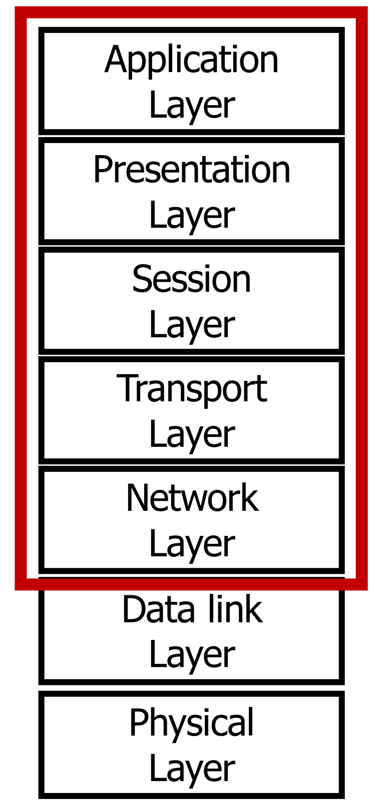
Computer Networks

Introduction

Chapters

1. **Introduction**
   - **Computer networks**
   - **Classification**
   - **Standardization**
2. Switching
3. Protocols
4. Application layer
5. Web services
6. Publish/Subscribe
7. Distributed hash tables
8. Time synchronization
9. Transport layer
10. UDP / RTP / TCP
11. Network layer
12. Internet protocol
13. Quality of service

Top-Down-Approach

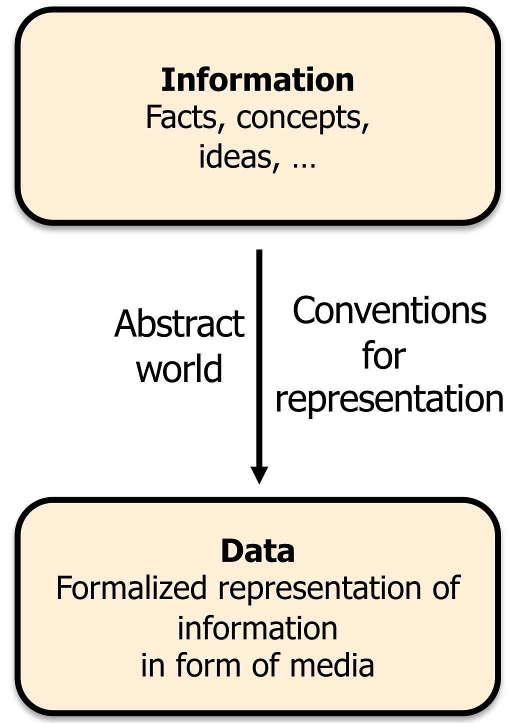


Introduction

What to Communicate: Information, Data

Technische Universität Berlin

- Information
  - Facts, concepts, ideas
  - A human-oriented term
- **Data** (encapsulated in media)
  - A formalized representation of facts, concepts, ideas
  - Example: text, speech, picture, video
  - A human interpretation of data, conferring meaning to data
- Note:
  - Only data can be communicated,
  - The recipient of data restores information,
  - The recipient interprets data – subject to her interpretation



Telecommunication Networks Group

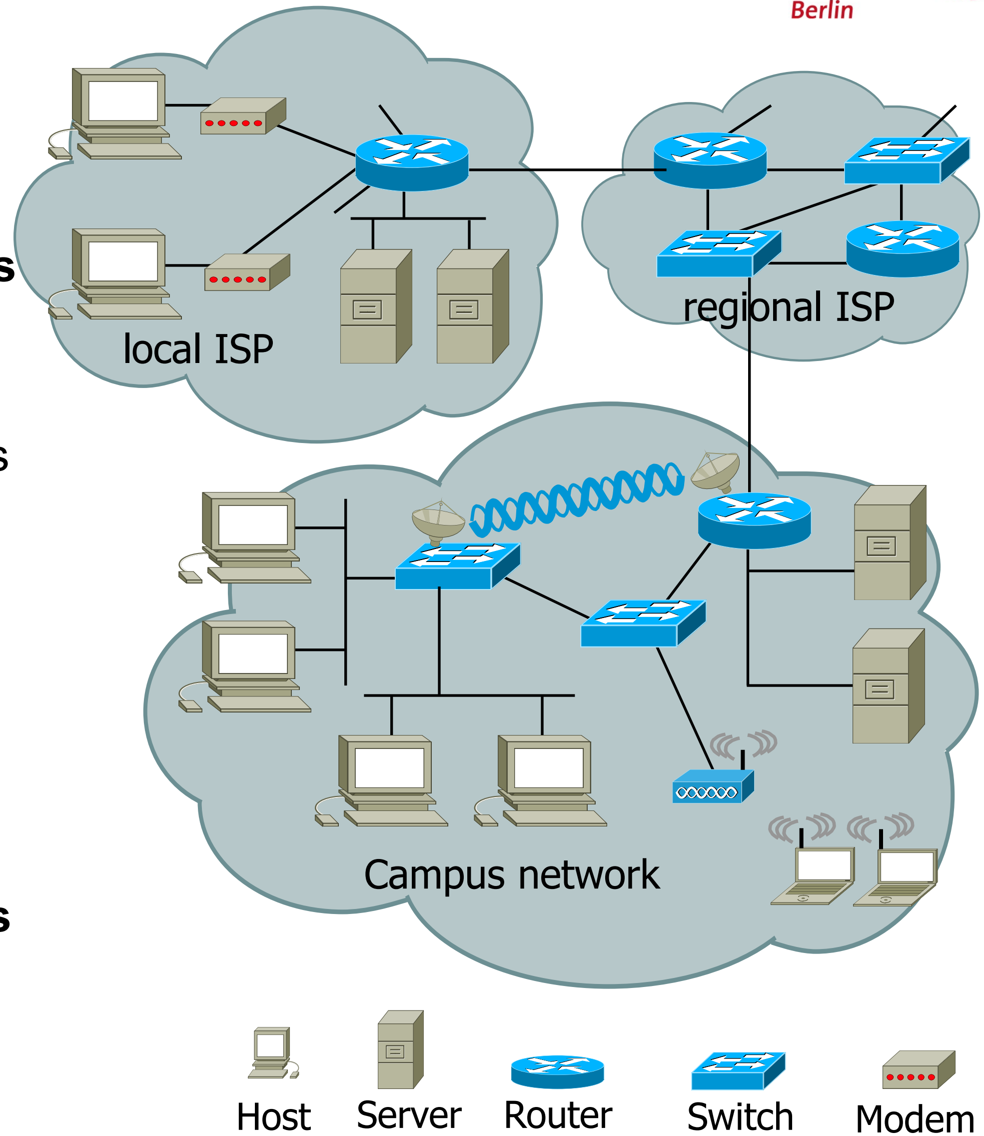
The Data Tsunami

- In 2000 years of recorded history humans created 2 Exabytes of data.
- We generate over 2.5 Exabytes of data/day now!
  - Different sources
- Problem: extracting information out of **data**
  - Where to process them?
  - Bringing data to the processing?
  - Processing data where it emerges and transport (partial) results?

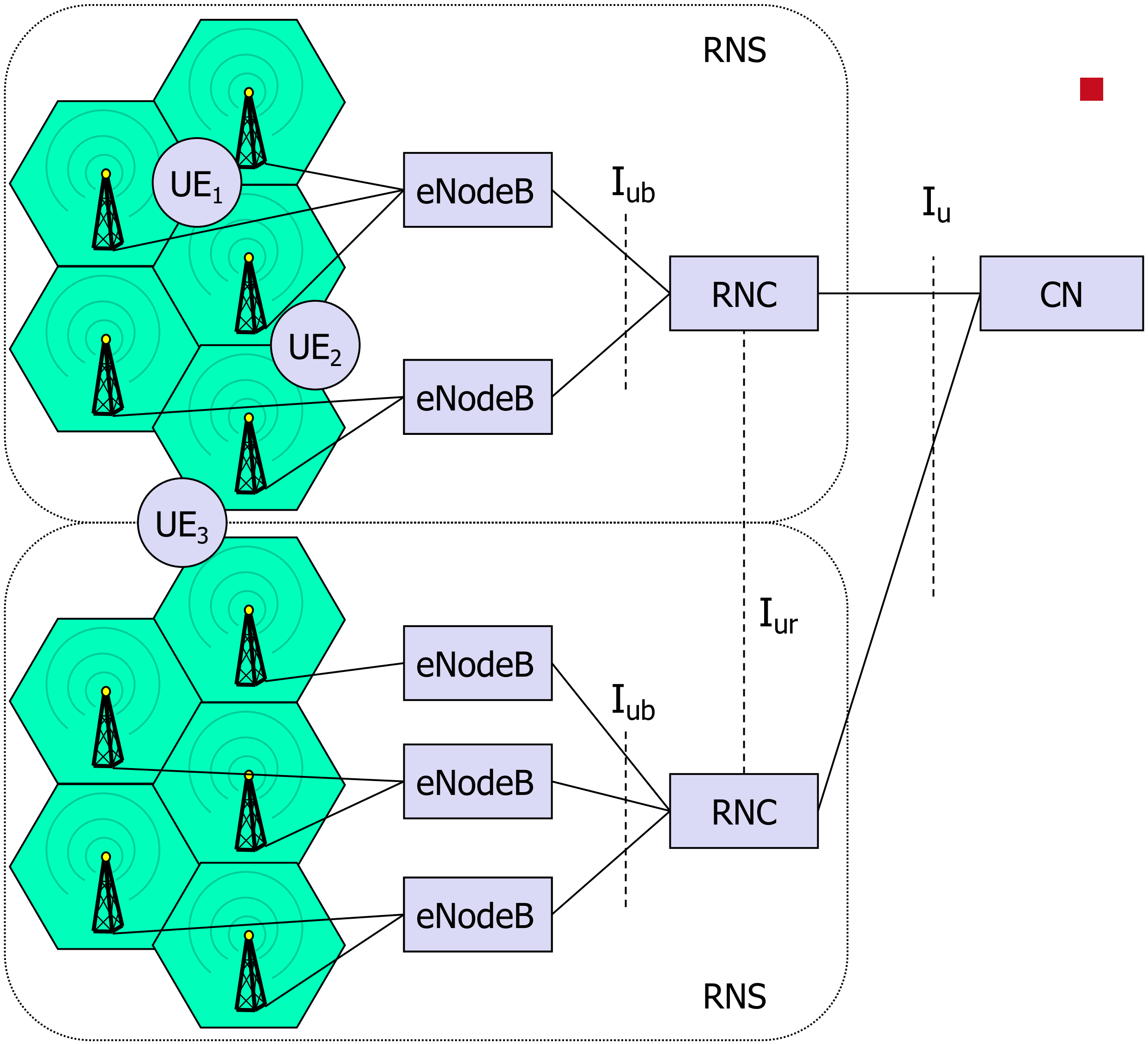
Value		Metric
1000	kB	kilobyte
1000^2	MB	megabyte
1000^3	GB	gigabyte
1000^4	TB	terabyte
1000^5	PB	petabyte
1000^6	EB	exabyte
1000^7	ZB	zettabyte
1000^8	YB	yottabyte

Examples of computer networks

- Internet
  - Communication between **Applications** on **End Systems** (**Host**, **Server**)
  - Use of **Internet Protocols** (e.g., TCP, UDP, IP) and others (e.g., Ethernet, WLAN)
  - Infrastructure consists of **Switches** and **Routers**, also RF base stations and modems
  - Wired and wireless **Connections**
  - Differentiation between **access network** and **core network**
  - Internet Service Provider (ISP)



Examples of computer networks



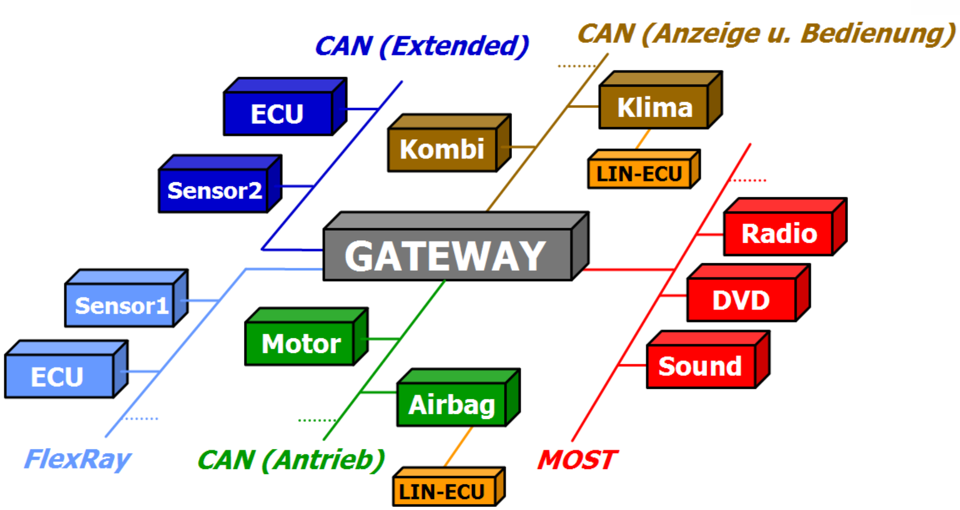
- Cellular networks
  - mobile telecommunications: voice and data
  - Mobile station (User Equipment)
  - Radio Network Subsystem for RF-based cellular networts and base stations (eNodeB / gNodeB)
  - Access network, core network for signaling, transport, connection to other networks

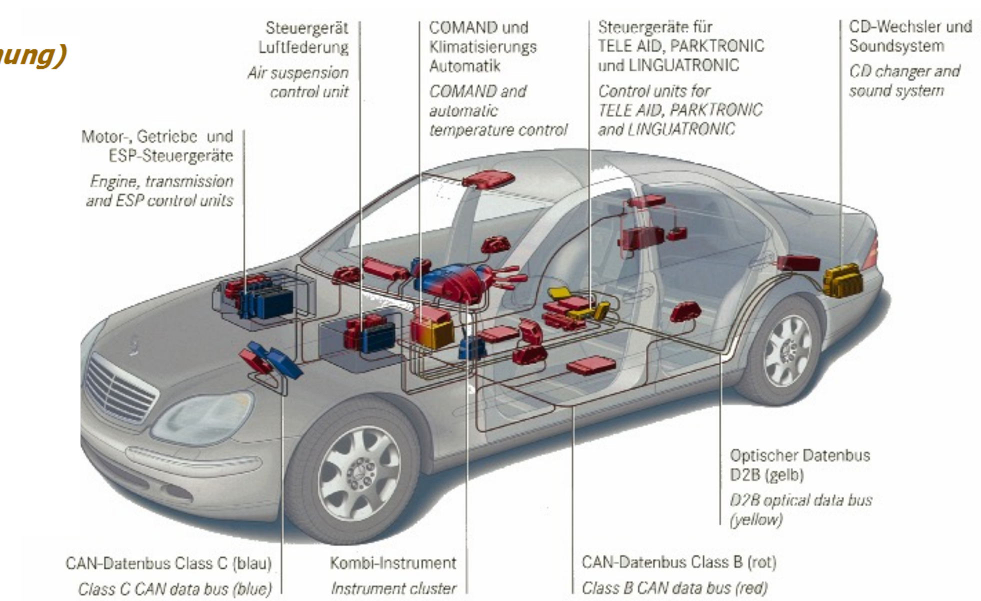
Examples of computer networks

- Ad Hoc Networks, Sensor Networks
  - Ad Hoc Networks: no infrastructure, duality between end systems and routers, self-organizing
  - Wireless Sensor Networks (WSN), modern IoT
    - Small and simple components
      - Small (cm^2, mm^2), cheap
      - Limited energy (battery or energy harvesting)
      - Micro controller-based
      - Wireless communication
      - Sensors such as light, humidity, pressure, acceleration, …
    - Application in logistics, farming, health care, home automation…

Examples of computer networks

- In-car communication networks
  - Mid-size cars have 60-100 electronic control units (ECU) for engine control, driver assistance, comfort, infotainment
  - Bus systems like Controller Area Network (CAN)
  - Strict requirements on robustness and real-time capabilities





Fundamentals

A Distributed System

- A distributed system is a collection of **independent** computers that **appears** to its users as a **single coherent** system.

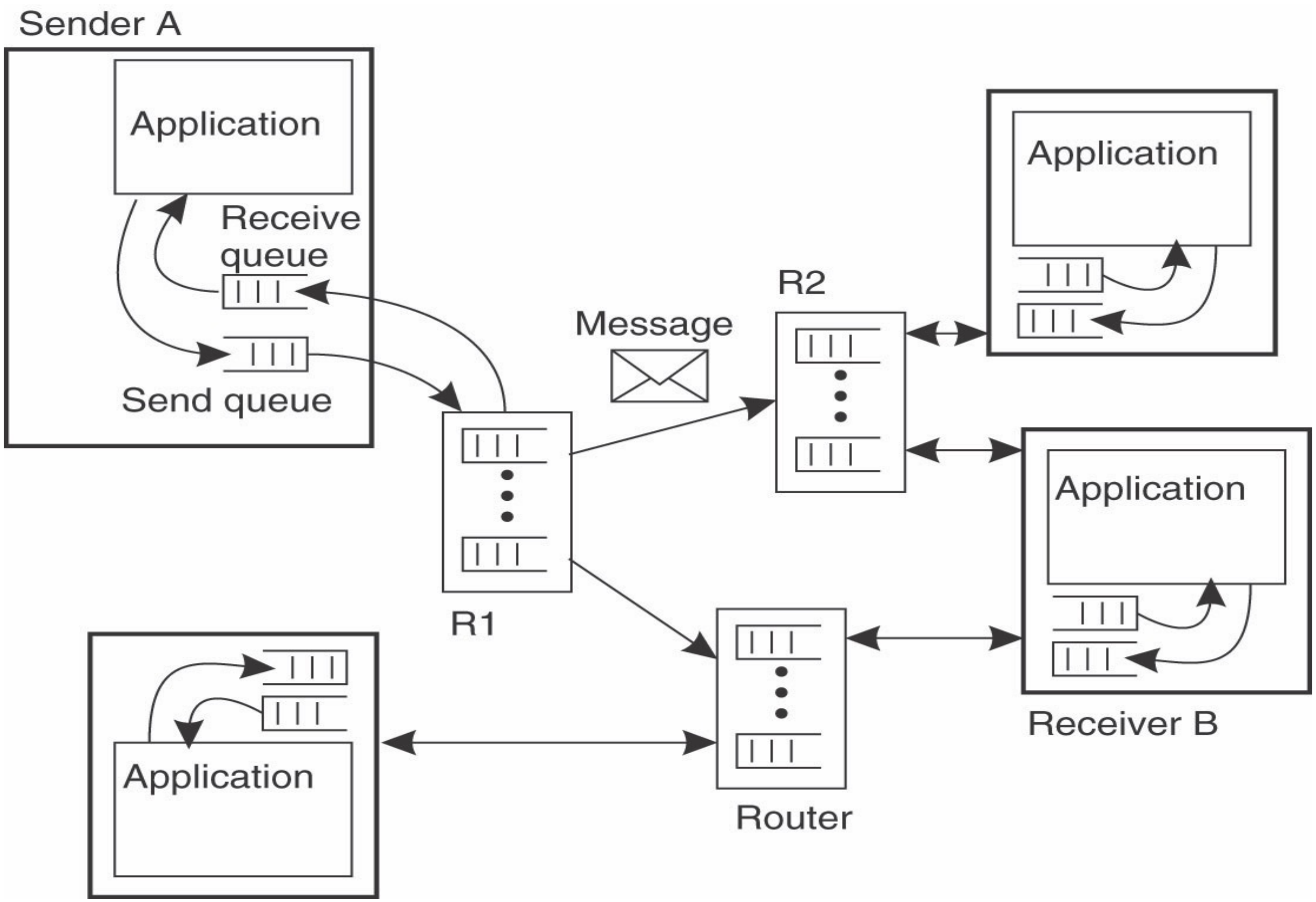
  A. S. Tanenbaum

- A distributed computing system consists of multiple **autonomous** processors that do not share primary memory but **cooperate** by sending messages over a communication network.

  Henri Bal / Colouris

- Comments:
  - Each individual entity is typically a full-fledged system operating on its own
  - Individual entities might follow local policies, be subject to local constraints
  - Each of them might fail (completely or operate wrongly) at any time

The Message Passing System



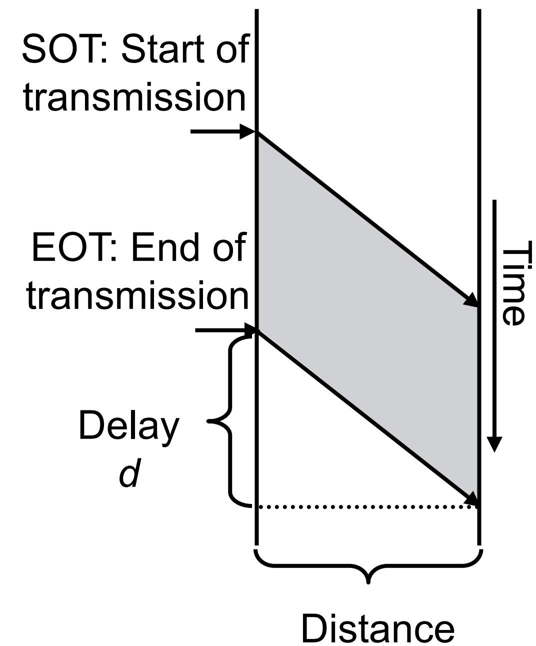
[Tannenbaum, op. cit.]

Properties of Message Passing

- Non-Zero delay between sending and receiving,
- A bounded amount of messages can be posted in a time unit,
- Loss/errors in data transmission,
- In addition also: message duplication, variable delay (jitter), and message re-ordering.

Realistic Transmission

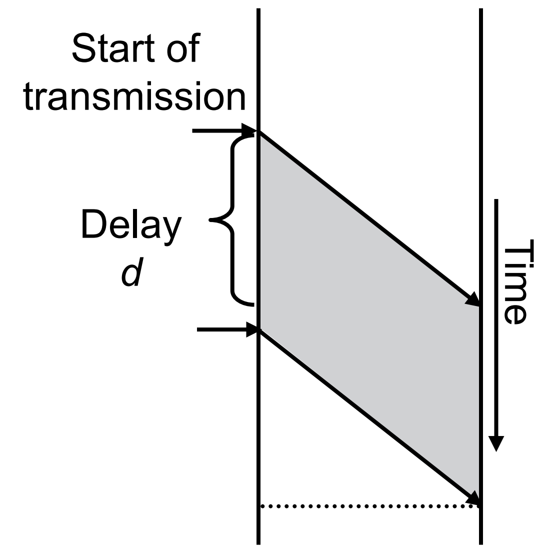
- **Propagation delay *d***:
  - Propagation speed v:
    - Speed of light: $v=c$,
    - In copper/fiber: $v \sim 2/3\ c$
  - d = distance / v
- **Data rate r**: How much bits/second can a sender transmit?
  - (EOT – SOT) = data size [bits] / data rate [bits/s]
- **Error rate**: What is the rate of incorrect bits arriving at the receiver?
  - Messages containing incorrect bits might be discarded



Telecommunication Networks Group

Transmission Medium can store Data

- What happens during a transmission?
  - Bits propagate to the receiver,
  - Sender keeps sending bits,
  - First bit arrives after *d* seconds,
  - In this time, sender has transmitted $d \times r$ bits
  - They are "stored in the wire" (or in the air!)
- $d \times r$ is the product of delay and data rate
  - Commonly called **bandwidth-delay product**
  - Crucial network property



Example (transcontinental cable):

- Data rate 100 Mbit/s
- Delay 4000 km / (2/3c) = 0.02 s
- $d \times r = 2$ Mbit (in the wire)

Keeping the Pipe full

- $v = 2/3\,c$

Data rate	Propagation distance of 1 bit	Bandwidth-delay product over 5000 km link
300 bps	666 km	7 bits
3.3 kps	60 km	82 bits
56 kps	3.6 km	1.4 kbits
1.5 Mbps	133 m	37 kbits
150 Mbps	1.3 m	3.7 Mbits
1 Gbps	20 cm	25 Mbits
10 Gbps	2 cm	250 Mbits
100 Gbps	2 mm	2.5 Gbits

Why important? Efficiency!

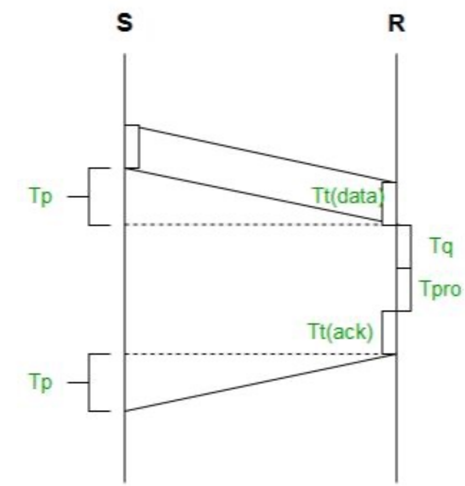
- Ex.: **Stop and Wait protocol**
  - … a very simple flow control protocol
  - Sender sends one packet and waits for the receiver to acknowledge
  - In case the acknowledgment is not received, the sender will retransmit the packet
- Total time taken to send one packet:

  = Tt(data) + Tp(data) + Tq + Tpro + Tt(ack) + Tp(ack)

  - Since, Tp(ack) = Tp(data) and Tt(ack) << Tt(data), we can neglect Tt(ack) (set Tq = 0 and Tpro = 0)
  - Total time = Tt(data) + 2 * Tp
  - **Efficiency (η)** = Useful time / Total cycle time

    = Tt / (Tt + 2*Tp)

    = 1 / (1+2*(Tp/Tt))



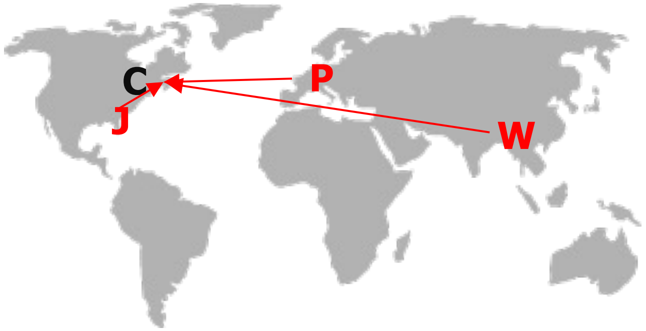
Tt(data) : Transmission delay for data p.
Tp(data) : Propagation delay for data p.
Tq: Queuing delay
Tpro: Processing delay
Tt(ack): Transmission delay for ACK p.
Tp(ack) : Propagation delay for ACK p.

Implication of the Real Message Passing

- No participant has complete information about the system state.
  - I do not know what you know – NEVER
- Participants make decisions based only on local information,
- There is no implicit assumption that a precise common understanding of time exists,
- Can be elaborated … with some accuracy…

Example: Distributed Stock Trading

- Brokers distributed all over the (connected) world:
  - Each of the brokers sets his computer to:
    - Buy stocks of company *HOPE* if they fall under X $
- The central unit (located at Wall Street, NY) defines the value of the *HOPE* stocks and announces them
- Due to different transmission delays broker John at Wall Street will always **react quicker** than Paul in Frankfurt or Wendong in Peking!
  - Will he really?
- Will brokers located close to each other receive the information simultaneously?
  - The notion of fairness



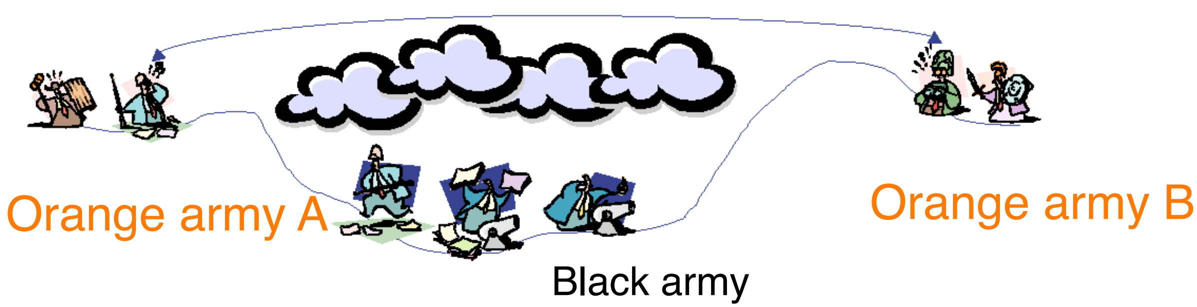
A Dead Man that Shoots

- Consider an action game: player **A** shoots player **B**:
  - Application **A** creates a bullet entity with a certain heading and velocity. It will then transmit the state of that bullet entity.
  - Upon receiving the state of the bullet application **B** will start to check whether any entity under its control is hit by the bullet.
- There is, however, a transmission delay
  - This network delay may be large (~>100ms) although the players in “virtual space” are close
  - During this time player **B** might take actions,
    - shoot at another player **C**, even though “he’s already dead”.
    - move so that the bullet would not hit him - from his point of view!
- Q.: Which hits should be scored?

[Mouve, Mannheim, 2005, op. cit.]

The Two Army Problem

- Principles:
  - Orange army: A and B have 2N soldiers each. The black army has 3N soldiers.
  - In case of conflict the bigger force wins.
  - The two orange forces have to communicate to synchronize their attack. Can they?
- For this communication they need:
  - A common language (possibly not understood by the black army)
  - A communication path - a messenger - who has to pass through the land occupied by the enemy (and could be intercepted with a given probability *p*). A successful trip takes time *D*.

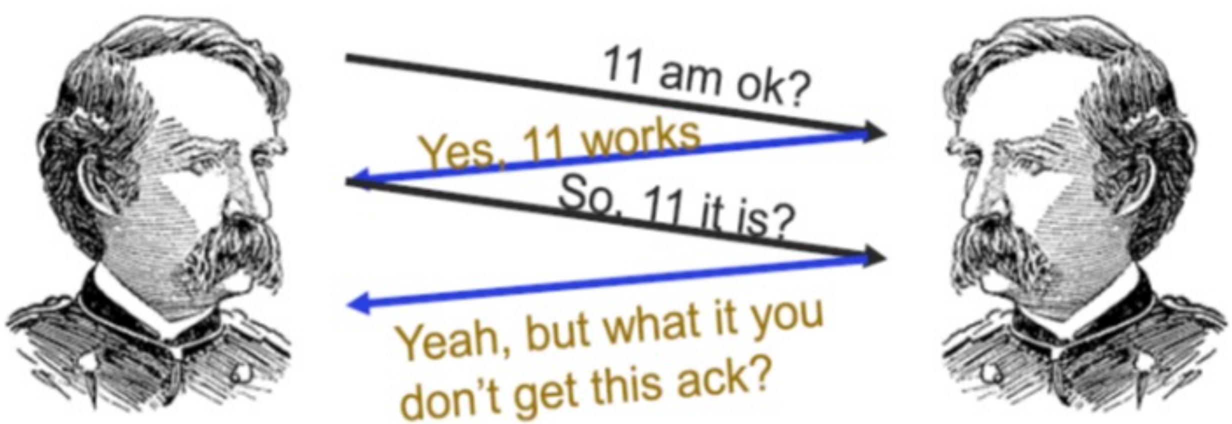


Some Difficulties

- Assume that commander A and commander B send a messenger to the counterpart with different suggested time for an attack. Afterwards both agree for the suggestion of the partner.
  The play goes on ....
- Let the Commander A (senior) send a messenger with an order.
  - Did the Commander B receive the message? When? How does Commander A know?
  - Let request the commander B to acknowledge his readiness to follow the orders. Is the victory sure? How does B know that A got the acknowledgment?
- The attack of the orange armies will always fail with probability *p*!

The Phenomenon illustrated

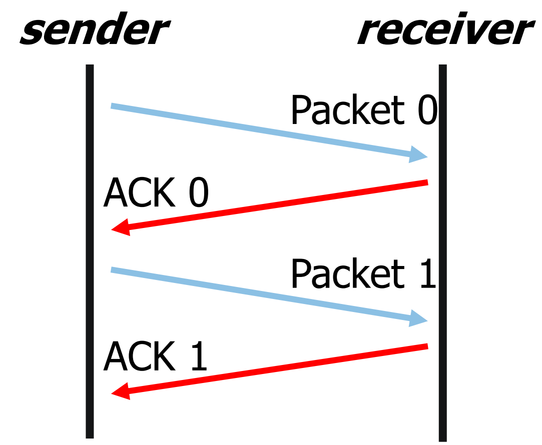
- Can messages over an unreliable network be used to guarantee two entities do something simultaneously?
  - No, even if all messages get through
- No way to be sure that last message gets through!

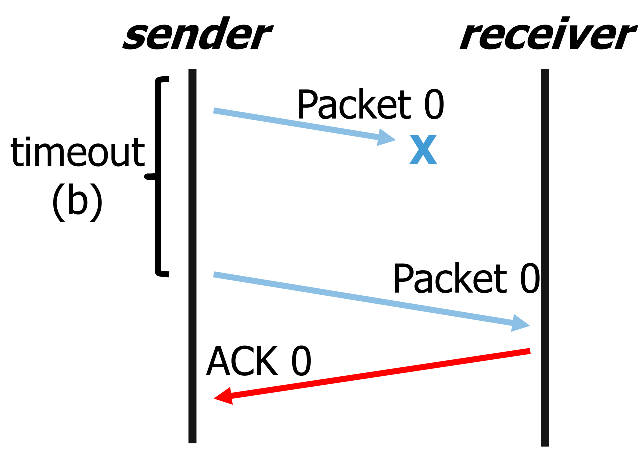


Can we find a “better” strategy?

- Sender:
  - Sends the information.
  - If no acknowledgment has been received within time “b“ → send again.
- Receiver:
  - Waits for an information.
  - Acknowledge the received information.

- This is called a “send/stop and wait” protocol
- **Note**: One possible mechanism to avoid losses





Let us do some Math

- Successful transmission is acknowledged after a constant time b.
- Assume that with probability Π there is no acknowledgement.
- The time needed for a successful transmission is
  - b with probability $(1-\Pi)$
  - 2b with probability $(1-\Pi)\,\Pi$
  - 3b with probability $(1-\Pi)\,\Pi^2$
  - (k+1)b with probability $(1-\Pi)\,\Pi^k$
- Assume that there is only time for N „rounds“
- The probability of successful attack is $(1-\Pi^N)$
- In a finite time it is not possible to achieve a delivery with probability equal to 1.0

Another Example - A Bank Transfer

- Consider a bank with headquarters/accounting in METROPOLY which has a branch in VILLAGE
- Client: Has 5000 € on his account. He enters the branch in VILLAGE for to cash 1000€
- Send-and-Wait will be used to transmit this request:
  - Clerk 1 (branch): Posts and repeats upon timer expiration!

“If Balance > 1000 €

then (subtract 1000 € and acknowledge),

otherwise call the police in VILLAGE.

  - Clerk 2 (headquarters): Executes the request.
- Imagine the acknowledgement is lost, several times in sequence!

Client – Server Model (distributed computing)

- The clerk 2 is „only" executing the requests posted by the clerk 1,
- Conceptually, clerk 1 might have done it himself,
  - But did not have the data „handy",
  - And, possibly, could serve next customer while clerk 2 has been processing his request - **pipelining**
- Such schema of operation is called „client-server" model
  - **Service**: Any act or performance that one party can offer to another that is essentially intangible and does not result in the ownership of anything. Its production may or may not be tied to a physical product.

    D. Jobber, Principles and Practice of Marketing
  - Focus is on the **output**, the result of the service not the means to achieve it

    D. Jobber, Principles and Practice of Marketing
- We will discuss this in detail later …

But how to solve our problem?

- Remedy?
  - One way to go: Change the way of interaction between the clerks!
  - Another way to go: Replace the Send-and-wait by some better solution (without duplications!)

1st Option - Restructuring the Interaction

- Check the balance
  - Clerk 1: Posts a following message and sets a timer!

  Post me the balance, account no. X // comment: can be repeated several times
  - Clerk 2: Executes the request
- Set New Balance
  - Clerk 1: Computes the new balance;
  - Clerk 1: Posts and repeats upon timer expiration!

  New balance for account no. X is 4000 $, confirm
  - Clerk 2 : Executes the request
- **Idempotent actions**: activities which result does not change in case of repetitive execution.

2nd Option - Restructuring the Interaction

- Add to each request of clerk 1 a transaction identifier (TxID)
  - Use the same TxID for an original request and for any possible repetition of this request
  - Do not use the same TxID for different transactions!
- Good news. It seems to work.
- What happens if messages might be heavily delayed
  - Clerk 2 should memorize the list of all the “already used” TxIDs for each account number
- He has something to memorize ….
- Construction of such protocol is not trivial. We will discuss this later in this class!

The Concept of **State**: intuitively

- Information relevant to the progress of some activity. Access to this information is frequently necessary to assure proper continuation of this activity.
- Send-and-wait
  - Sender has to be aware of the timer!
  - Receiver is **stateless**
- In the first solution – being stateless is a desired feature of a server!
  - Consider changing the server? Server collapse?
  - Consider **scalability**: if there are many clients and many accounts - state has been memorized for each of them?

Hard State vs. Soft State

- Introducing state might be necessary. Typically if some resources have to be kept available …
  - Consider the restructured banking example. If several clients would use the same account a „lock“ on access would be needed between Checking Balance and setting New Balance.
- Hard state approach:
  - State information has to be deleted as result of a proper action!
  - What happens with the “lock” if the computer of clerk 1 breakes down after “checking the balance”?
- Soft State Approach:
  - The state information removed if not reactivated since X.
  - Analogy: like putting shoes in the store on hold for 2 days …

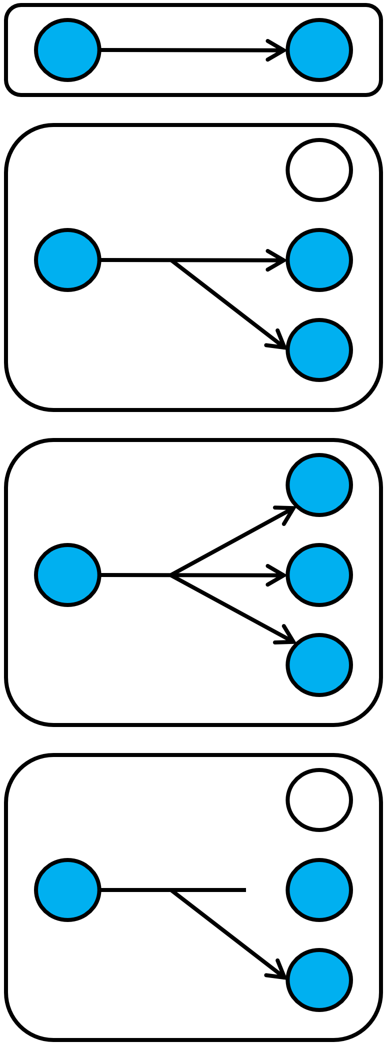
Service: is a very general notion …

- In the bank example:
  - Clerk 2 has been offering some service to clerk 1.
  - The banking offers some set of services to the customers!
- But also the communication can be considered as a service:
  - Imagine that clerk 1 and clerk 2 ask their assistants to handle the communication (using Send-and-Wait by e.g. phone?)

Classification of communication networks

Classification of communication networks

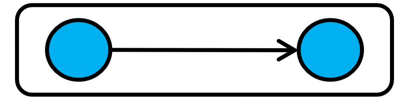
- Communication partners
  - **Unicast** (point-to-point): one sender, one receiver
  - **Multicast** (point-to-multipoint): one sender, a group of receivers
  - **Broadcast** (to everybody): message to all nodes in the network
  - **Anycast** (to somebody): message to one receiver out of a group of potential targets



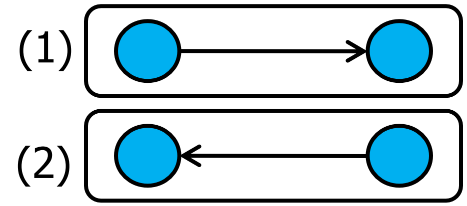
Classification of communication networks

Technische Universität Berlin

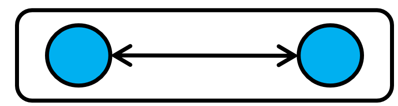
- Direction of communication
  - **simplex**: unidirectional connection



  - **half-duplex**: bidirectional connection but only simplex per time



  - **(full-)duplex**: bidirectional at the same time



Classification of communication networks

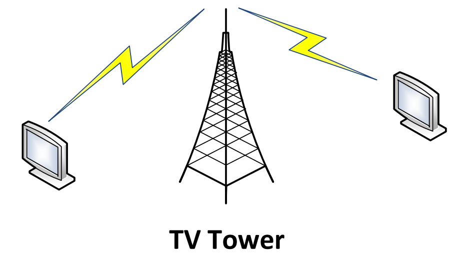
- Communication medium
  - **wired**
    - e.g., copper cables or fiber optics
    - Bit rates of kbps to Gbps
    - Signal propagation speed close to speed of light (depending on medium)
      $c \approx 2 \cdot 10^8$ m/s = 200 m/µs
    - Low bit error rates, e.g., 10^{-10} on fiber optics
  - **wireless**
    - e.g., RF (radio frequency) or infrared
    - Bit error rates rather high (many signal propagation issues): 10^{-5} to 10^{-2}
    - Bit errors often appear in bursts

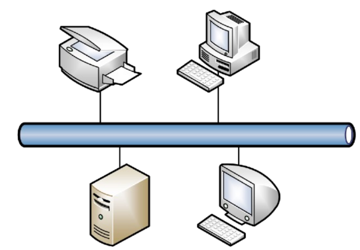
Classification of communication networks

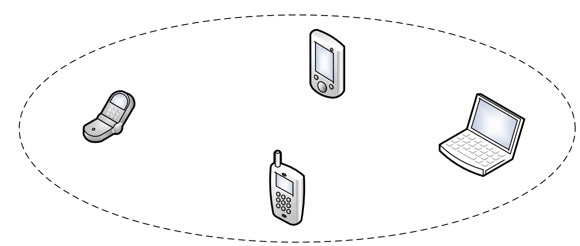
- Switching mode
  - **Circuit Switching**
    - Connection is established between sender and receiver using a **signaling protocol** (e.g., time or frequency multiplexing)
    - Available bit rate needs to be shared among all connections
    - Standard for classic telephony, inefficient for bursty data communication
  - **Packet Switching**
    - Data is being sent in form of packets from sender to receiver
    - Sharing of bit rate more efficient
    - Temporary busts can be dealt with using buffers
    - This may lead to artificial delays and even buffer overflow

Broadcast Medium & Multiple Access

- Common characteristic of a broadcast medium:
  - Only a single sender at a time,
  - Exclusive access is necessary,
  - Simple to achieve with a multiplexer
- What if no multiplexer is available?
  - E.g., a bus: all nodes connected to a single wireline
  - Or a group wireless devices? Compare: group of kids …
- Exclusive access has to be ensured
  - Rules have to be agreed upon





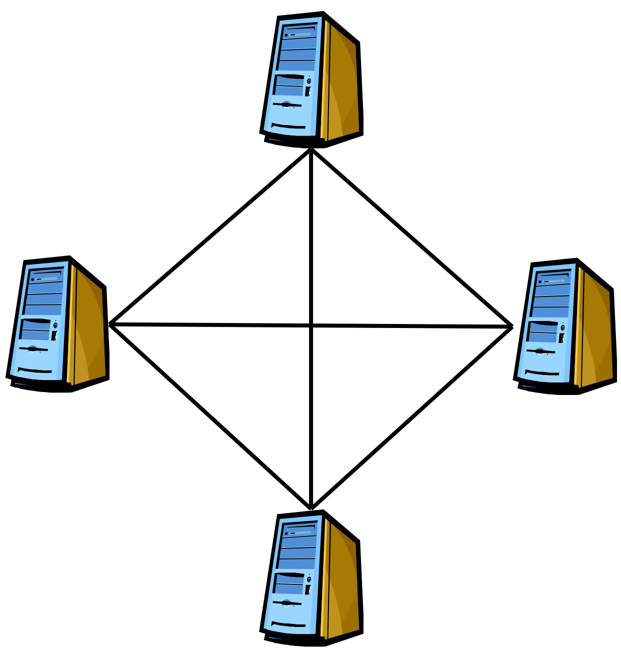


Network topologies

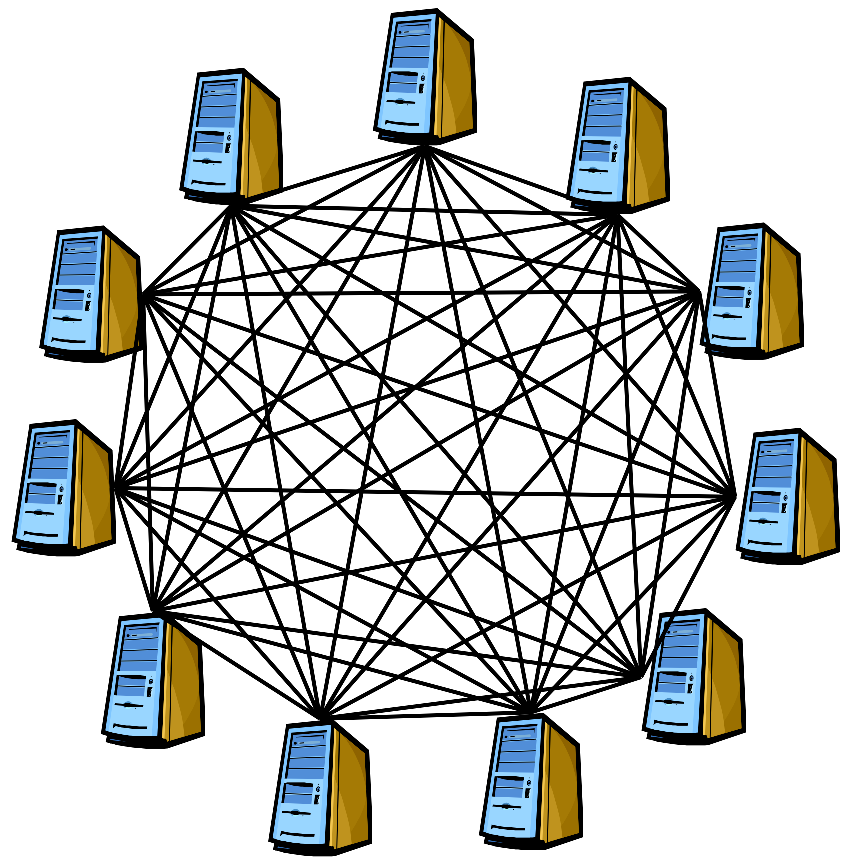
Classification of communication networks

- Topology

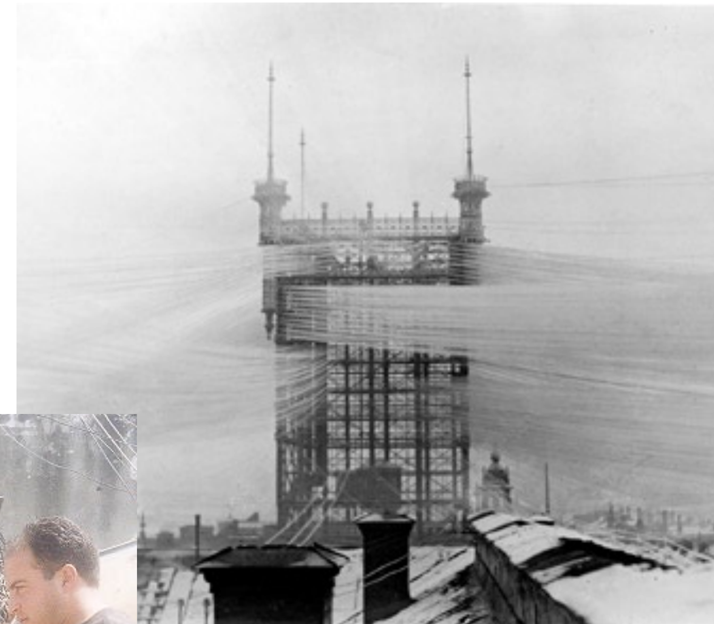
Four nodes



Eleven nodes



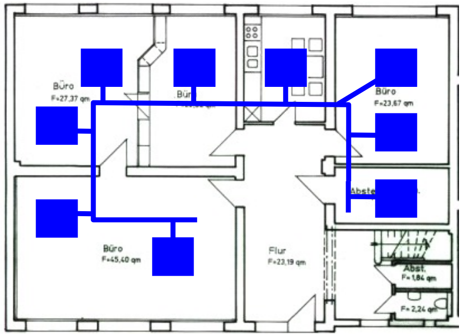
Topology in reality

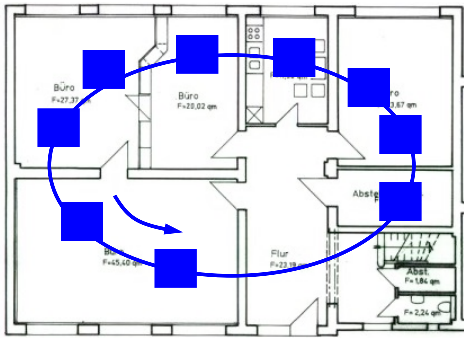


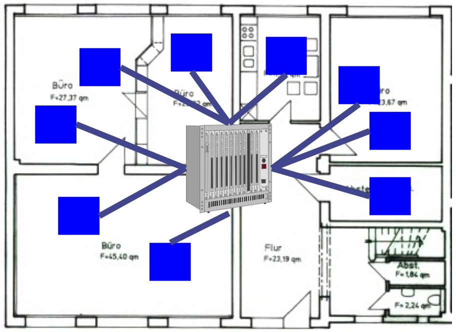
Telecommunication Networks Group

Relevant topologies

- Structured networks
  - Bus, Ring, Star

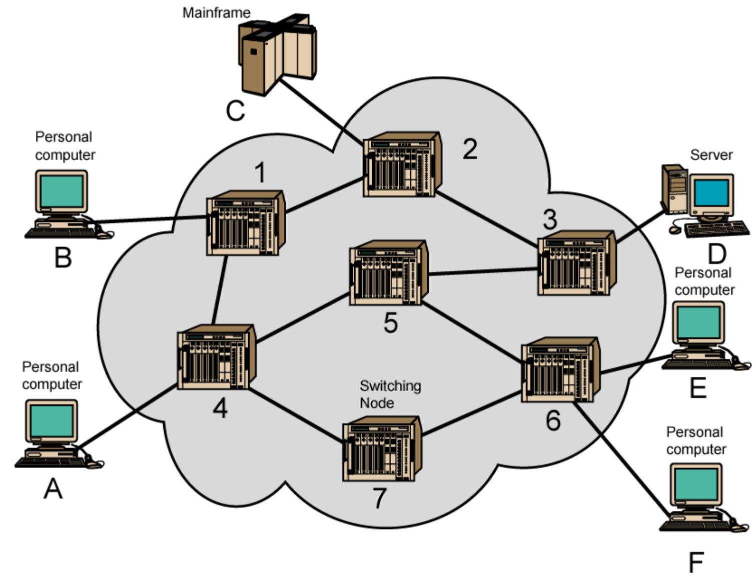






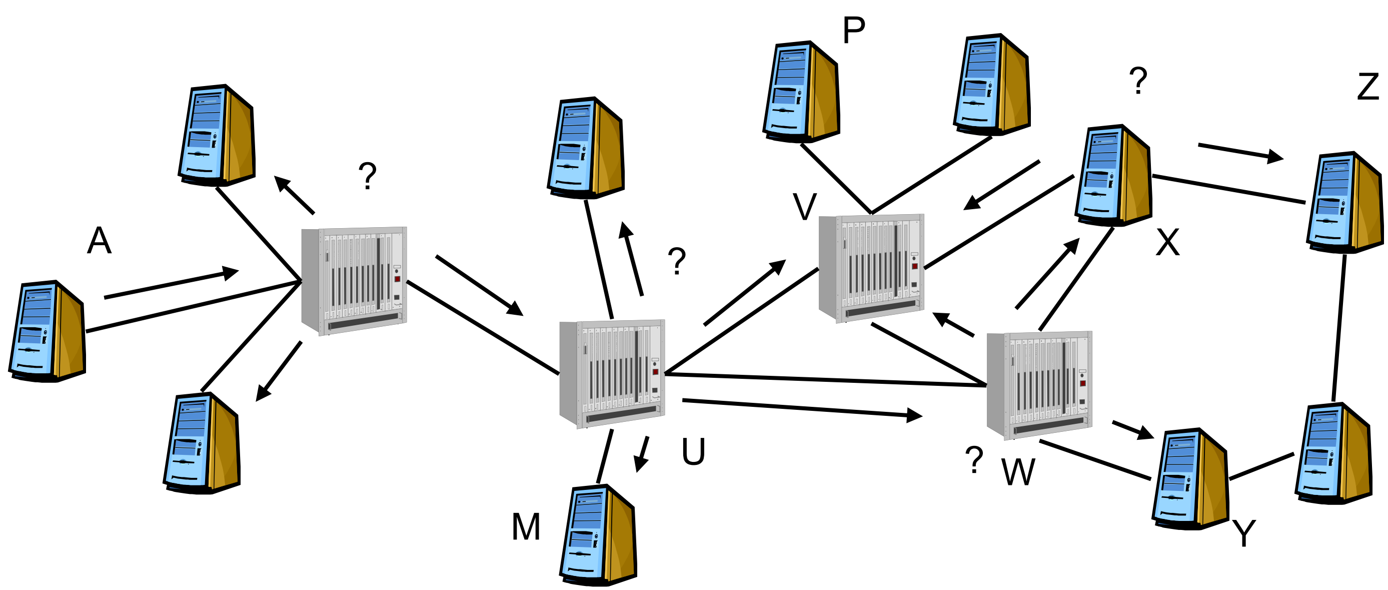
Reminder - Switched Network

- A set of path sections (e.g., electrical cables) and switches,
- “end systems” (terminals/user devices) vs. “switching elements” (routers/bridges)



Forwarding and Next Hop Selection

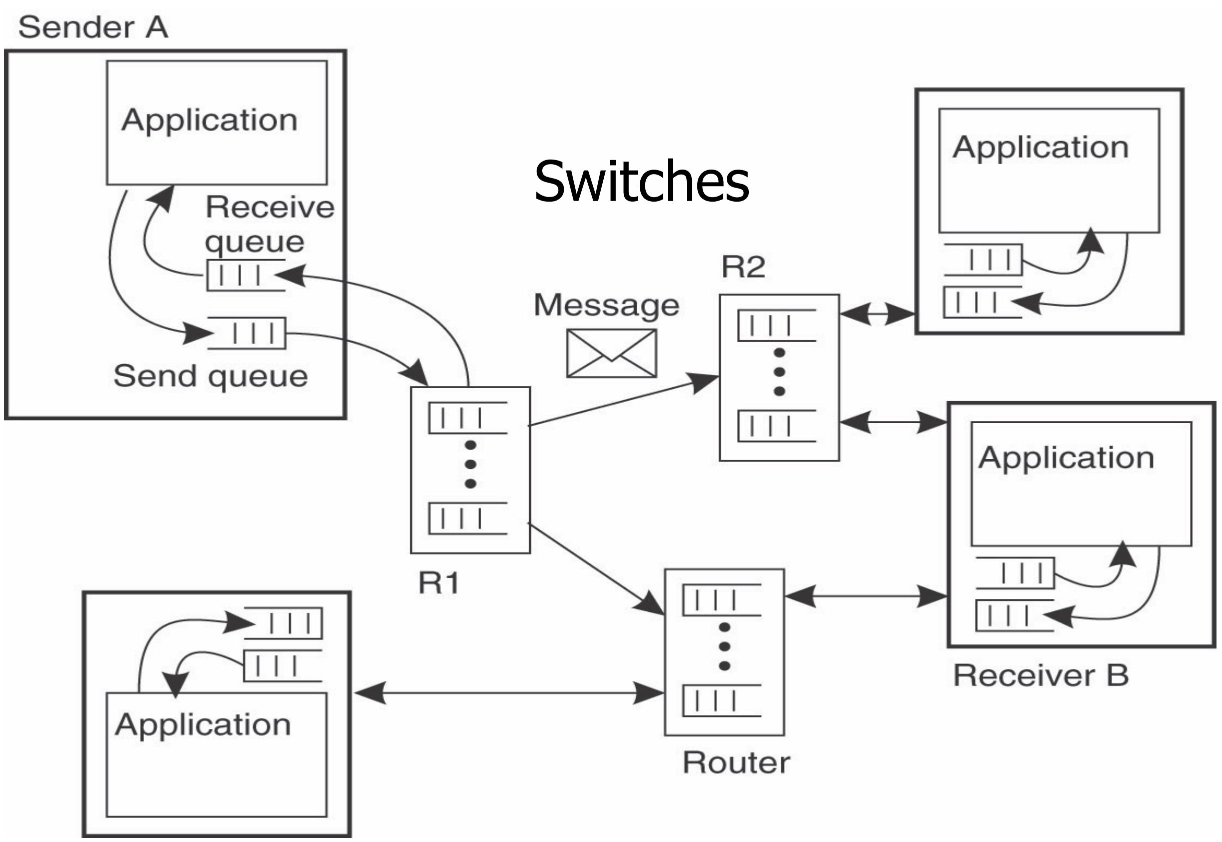
- Switch forwards a packet onto the next “piece”
- Recall: A switching element → a hop towards its destination
- How does a switch know which of its neighbors is the best one towards a destination?
- What is a “good” neighbor, anyway?



Addressing, Routing, Forwarding

- **Name**: whom would you like to reach? (object identity)
- **Address**: where is the object? (locator)
- **Routing**: each switch has to know which of his outputs should be used for a given destination address
  - Hopefully contributes to short "overall trip distance, time"
  - Some understanding of the possible routes is necessary to decide
- **Forwarding**: a packet has arrived. How to "get rid of it" in the way consistent with the routing?
  - With possibly short delay and - hopefully - little delay variation,
  - Structuring of the information describing packet destination and the way routing information is stored matters for execution time

General Architecture of a Message System



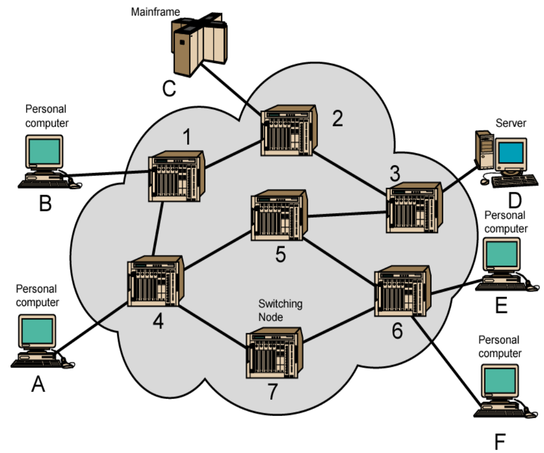
What is the value of a Network?

- Communications networks increase in value as they add members - but by how much?
  - How useful is a single phone using a unique new technology? Two phones? 20 phones? 1 billion of phones …
  - Btw. as by 2017 they are around 5 billion mobile communication users out of worlds population of over 7.8 billions of people
- The Metcalfe´s Law „The value of a communication network is proportional to the square of the user number"
- Other: **n log(n) law** [1]

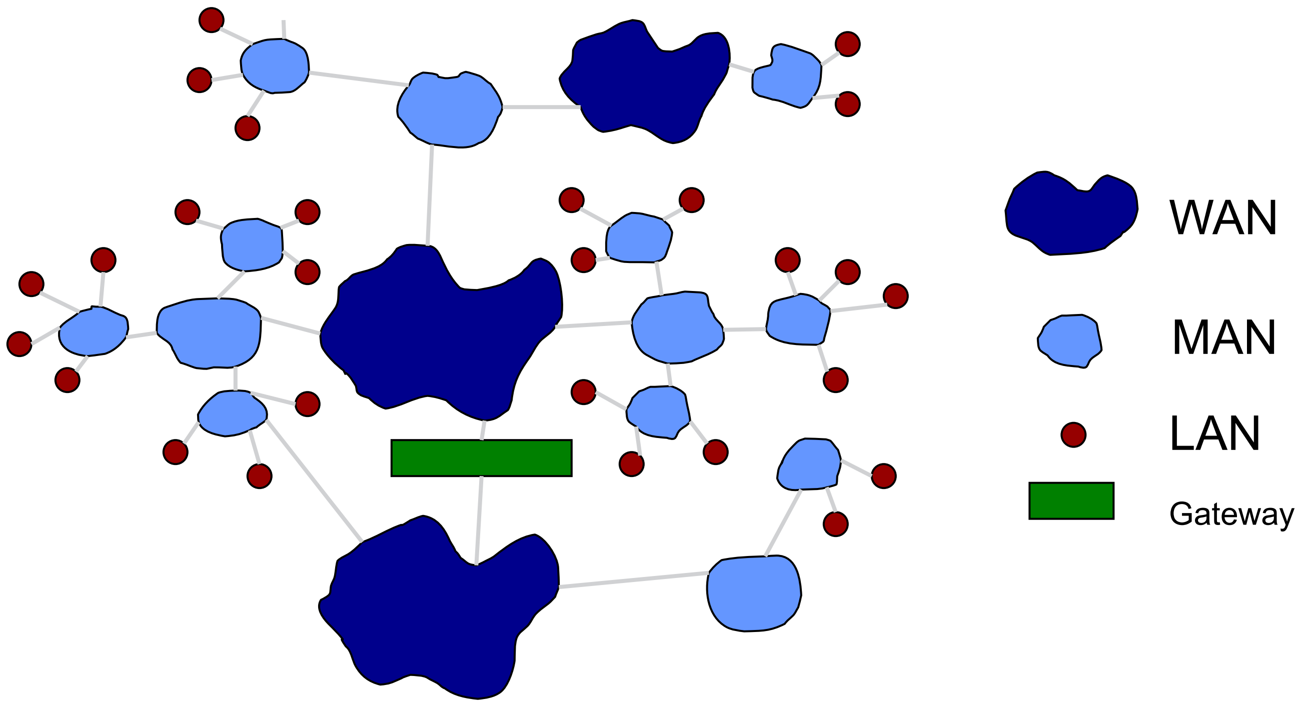
[1] Briscoe, B., Odlyzko, A., & Tilly, B. (2006). Metcalfe's law is wrong-communications networks increase in value as they add members-but by how much?. IEEE Spectrum, 43(7), 34-39.

Large Networks need Structure! Why?

- Scaling
  - Remember: each switch knows route to each destination …
  - Hierarchy usually simplifies a lot …
- Locality
  - Close hosts are clustered,
  - Local networks
- Heterogeneity
  - Different applications (e.g., control, sensing) have different requirements,
  - Multiple technologies for access (e.g., wired, wireless)
- Administration
  - Who sets the rules for usage?



Internet: Interoperability vs. Heterogeneity



WAN = Wide Area Network, MAN = Metropolitan Area Network, LAN = Local Area Network

Standardization

Standardization

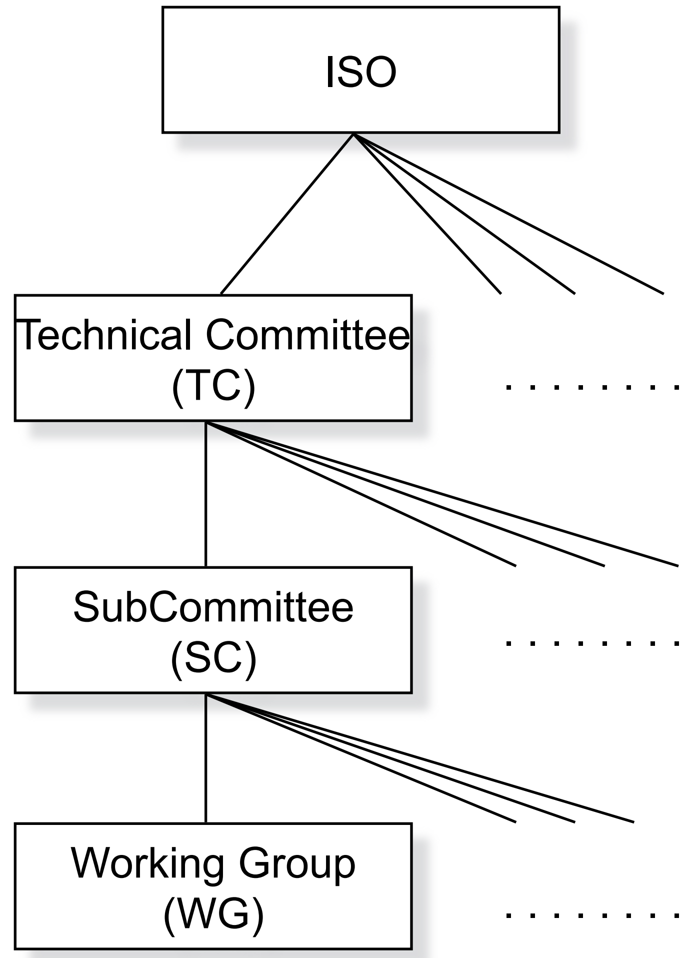
- Essential to realize large, world-wide networks
- Traditionally coordinated by organizations with telecommunications background
  - Established, world-wide coverage, often slow “time to market”
- Internet
  - Internet Engineering Task Force (IETF)
  - Consensus-oriented, focus on “working” implementations
  - Initially fast “time to market”, today much slower
- Many companies try to push their interests
  - “de facto” standards

Standardization: more traditional organizations

- ITU – International Telecommunication Union (formerly CCITT und CCIR)
- CCITT – Consultative Committee on International Telegraphy and Telephony (Comité Consultatif International Télégraphique et Téléphonique)
- CCIR – Consultative Committee on International Radio
- CEPT – Conférence Européenne des Administrations des Postes et des Télécommunications
- ISO – International Organization for Standardization
- DIN – Deutsches Institut für Normung
  - German partner organization of ISO

ISO-Standardization

- WG meetings:
  - Every 6-9 month
  - National organizations get time to accept proposed concepts
  - Afterwards: standardization
    - DP: Draft Proposal
    - DIS: Draft International Standard
    - IS: International Standard
- Standard is more a recommendation for others by means of international consensus
- Very slow process



IETF

- IETF is organized in Areas and Working Groups
  - Members from industry, academia, governments
- Drafts/Proposal can be submitted by anybody
  - „on-demand“
- Standardization requires at least two independent implementations
- Informal voting in working groups
  - „Humming“
  - Three meetings per year
- Result:
  - RFC – request for comment (the standard)
  - FYI – informal or informational