocol stack

- **Application**: native applications, other protocols
  - ✓ LAN Emulation, IP, Signaling
- **Transport**: host-host data transfer
  - ✓ SSCOP
- **Adaptation**: adapt the ATM layer to different types of applications
  - ✓ circuit emulation, real-time data
  - ✓ AAL5 suitable for IP traffic
- **ATM**: cell switching over virtual circuits
- **Physical**: bits "on the wire", usually fiber

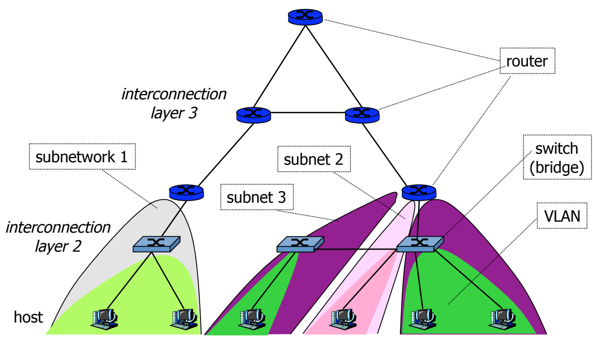


LAN stack

- **Management**: e.g. construct forwarding tables
  - ✓ SNAP: Spanning Tree protocol
- **LLC**: multiplex different protocols
  - ✓ IP, IPX, SNAP
- **MAC**: medium access
  - ✓ 802.3 (Ethernet), 802.4 (Token Ring), 802.5 (Token Bus), 802.11 (Wi-Fi)
- **Physical**: bits "on the wire"



Interconnection structure - layer 3



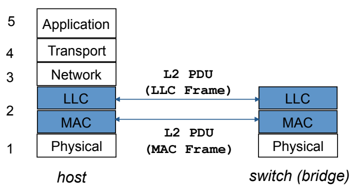
Interconnection at layer 2

- Switches (bridges)
  - ✓ interconnect hosts
  - ✓ logically separate groups of hosts (VLANs)
  - ✓ managed by one entity
- Type of the network
  - ✓ broadcast
- Forwarding based on MAC address
  - ✓ flat address space
  - ✓ forwarding tables: one entry per host
  - ✓ works if no loops
    - careful management
    - Spanning Tree protocol
  - ✓ not scalable

Interconnection

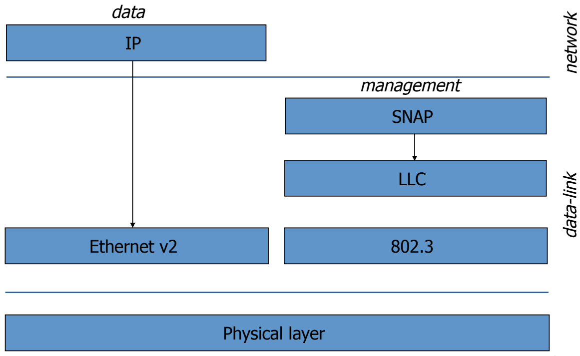
- L2
  - ✓ Hub
    - Repeat
  - ✓ Bridge
    - Connect two networks
    - Selective repeat
  - ✓ Switch
    - Forwarding
- L3
  - ✓ Router
    - Finds a IP route
    - Scalable

Protocol architecture



- Switches are layer 2 intermediate systems
- Transparent forwarding
- Management protocols (Spanning Tree, VLAN)

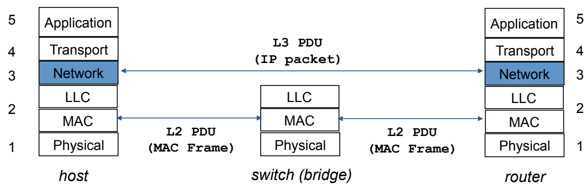
Protocols



Interconnection at layer 3

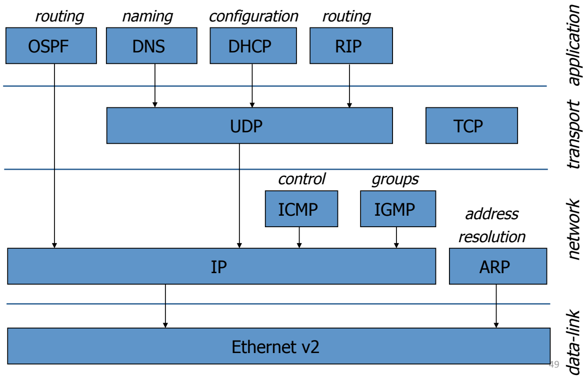
- Routers
  - ✓ interconnect subnetworks
  - ✓ logically separate groups of hosts
  - ✓ managed by one entity
- Forwarding based on IP address
  - ✓ structured address space
  - ✓ routing tables: aggregation of entries
  - ✓ works if no loops - routing protocols (IGP - Internal Routing Protocols)
  - ✓ scalable inside one administrative domain

Protocol architecture

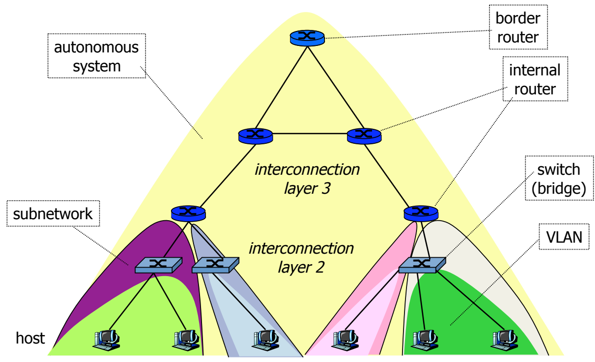


- Routers are layer 3 intermediate systems
- Explicit forwarding
  - ✓ host has to know the address of the first router
- Management protocols (control, routing, configuration)

Protocols



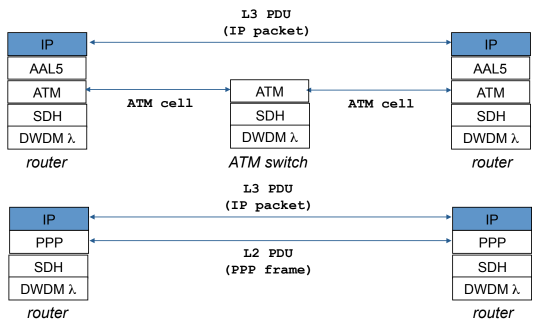
Autonomous systems



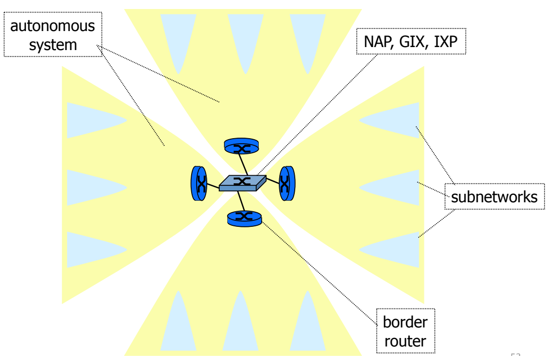
Overlaid stacks? Long-haul links

- Fiber at physical layer (SONET/SDH)
  - ✓ Dense Wave Division Multiplexing (DWDM)
    - one color of the light λ
- Different technologies
  - ✓ ATM
  - ✓ Frame Relay
  - ✓ POS (Packet over SONET/SDH)
- Type of the network
  - ✓ NBMA (Non Broadcast Multiple Access) or point-to-point
- Complex protocol hierarchies
  - ✓ IP over ATM

Protocol architecture



Internet



LIX: London

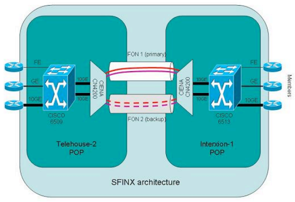
Over 280 members

240Gbps peak traffic

…

UK Broadband, at&t, webtapestry, Nerim, n9uf telecom, ftgroup, free, FUJITSU

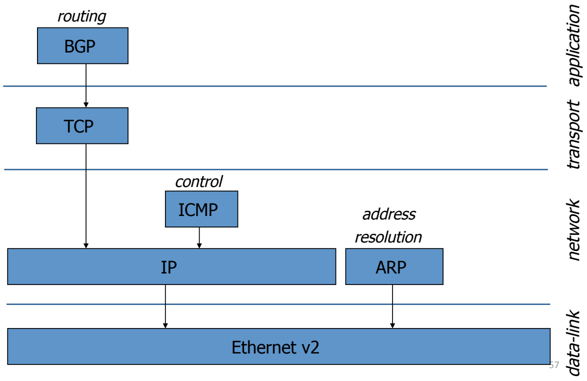
SPHINX



Interconnection of AS

- Border routers
  - ✓ interconnect AS
- NAP or GIX, or IXP
  - ✓ exchange of traffic - peering
- Route construction
  - ✓ based on the path through a series of AS
  - ✓ based on administrative policies
  - ✓ routing tables: aggregation of entries
  - ✓ works if no loops and at least one route - routing protocols (EGP - External Routing Protocols)

Protocols

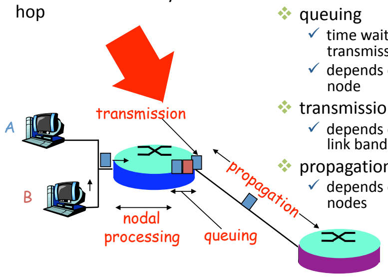


Performance

- Bit Rate of a transmission system
  - ✓ bandwidth, throughput
  - ✓ number of bits transmitted per time unit
  - ✓ units: b/s or bps, kb/s = 1000 b/s, Mb/s = 10e+06 b/s, Gb/s=10e+09 b/s
  - ✓ OC3/STM1 - 155 Mb/s, OC12/STM4 - 622 Mb/s, and OC48/STM-16 - 2.5 Gb/s, OC192/STM-48 10 Gb/s
- Latency or Delay
  - ✓ time interval between the beginning of a transmission and the end of the reception
  - ✓ RTT - Round-Trip Time

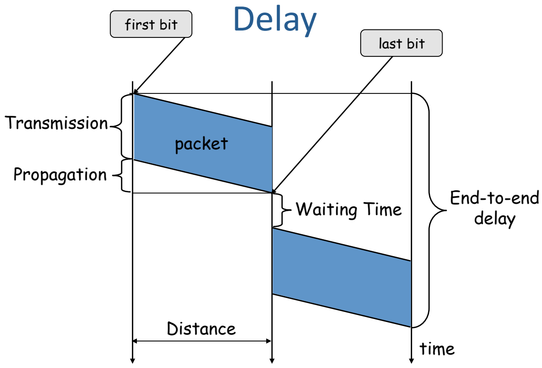
Delay in packet-switched networks

- packets experience delay on end-to-end path
- four sources of delay at each hop



- nodal processing
  - ✓ check bit errors
  - ✓ Determine output link
- queuing
  - ✓ time waiting at output link for transmission
  - ✓ depends on congestion level of node
- transmission:
  - ✓ depends on packet length and link bandwidth
- propagation:
  - ✓ depends on distance between nodes

Delay



Performance

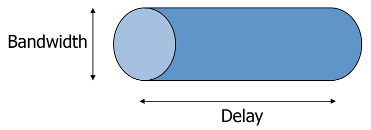
- Latency
  - Latency = Propagation + Transmission + Wait
  - Propagation = Distance / Speed
    - copper : Speed = 2.3×10^8 m/s
    - glass : Speed = 2×10^8 m/s
    - Transmission = Size / BitRate
- 5 µs/km
- New York - Los Angeles in 24 ms
  - request - 1 byte, response - 1 byte: 48 ms
  - 25 MB file on 10 Mb/s: 20 s
- World tour in 0.2 s

Example

- At time 0, computer A sends a packet of size 1000 bytes to B; at what time is the packet received by B (speed = 2e+08 m/s)?

distance	20 km	20000 km	200 m	20 m
bit rate	10kb/s	1 Mb/s	100 Mb/s	4 Gb/s
Propagation	0.1ms	100 ms	1µs	0.1µs
Transmission	800 ms	8 ms	80 µs	2 µs
latency	800.1ms	108ms	80.1 µs	2.1 µs
	modem	*satellite*	*LAN*	*Fiber Channel*

Bandwidth-Delay Product



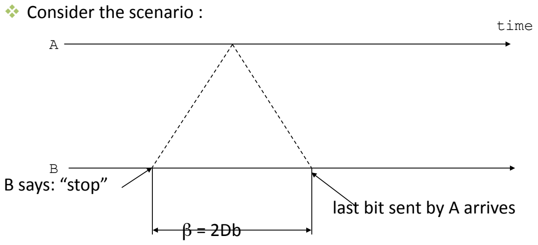
- Bandwidth-Delay product
  - how many bits should we send before the arrival of the first bit?
  - good utilization - keep the pipe filled!

Bandwidth-Delay Product

- File transfer: 1 Mbit, 100 ms delay
  - 1 Mb/s link, $D_{\times}b$ = 0.1 Mbit
    - 10 transmissions, 10% each time
  - 1 Gbit/s link, $D_{\times}b$ = 100 Mbit
    - 1 transmission, pipe not filled

Bandwidth-Delay Product

- Consider the scenario :



- β = maximum number of bits B can receive after saying stop
- large β means: delayed feedback
- amount of data "in the pipe"

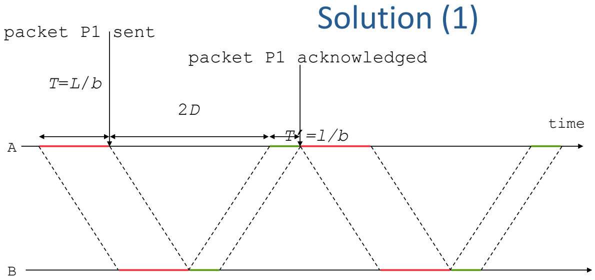
A Simple Protocol: Stop and Go

- Packets may be lost during transmission: bit errors due to channel imperfections, various noises.
- Computer A sends packets to B
- B returns an acknowledgement packet immediately to confirm that B has received the packet
- A waits for acknowledgement before sending a new packet
- If no acknowledgement comes after a delay T1, then A retransmits

A Simple Protocol: Stop and Go

- Question: What is the maximum throughput assuming that there are no losses?
- notation:
  - ✓packet length = L, constant (in bits);
  - ✓acknowledgement length = l, constant
  - ✓channel bit rate = b;
  - ✓propagation = D
  - ✓processing time = 0

Solution (1)



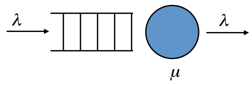

```
cycle time = T + 2D + T'
useful bits per cycle time = L
throughput = Lb / (L + l + 2Db)= b /(ω + β/L)
with ω=(L+l)/L=overhead and β=2Db=bandwidth-delay product
```


Solution (2)

distance	20 km	20000km	200m	20 m
bit rate	10kb/s	1 Mb/s	100 Mb/s	4 Gb/s
propag	0.1ms	100 ms	1 µs	0.1µs
transmission	800 ms	8 ms	80 µs	2 µs
Reception	0.8 s	108 ms	80.1 µs	2.1 µs
	modem	satellite	LAN	FC
β=2Db	2 bits	200 kb	200 b	800 b
Efficiency	99.98%	3.8%	97.56%	90.90%
throughput	efficiency * BW			

Waiting time

- Queueing system M/M/1
  - ✓interarrival times ~ exponentially distributed
  - ✓service times ~ exponentially distributed
  - ✓arrival rate λ, service rate μ, utilization ρ= λ/μ
  - number of packets N, waiting time T



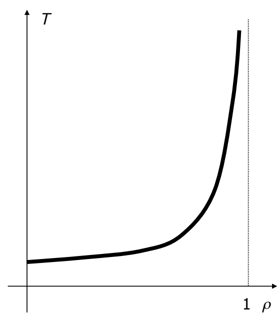
$$N = \frac{\rho}{(1-\rho)} \qquad T = \frac{1}{\mu(1-\rho)} \qquad T = \frac{N}{\lambda}$$

Waiting time

- Average packet length 1500 bytes
  - ✓ link with 1 Mb/s bit rate (propagation = 0)
    - transmission time 12 ms
    - service rate 83 packet/s

λ	[p/s]	10	40	60	70
$1/\lambda$	[ms]	100	25	16	14
T	[ms]	13	23	43	76

Waiting time



Summary

- Network architectures
  - protocol architectures
    - different protocol stacks, overlaid stacks
  - interconnection structure
    - switches, routers
  - related protocols
    - complex protocol families
- Performance
  - transmission
  - propagation
  - bandwidth-delay product
  - queueing delay

Explain in few lines the following concepts (1):

1. Functions of the physical layer
2. Functions of the link layer
3. Functions of the network layer
4. Functions of the transport layer
5. Circuit switching
6. Packet switching
7. Connection oriented communication
8. Switch

Explain in few lines the following concepts (1):

1. Reliable transport
2. Router
3. Client-server computing
4. Protocol
5. Error recovery
6. Flow control
7. PDU

Problem 1

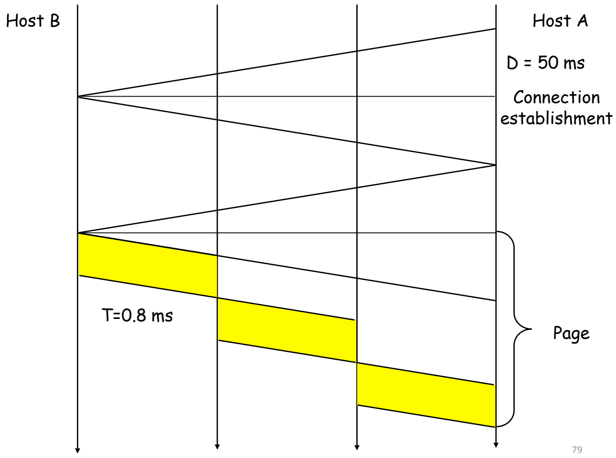
- Two hosts are connected via the Internet through 09 routers. The distance between the hosts is 10000 km. All the links in the network have 100 Mb/s bit rate.
- The user working on host A downloads a page of 1000 bytes from server B. What is the total time of the download (between the click and the instant when the page is downloaded)?
- Assume that:
  - HTTP uses a TCP connection with the MTU of 1460 bytes. Other TCP parameters are supposed to be known.
  - We ignore processing and waiting times, as well as the transmission time of short segments (short means that they are less than 1000 bytes), for instance connection establishment segments, ACKs, HTTP request. We also ignore the HTTP header attached to the page contents.

Problem 2

- Two hosts are connected via the Internet through 19 routers. The distance between the hosts is 10000 km. All the links in the network have 100 Mb/s bit rate.
- The user working on host A downloads a page of 1000 bytes from server B. What is the total time of the download (between the click and the instant when the page is downloaded)?
- Assume that:
  - HTTP uses a TCP connection with the MTU of 1460 bytes. Other TCP parameters are supposed to be known.
  - We ignore processing and waiting times, as well as the transmission time of short segments (short means that they are less than 1000 bytes), for instance connection establishment segments, ACKs, HTTP request. We also ignore the HTTP header attached to the page contents.

What are the *good* questions?

- What are the different steps?
- Direction of the transfert?
- What can be neglected?
- How many packets must I send?
- How does a packet transmission process in a router? In a link?



Solution

- First case
  - Delay = 3 x 50 ms + 50 ms + 10 x 0.8 = 200.8 ms
  - Throughput = 1000 x 8 bits / 200.8 ms = 39.84 kb/s
- Second case
  - Delay = 3 x 50 ms + 50 ms + 20 x 0.8 = 216 ms
  - Throughput = 1000 x 8 bits / 216 ms = 37 kb/s