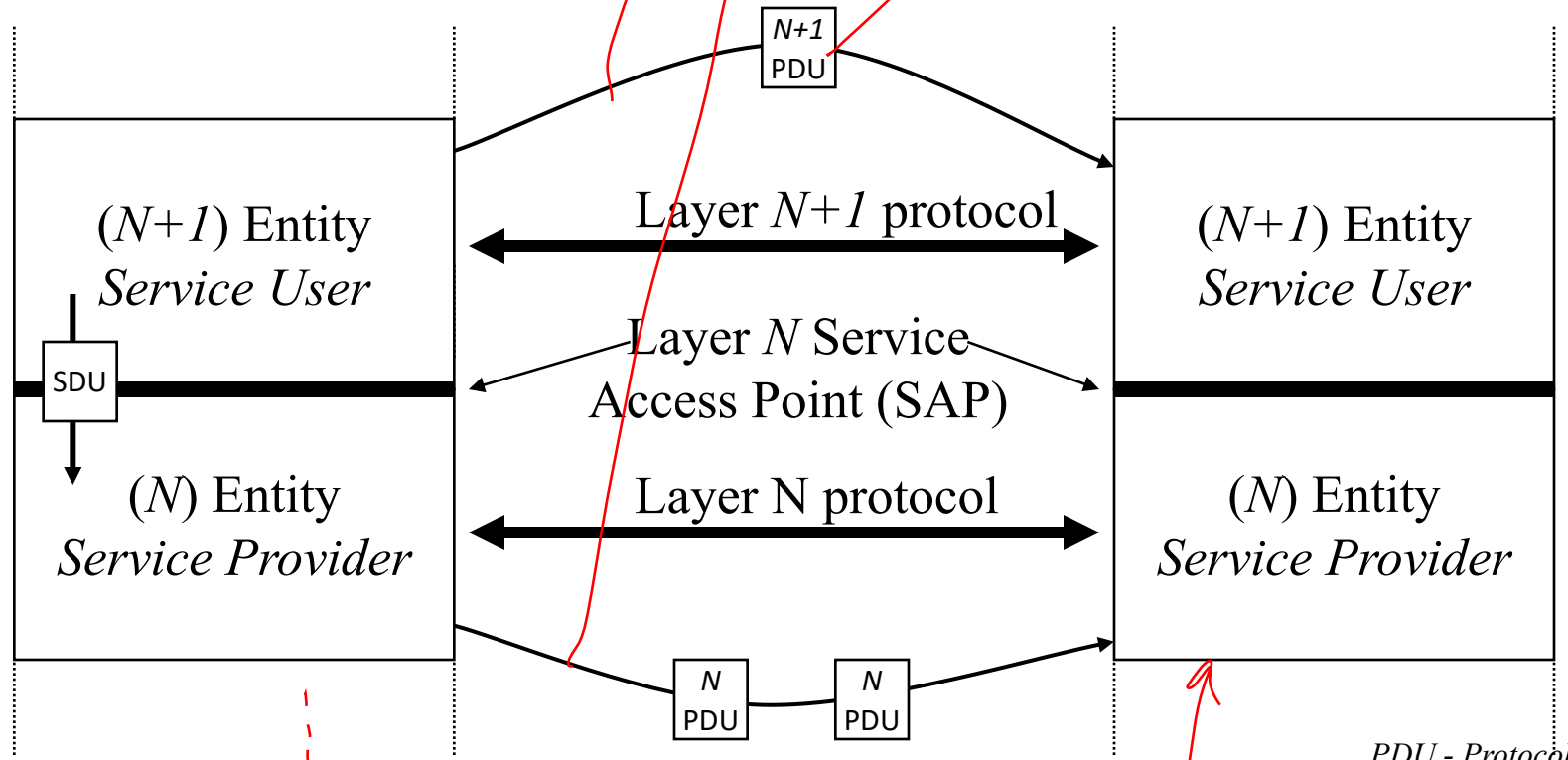
LEC 3

Services in the OSI Model

- In OSI model, each layer provide services to layer above, and ‘consumes’ services provided by layer below
- Active elements in a layer called *entities*
- Entities in same layer in different machines called *peer entities*

Layering Principles



PDU - Protocol Data Unit
SDU - Service Data Unit

- Layer *N* provides service to layer *N+1*

Connections

- Layers can offer *connection-oriented* or *connectionless* services
- Connection-oriented, like telephone system
- Connectionless, like postal system
- Each service has an associated *Quality-of-service* (e.g., reliable or unreliable)

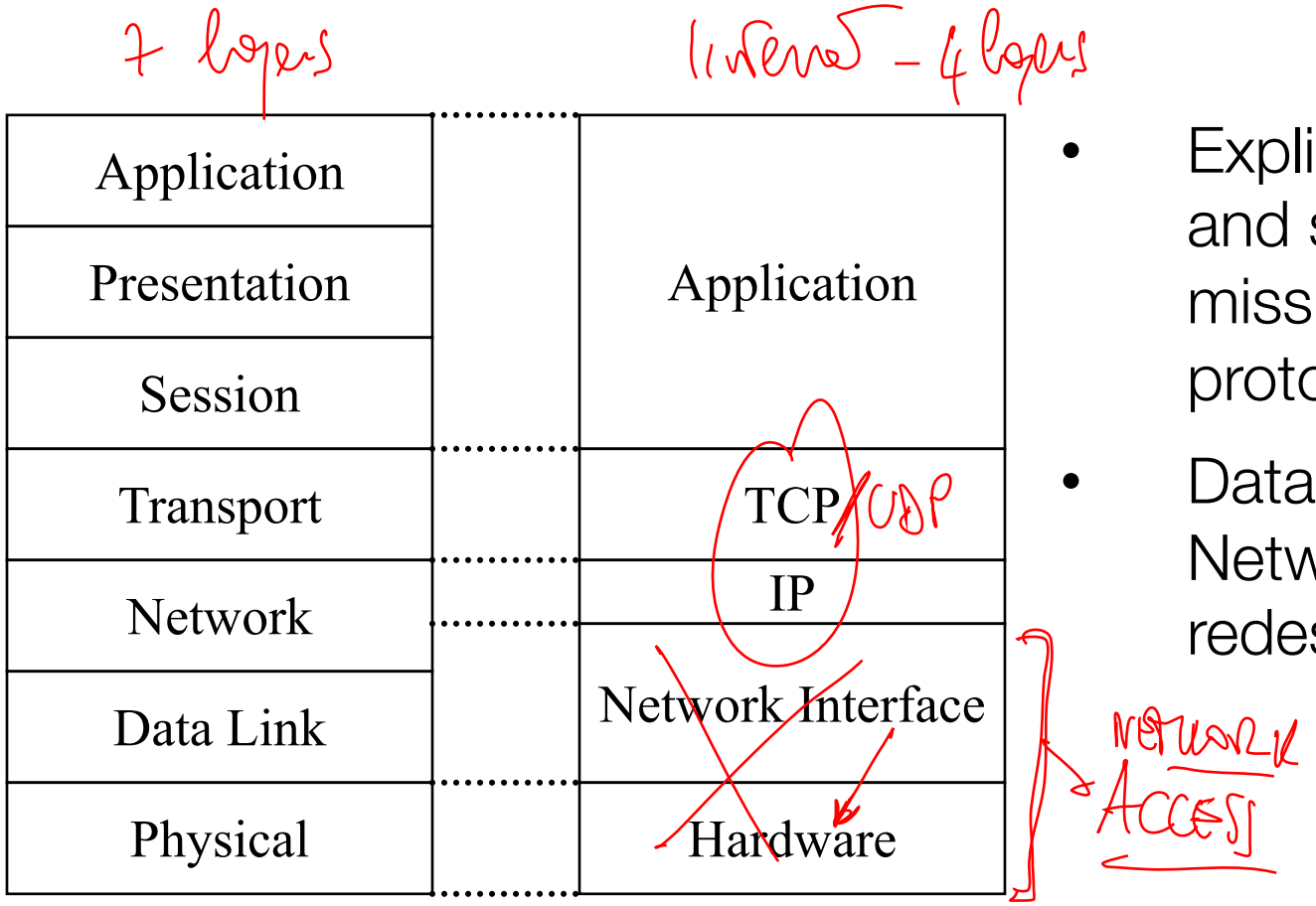
Reliability

- Reliable services never lose/corrupt data
- Reliable service costs more
- Typical application for reliable service is file transfer
- Typical application not needing reliable service is voice traffic
- Not all applications need connections

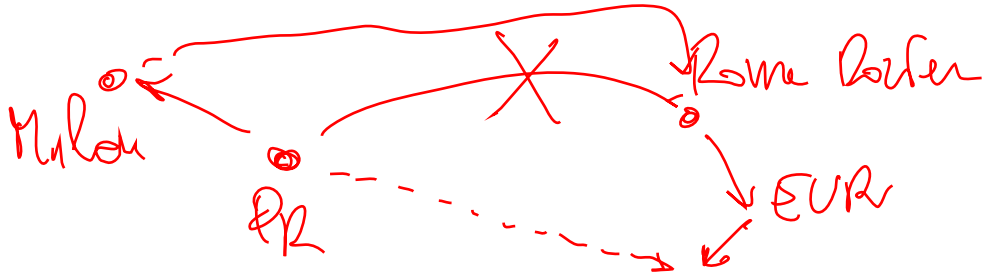
Topics

- *Service* = set of primitives provided by one layer to layer above
- Service defines what layer can do (but not how it does it)
- *Protocol* = set of rules governing data communication between peer entities, i.e. format and meaning of frames/packets
- Service/protocol decoupling very important

Internet Protocols vs OSI



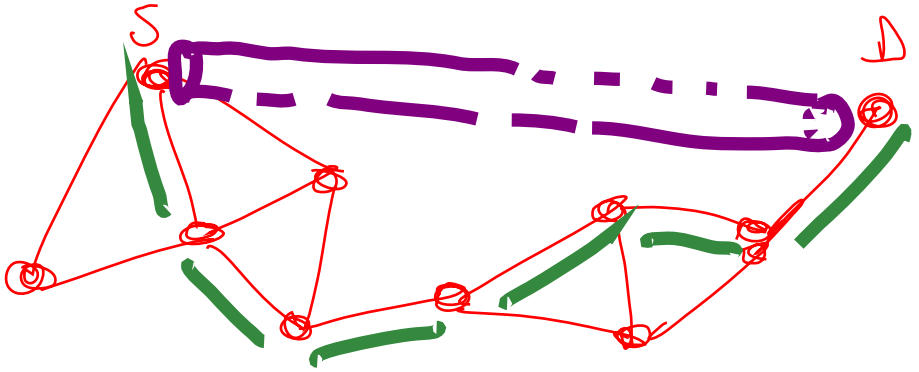
- Explicit Presentation and session layers missing in Internet protocols
- Data Link and Network layers redesigned



Functionalities

- Network Access Layer
  - includes the functions that in the OSI model are included in the physical and link layers and the low network layer
  - the service offered at the upper layer can be connection-based or connectionless
- Internet layer
  - enables the interconnection of the various component subnets with functionality that in the OSI model is located in the network layer
  - provides a connectionless layer service
  - uses the Internet Protocol (IP)

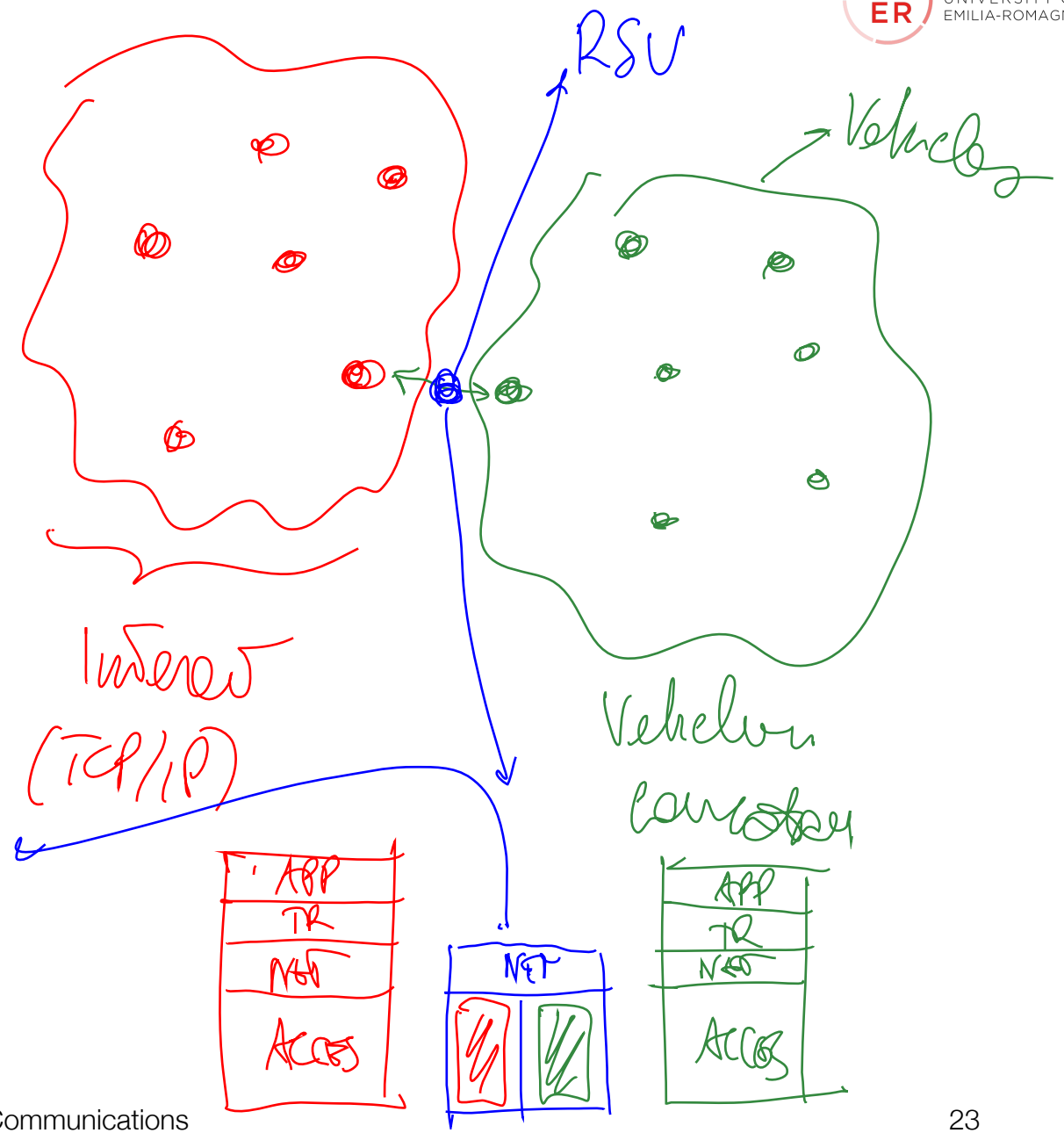
Functionalities



- Transport layer
  - corresponds to the transport layer and part of the session layer in the OSI model
  - Two types of service: reliable with connection or simpler without connection
- Application layer
  - corresponds to part of the session layer and the presentation and application layers
  - encapsulates all application-type protocols

Outline

- The OSI and Internet models
- Communication models
- Delimitation
- Sequence control
- Error management



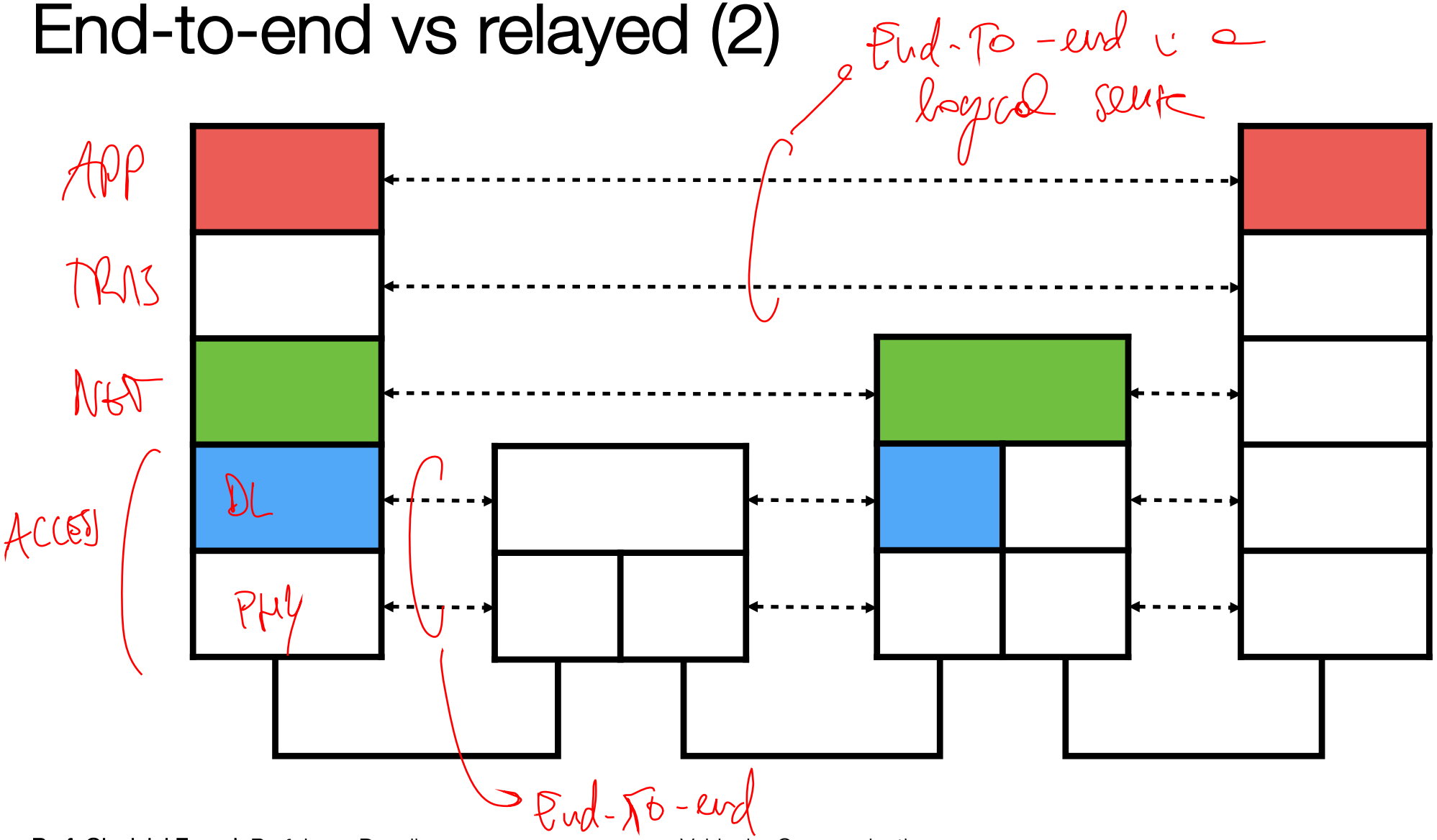
Communiction models

- Based on the entities involved in the communication
  - end-to-end and relayed (location)
  - unicast, multicast and broadcast (number)
  - client-server and peer-to-peer (role)
- Based on the mode of Information Unit (IU) transfer
  - with or without connection
  - reliable/real-time (QoS)
  - message-oriented and stream-oriented

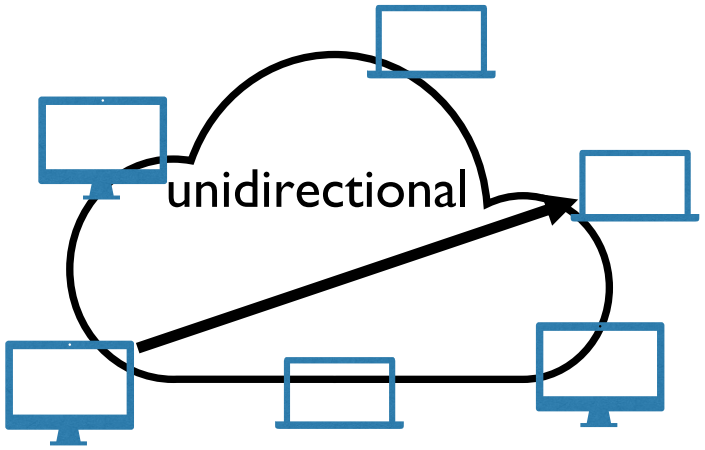
End-to-end vs relayed (1)

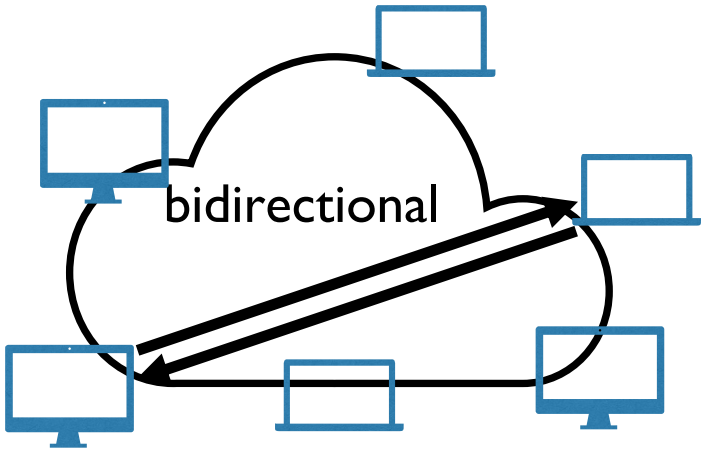
- End-to-end communication
  - directly between source and destination entities
  - does not require node addressing but only user addressing
- Relayed/switched communication (relayed)
  - occurs through the relaying of one or more intermediate nodes
- By extension a communication protocol may be end-to-end or relayed

End-to-end vs relayed (2)



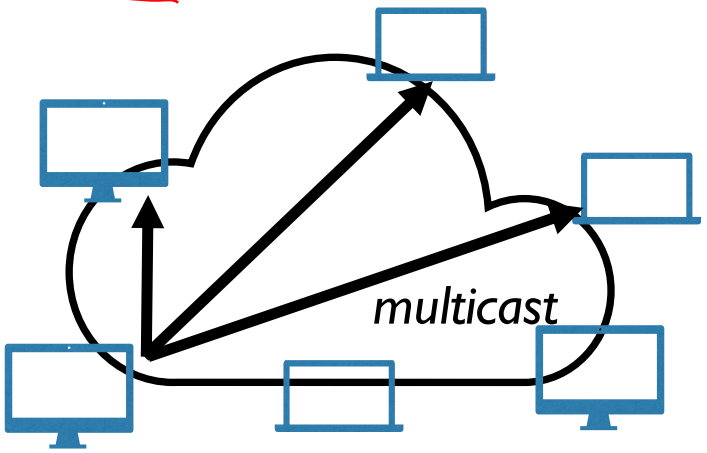
A.2 Unicast Communications

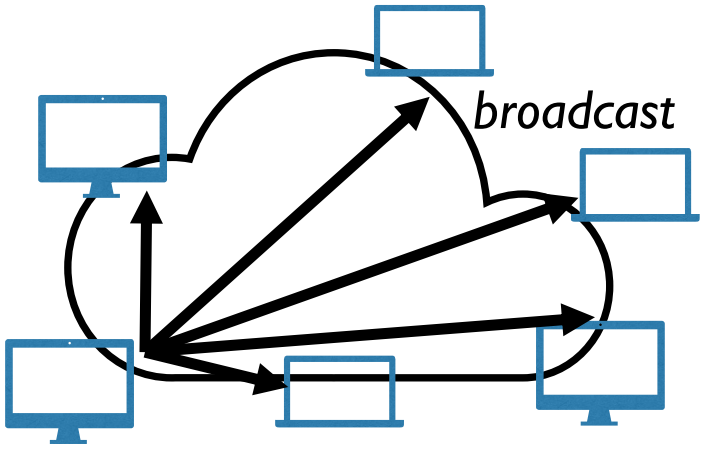




- 1 source and 1 destination
- Unidirectional or bidirectional
- Example: Classical telephony (fixed and mobile)

Multicast communications



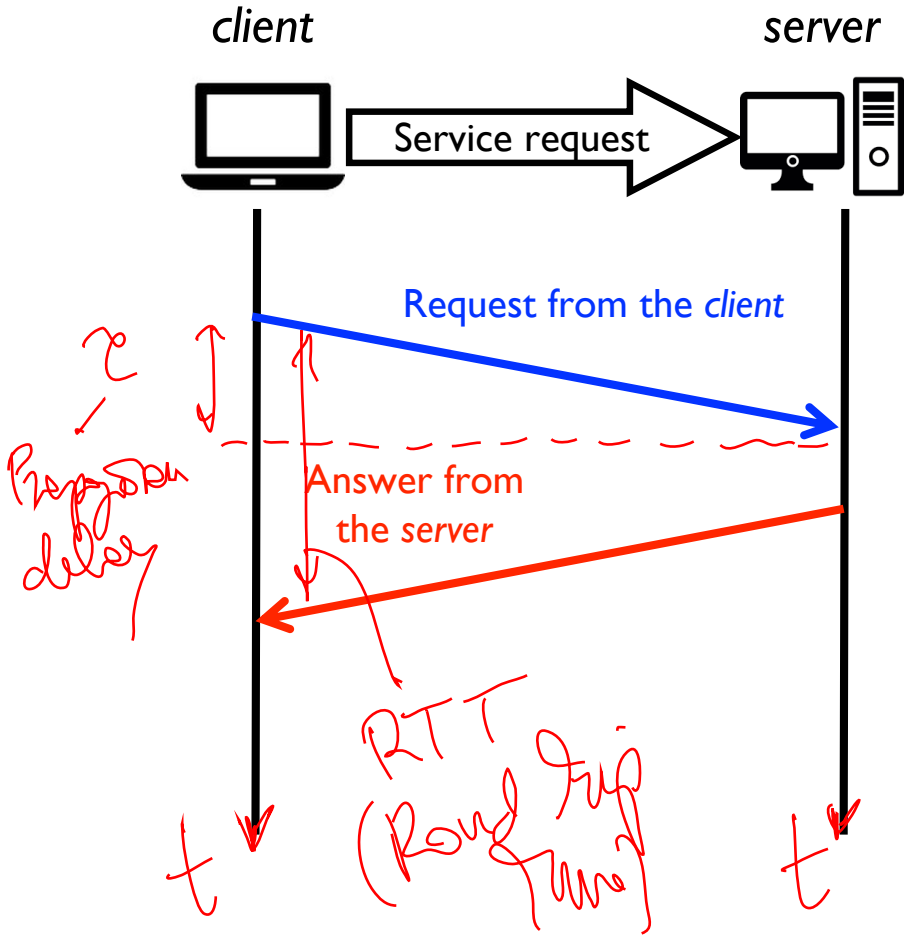


- Multicast: 1 source and multiple destinations
- Advanced telephone services (three-way calling)
- Broadcast: 1 source and all possible destinations
- TV, local networks

Protocols' Operations

- Some protocols support
  - only unicast mode
  - both unicast and broadcast
  - all three modes of communication (example: IP)
- Broadcast and multicast can be realized
  - taking advantage of any features of the underlying protocol
  - using multiple unicast communications at the underlying layer

Client-server model



- Direct communication between 2 entities with distinct roles
  - client: initiates communication and/or requests a service
  - server: opposite role
- The client forwards (service) requests to a server
- The server processes the requests and, if the request is accepted, responds to the client by providing the requested service
- Generally, the communication results in asymmetric type of communication

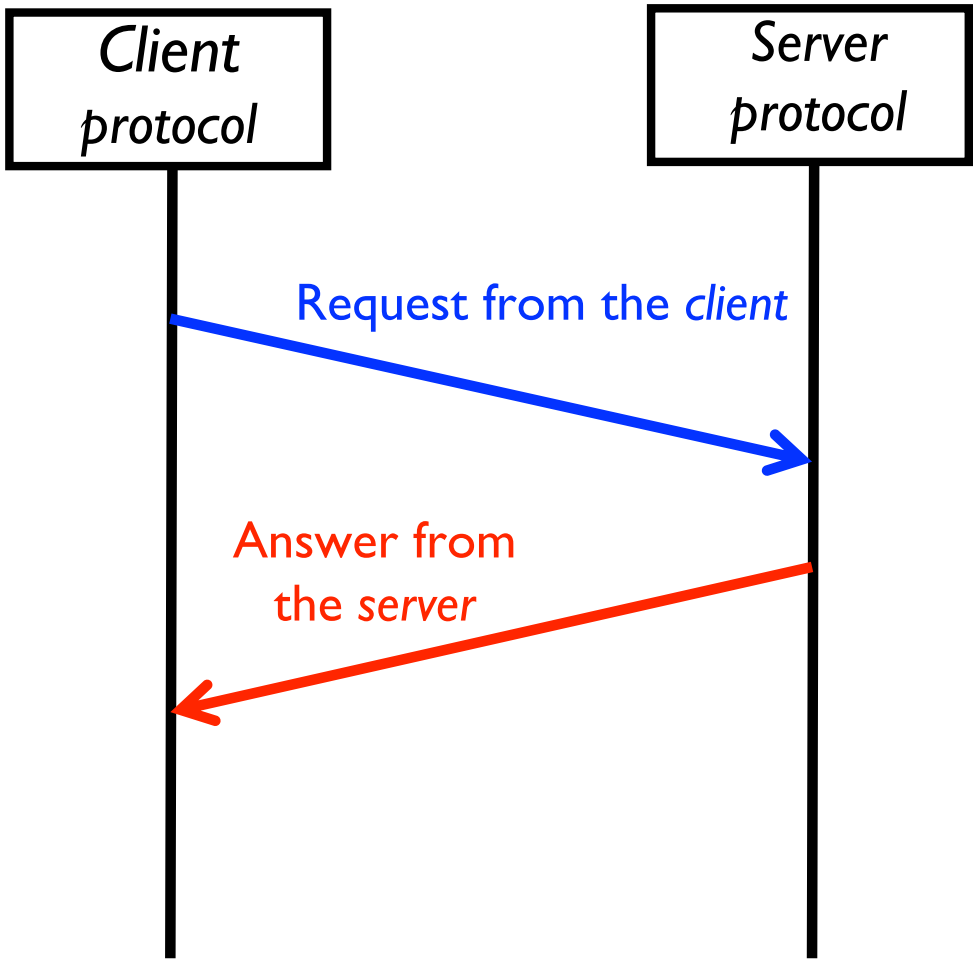
Client-server: definitions

- Client application or terminal process
  - calling party (*caller*, calling party) or requester
  - By extension terminal device that primarily hosts client-type applications
- Server application or terminal process
  - calling party (*callee*, called party) or respondent
  - by extension terminal device hosting primarily server-type applications

Client-server: examples

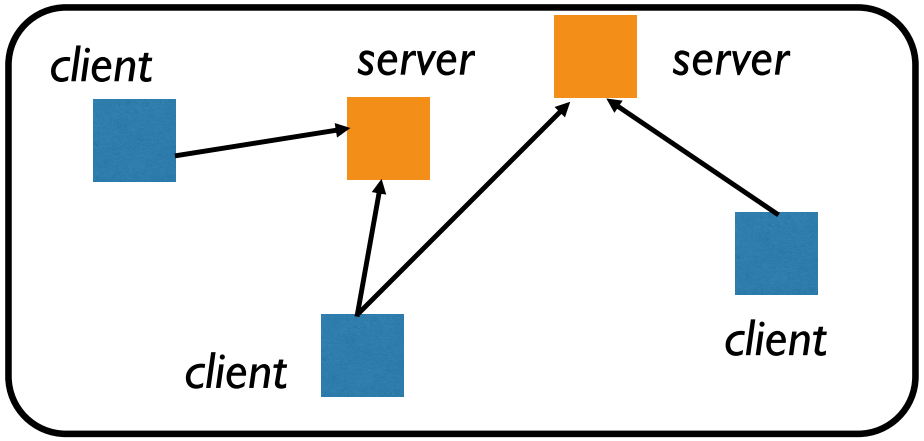
- Browser (client) vs web server (server)
- Email client (client) vs POP3/SMTP server (server)
- Some applications perform both the client and server sides of a communication
  - If hosted applications also have a server side, this side is often inhibited (by configuration, usage, or protection)

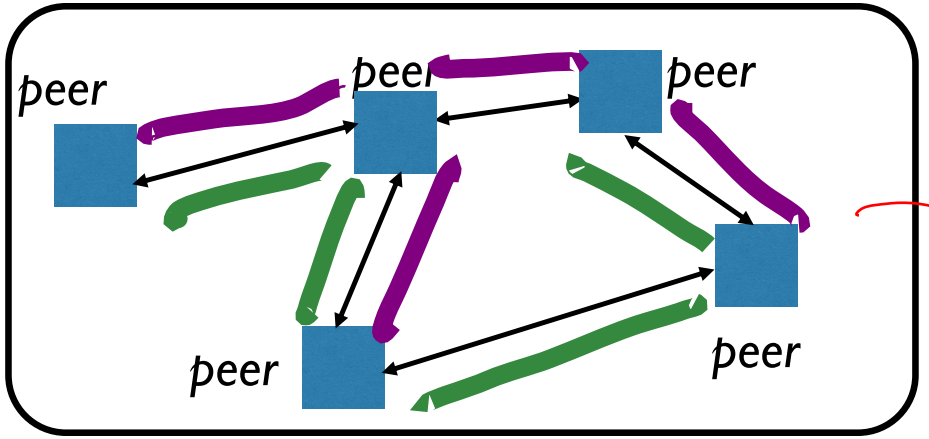
Client-server protocols



- A protocol is defined as client-server when the communication between the parties involved evolves in client-server mode
- Some protocols provide that the two parties can exchange the client or server function within the same communication
- In some cases the client or server mode characterizes only the first part of the communication, and the rest of the communication evolves symmetrically

Peer-to-Peer (P2P) Model





- Homogeneous nodes, generically referred to as peers
  - nodes act as both client and server
  - no distinction between intermediate (relay) and terminal nodes

Ad hoc network (no infrastructure)
⇒ Wifi in ad-hoc mode.

P2P Network Overlays

- In the case of routing based on node identifiers, the mapping of the nodes to the underlying network addresses and/or any routing is done through a special network structure called a P2P overlay
- Peers cooperate together in maintaining the overlay
- In some cases, a distinction is made between different types of nodes depending on their role in the network architecture (bandwidth, processing/storage capacity, etc.)
- Classification
  - **Unstructured** networks: all peers are equal (no structure)
  - **Structured** networks: the overlay is organized with a specific topology

Characteristics of a P2P network

- Characteristics of network architecture
  - **dynamic** (concepts of join, leave, and churn)
  - **reliable** (redundancy of communication resources and services provided)
  - **scalable** as the number of peers increases
- Main advantages
  - reliability and fault-tolerance (fault-tolerance)
  - automatic distribution of functions and services (self-organization)