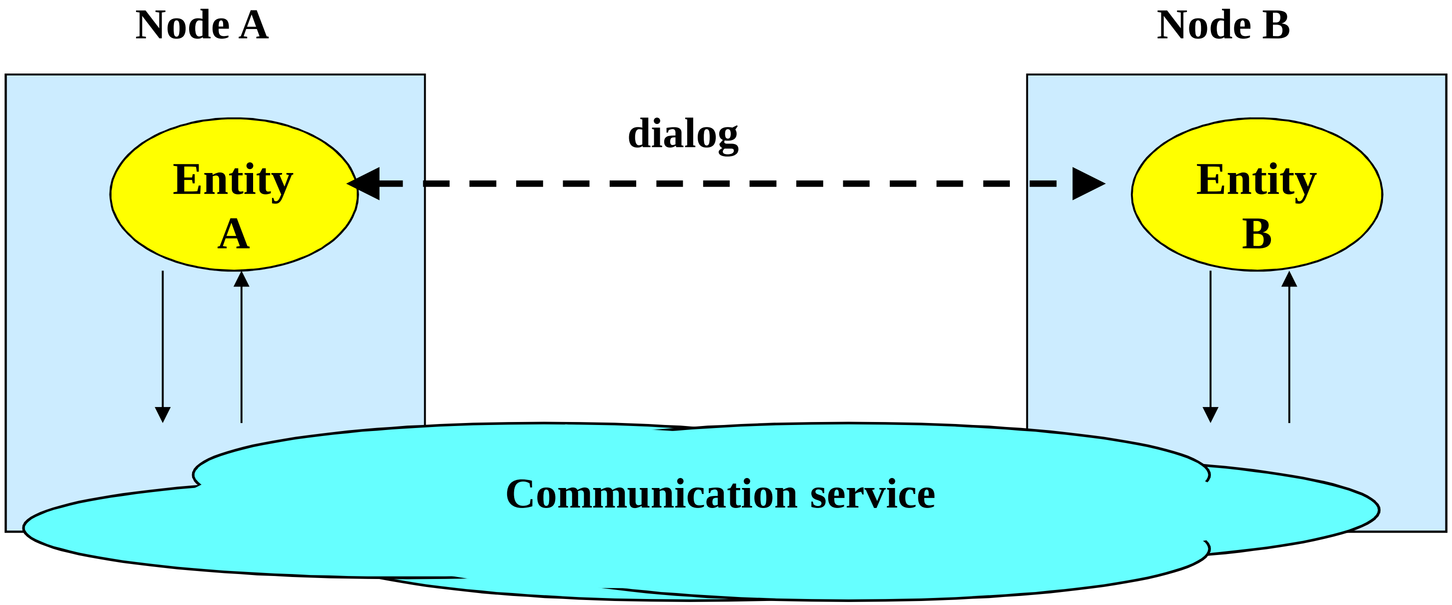
Reti Internet Multimediali

Prof. Fabio Martignon

Fundamentals of Protocols and Communication Services

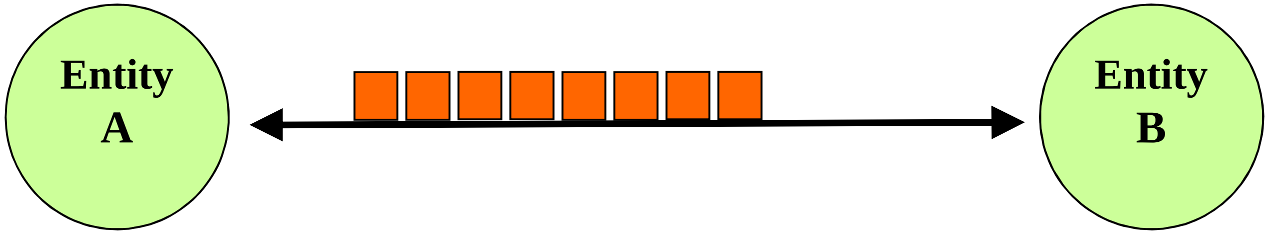
Communication Service

- Given two or more remote entities …
- … a *communication service* provides information transfer between the entities



Communication Service

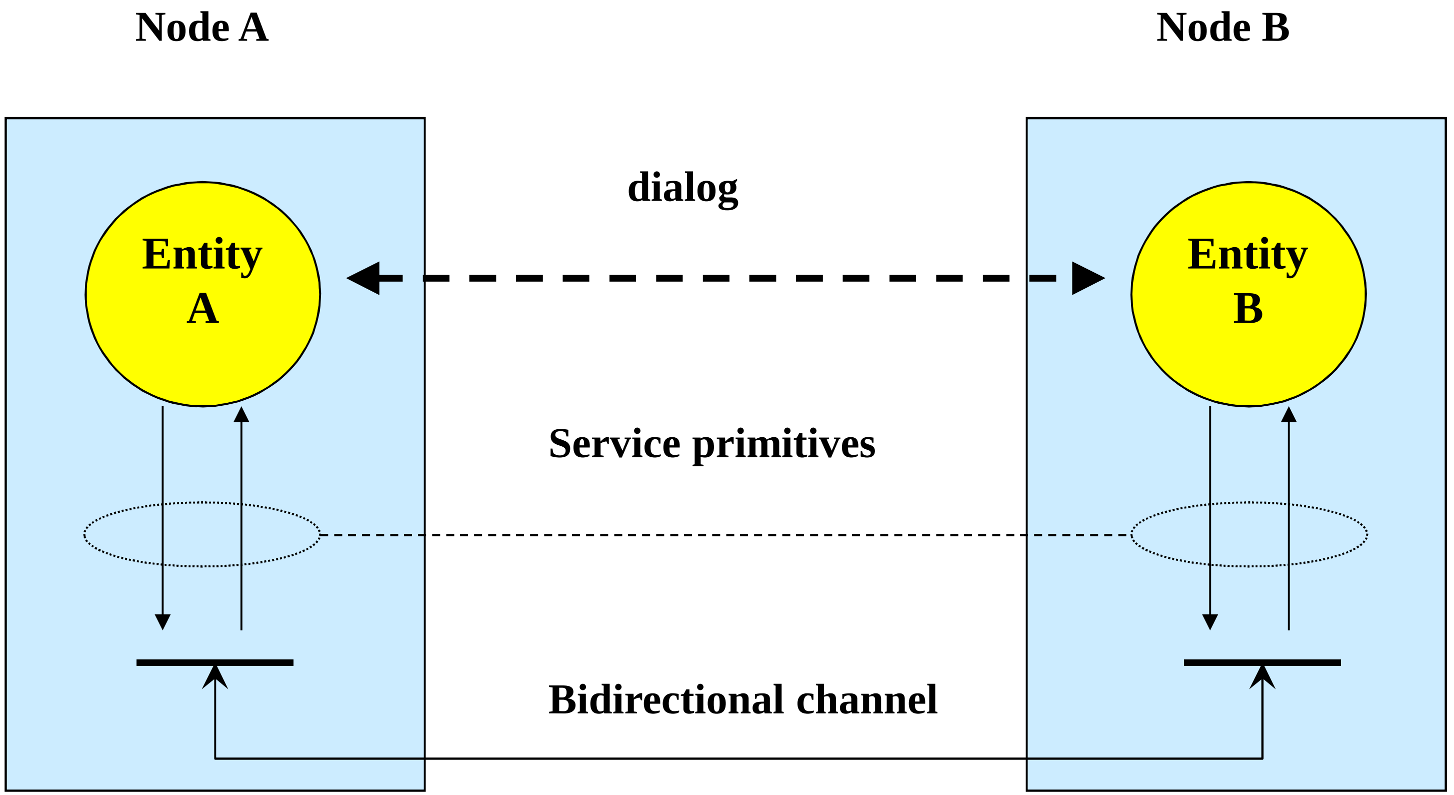
- Manages the exchange of information between two entities
- In general, it transfers information units:
  - words
  - bits
  - bunch of bits (frames or packets)
  - files
  - Multimedia flows



Communication Service

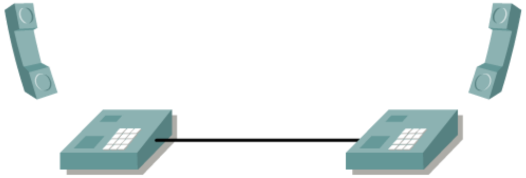
- May be described through *service calls* named *service primitives*
- The service primitives can be used to:
  - *Describe the service*
  - *Request the service*
  - *Gather info on the service*
- The service primitives are characterized by:
  - *The type of info to transfer*
  - *The destination address*
  - *The features of the required service*
  - etc.

The Service Primitives



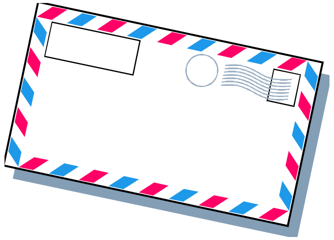
Types of Communication Services

- □ Connection Oriented

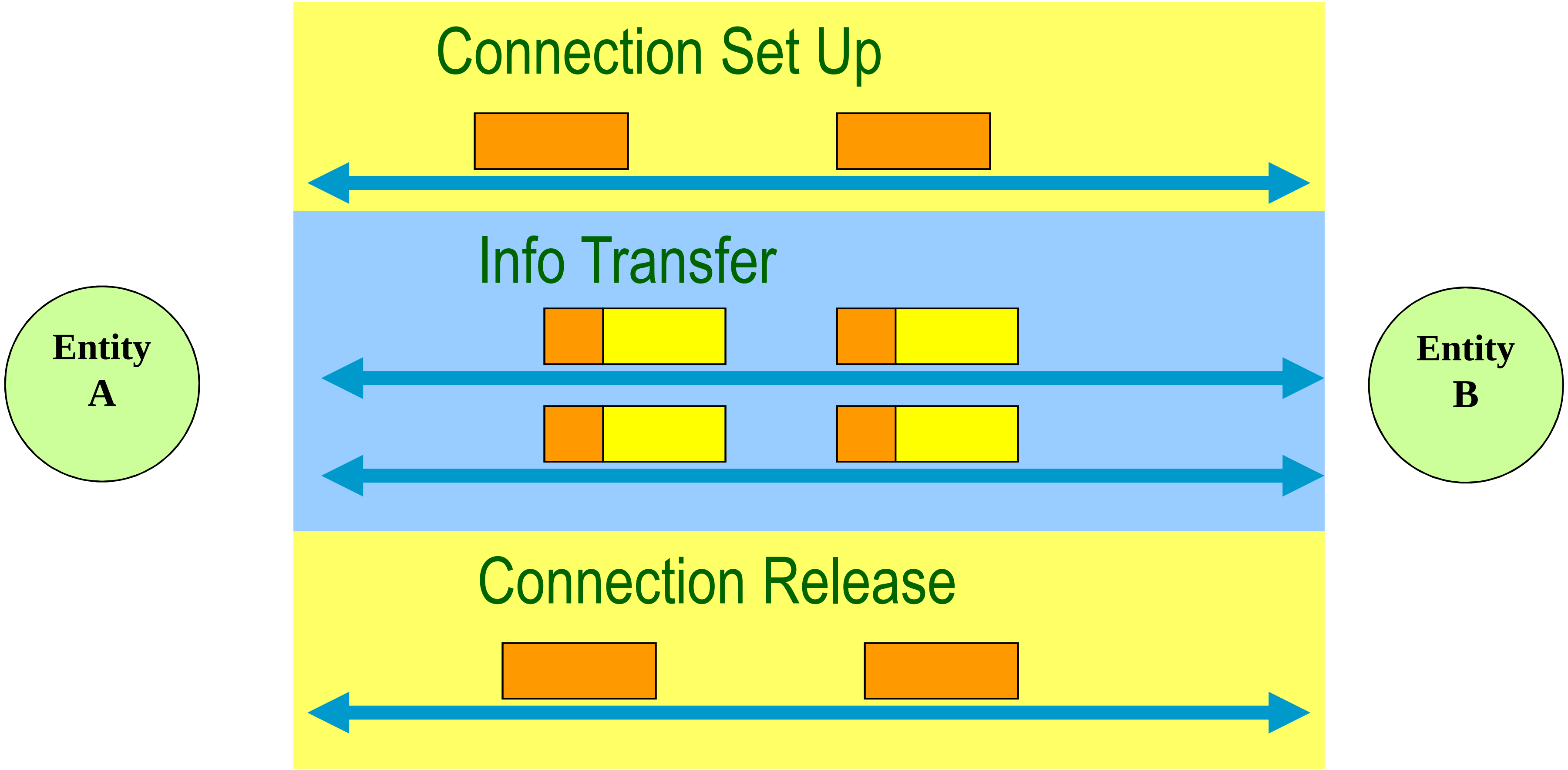


  1) Connection Set Up
  2) Data transfer
  3) Connection Release

- □ Connectionless
  - All in one
  - Asynchronous

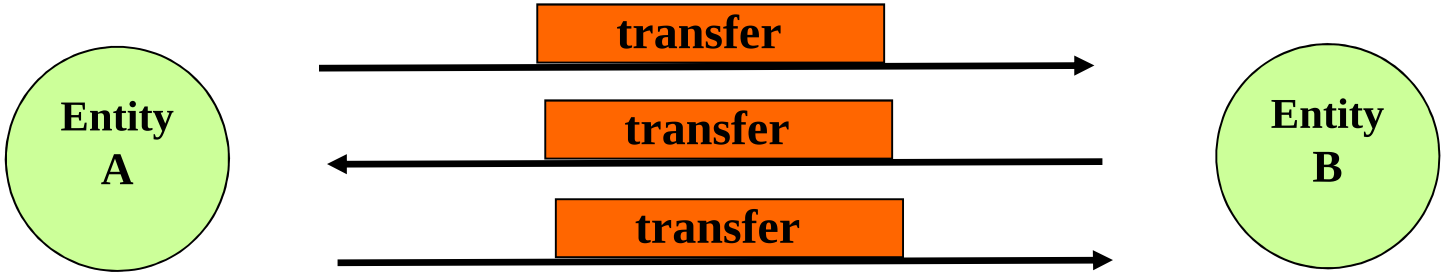


Connection Oriented Services



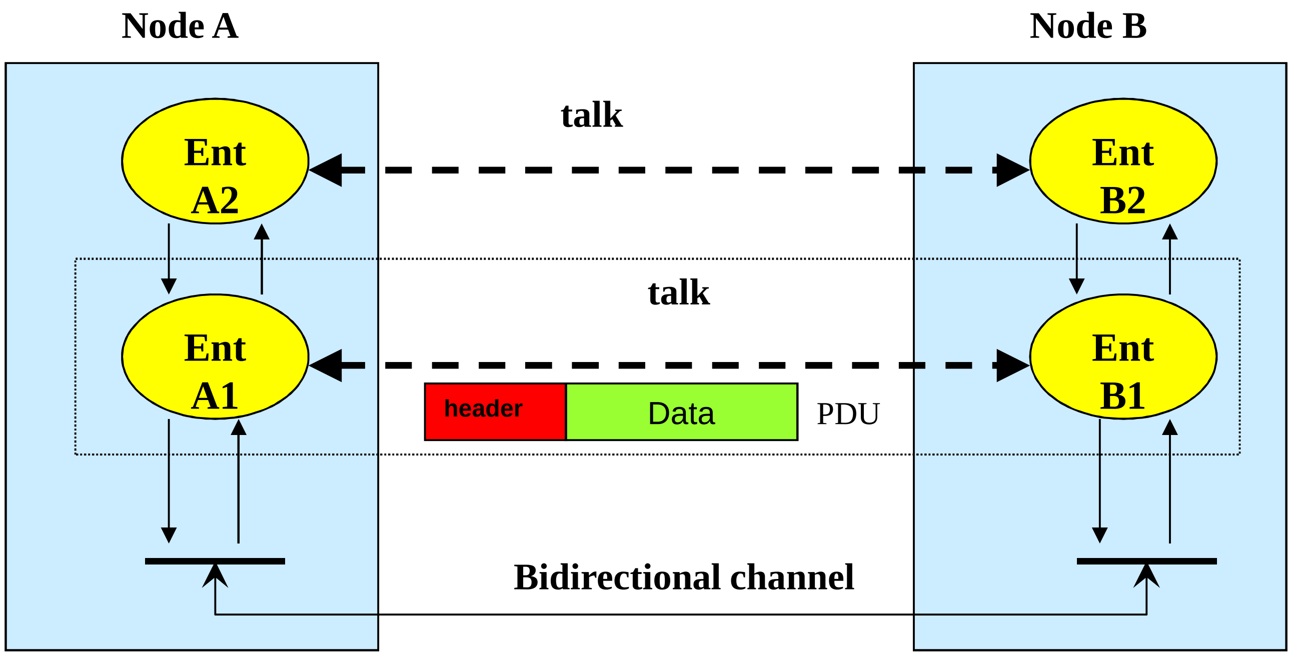
Connectionless Services

- □ Lack of set up coordination among entities
- □ different transfer sessions between the same entities may not be related
- □ problems in implementing the typical connection oriented services (VoIP)



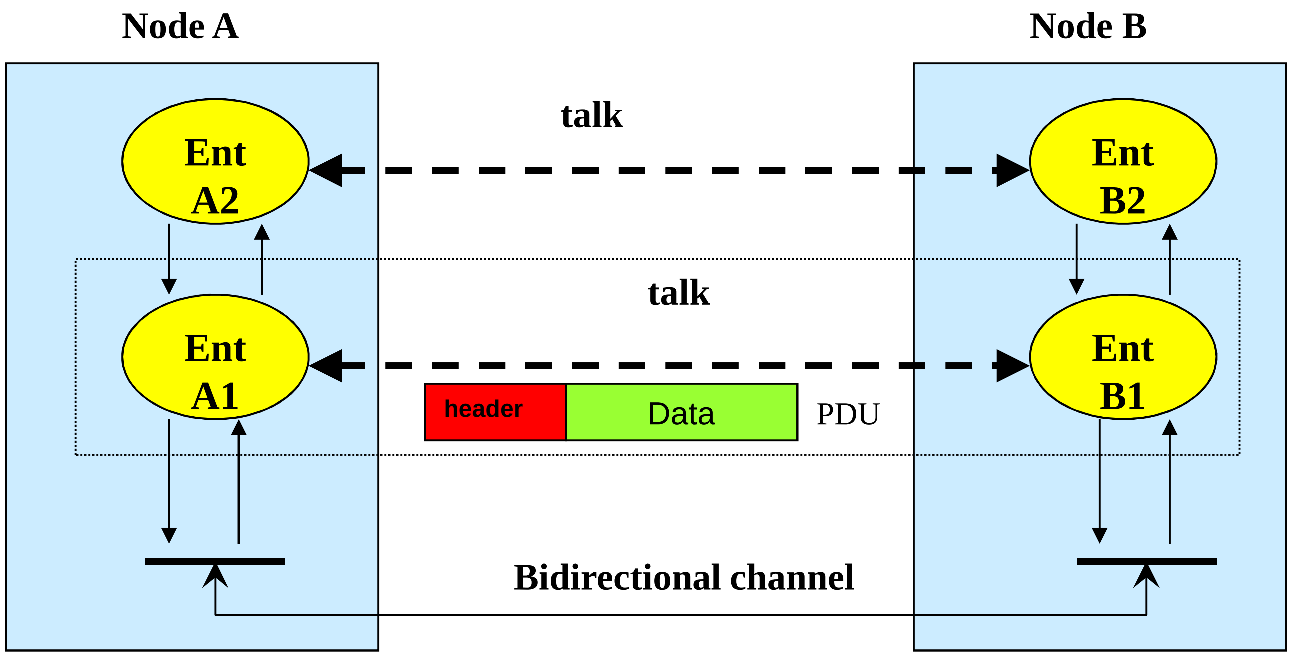
Layers

- Two entities at the same level may offer a communication service to *upper layer* entities



Layer *Functions*

- The communication service provided to the upper layer is “richer” thanks to specific *functions* implemented at the lower layer

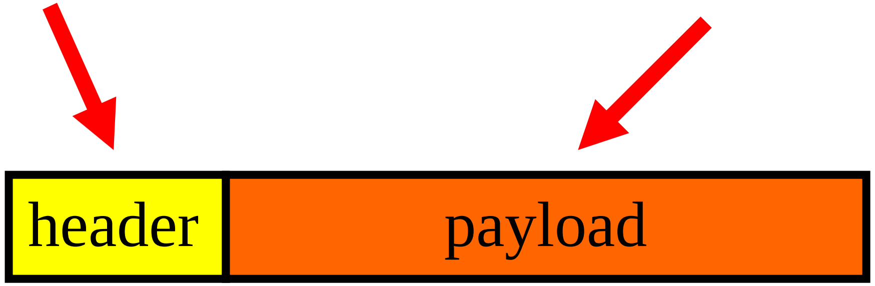


Communication Protocols

- Entities at the same level cooperate to provide *upper layer* entities with a communication service
- Entities at the same level exchange messages
- Protocol:
  - *Set of rules which handle the communication among entities at the same level*
    - Message format
    - Information on the service
    - Info transfer procedure
    - etc.

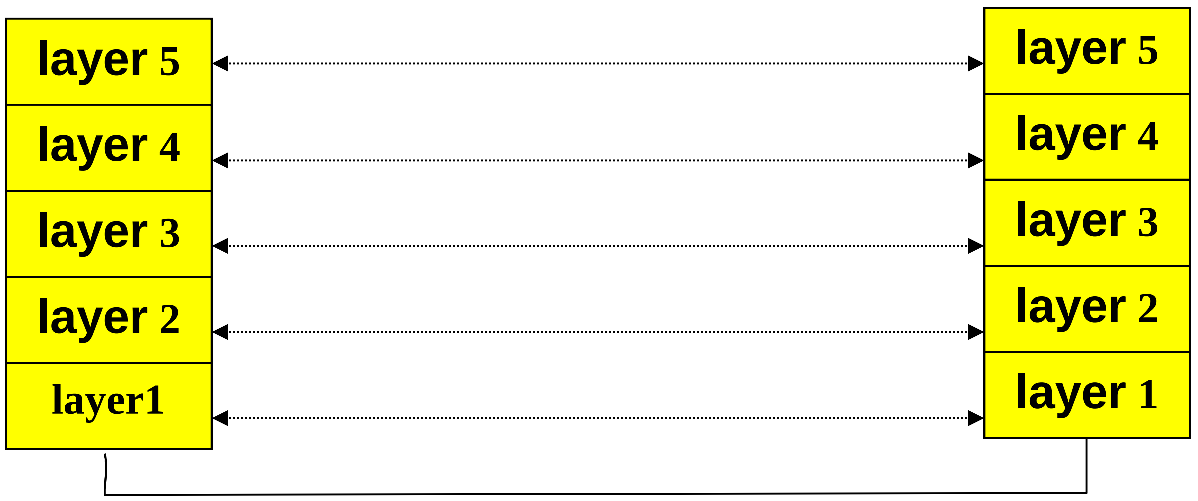
Packet Data Units **(PDU)**

- Information Units used within a protocol by entities at the *same level*
- PDUs may comprise:
  - **Signalling information (header)**
  - **Data received by upper layers (payload)**



Layered Architecture

- Complex Communication Services may be organized in layers
  - From a layer handling the bits …
  - … to a layer handling files and/or more complex objects and protocols



Protocol "Layers"

Networks are complex!

- many "pieces":
  - hosts
  - routers
  - links of various media
  - applications
  - protocols
  - hardware, software

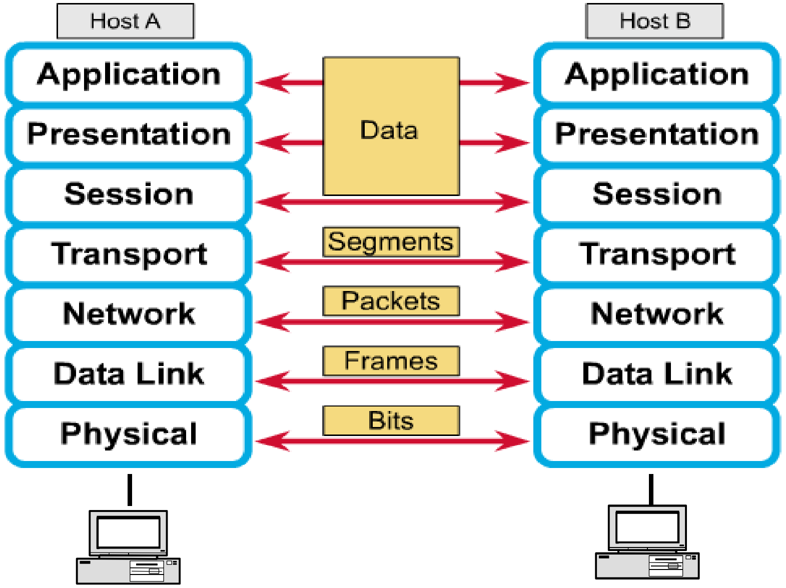
Question:

Is there any hope of *organizing* structure of network?

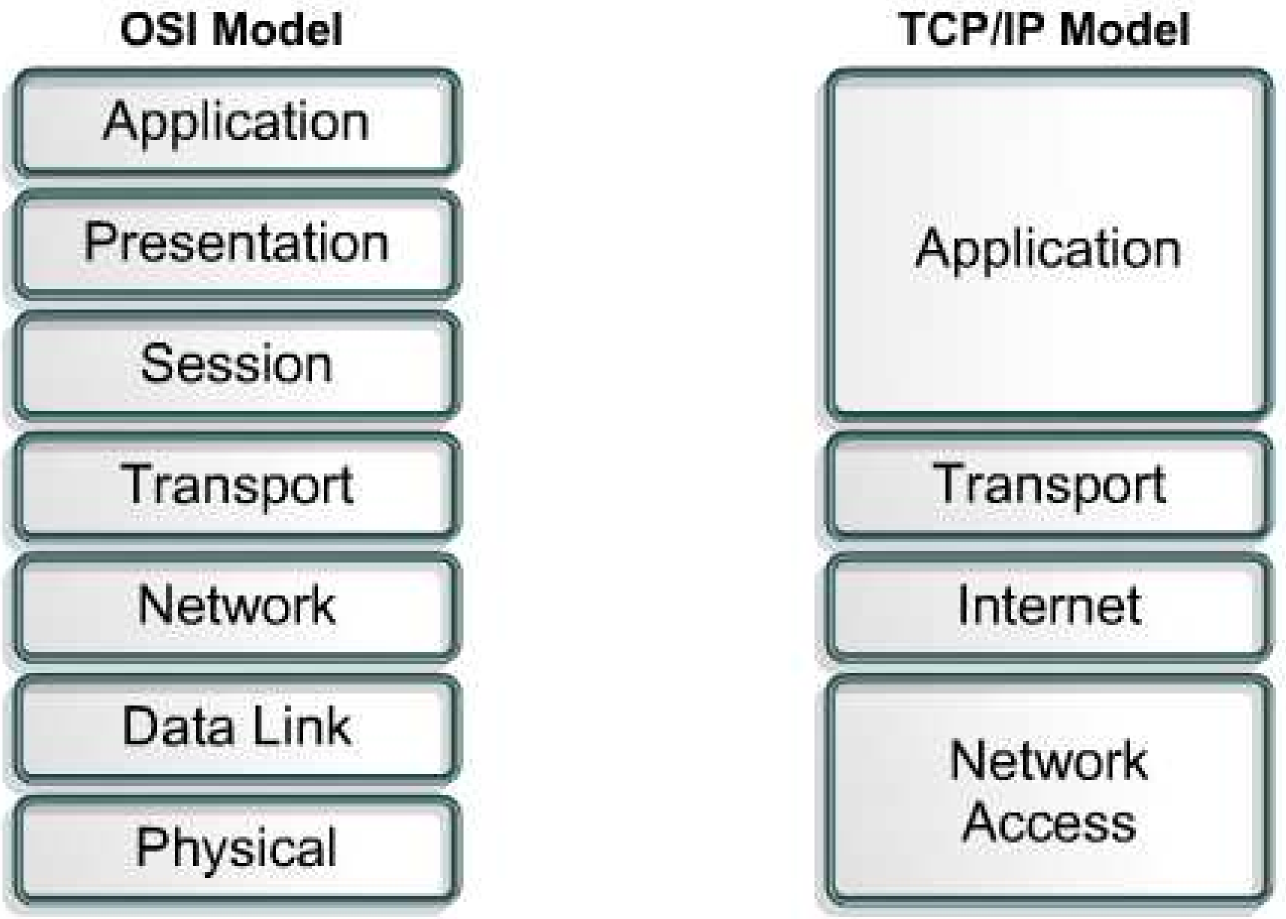
Or at least our discussion of networks?

Layered Architecture: PROs

- Reduces complexity
- Standardized Interfaces
- Fosters Modularity and Interoperability
- Ease Up Teaching
- OSI Model (1974)



TCP/IP Suite vs OSI Model

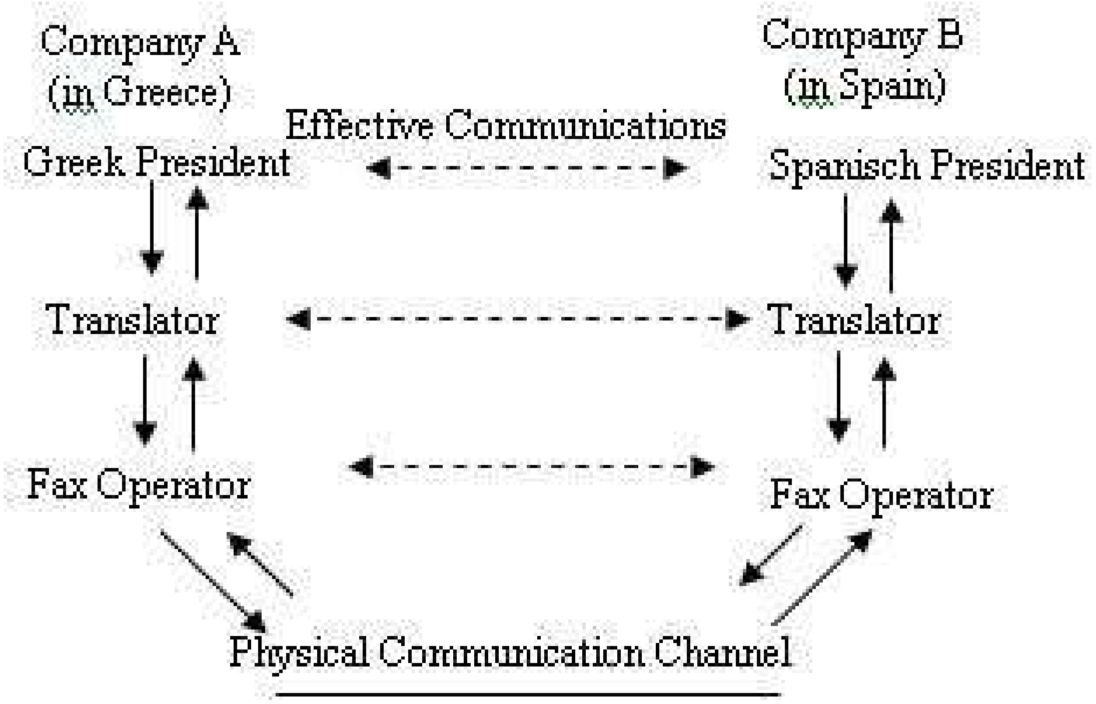


SMTP,
FTP, TFTP,
Telnet, Rlogin
SNMP, DNS

TCP, UDP

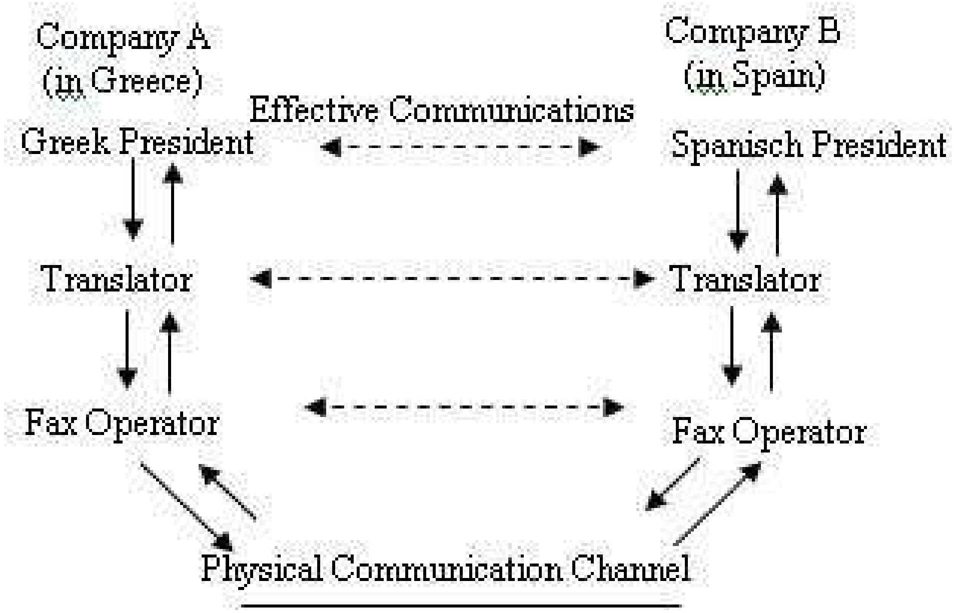
IP, ARP, RARP,
ICMP, IGMP

Company Organization
(Layering example in real life)



- a series of steps

Layering example



Layers: each layer implements a service

- via its own internal-layer actions
- relying on services provided by layer below

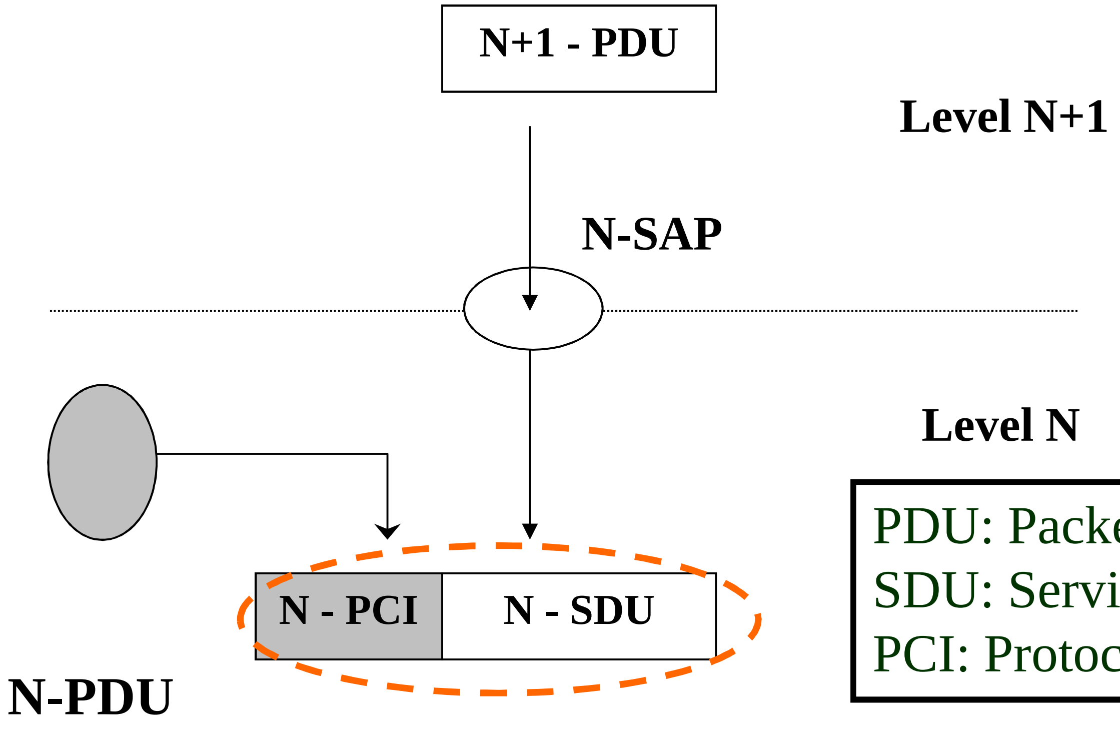
Why layering?

Dealing with complex systems:

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

Layer To Layer Interface

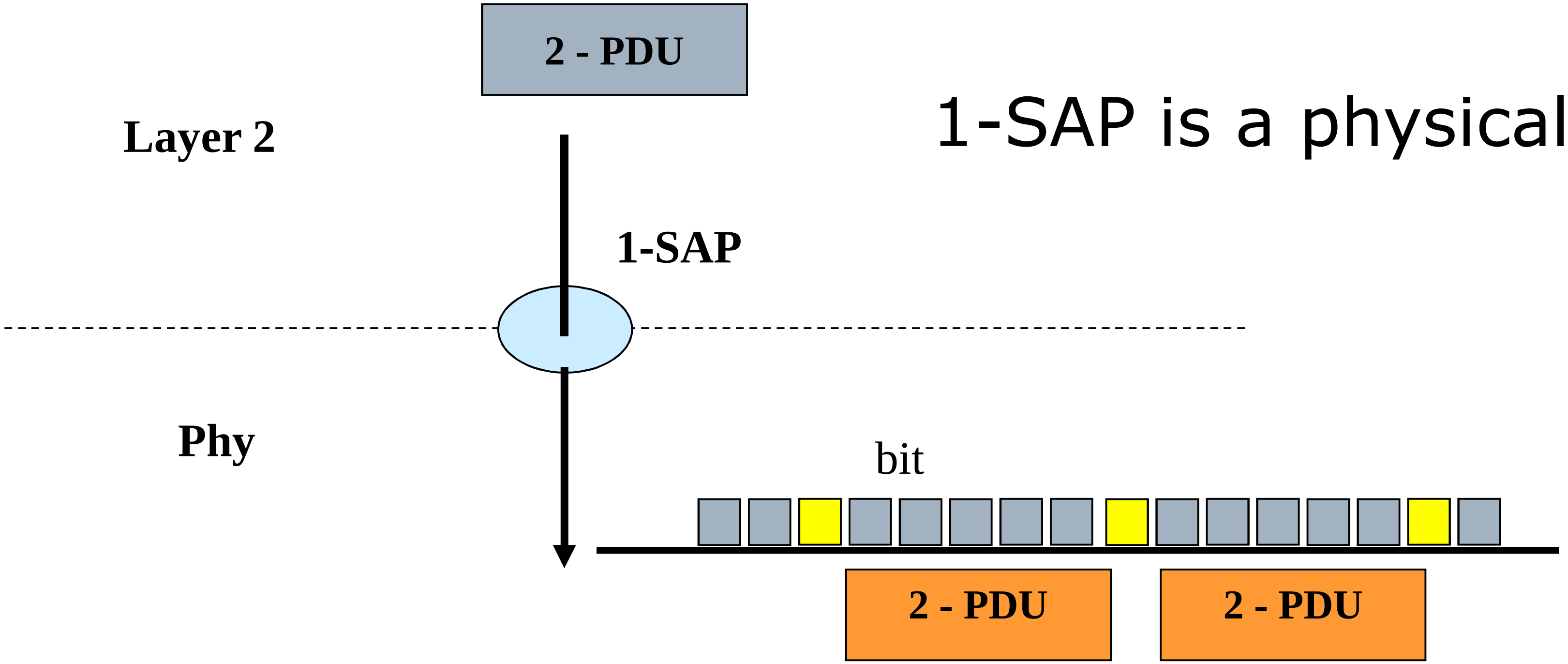
- The services offered by a given layer is characterized by a *Service Access Point (SAP)*



PDU: Packet Data Unit
SDU: Service Data Unit
PCI: Protocol Control Information

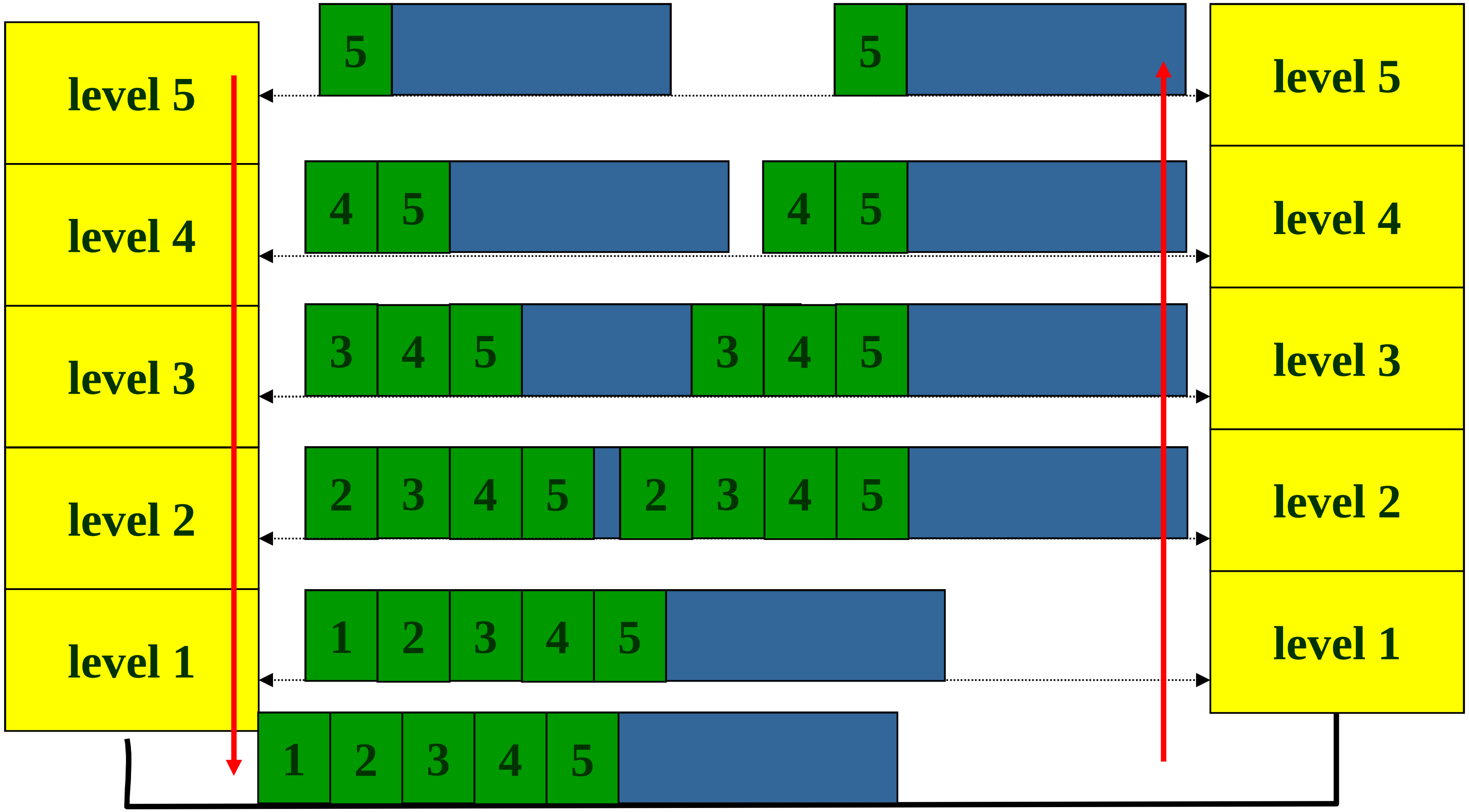
Layer To Layer Interface

- □ The lowest layer is the Physical one
- □ Phy-PDU = bit flows



1-SAP is a physical port

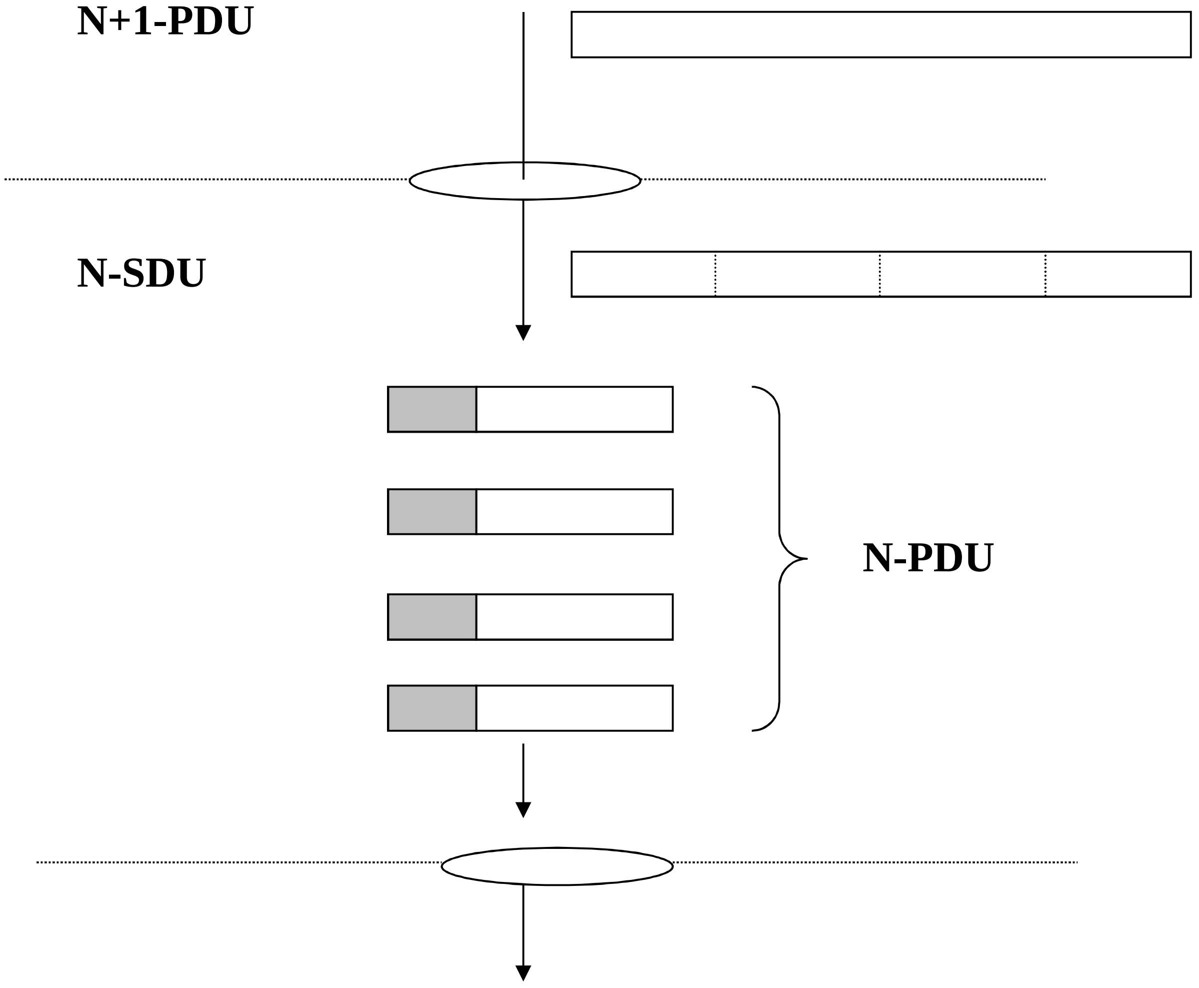
General Architecture



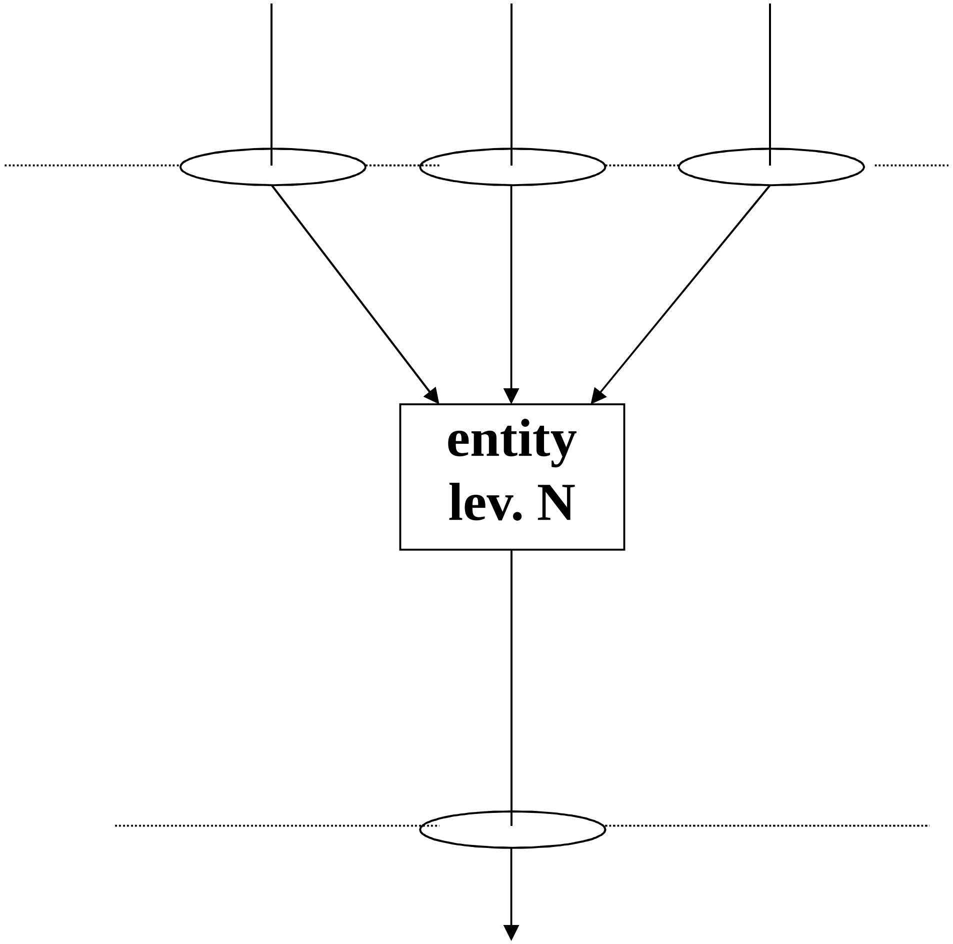
Functions

- Can be divided into:
  - Adaptation functions
    - multiplexing
    - segmentation
  - Enhancement functions
    - Error control
    - Sequencing

Segmentation

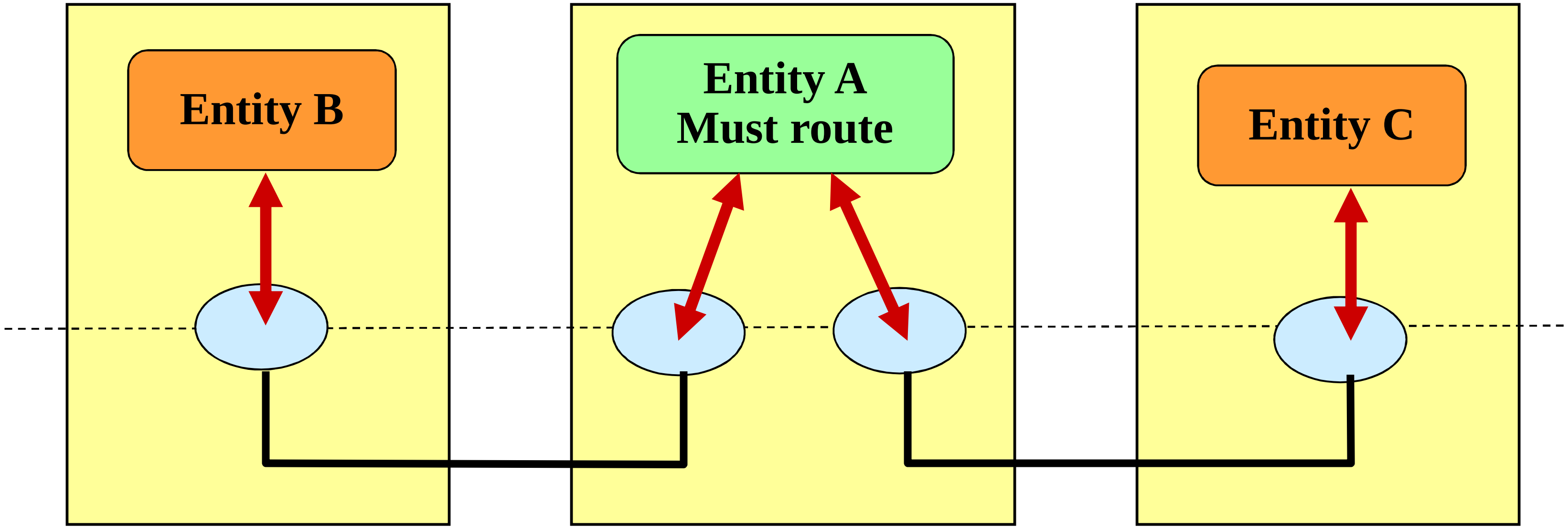


Multiplexing



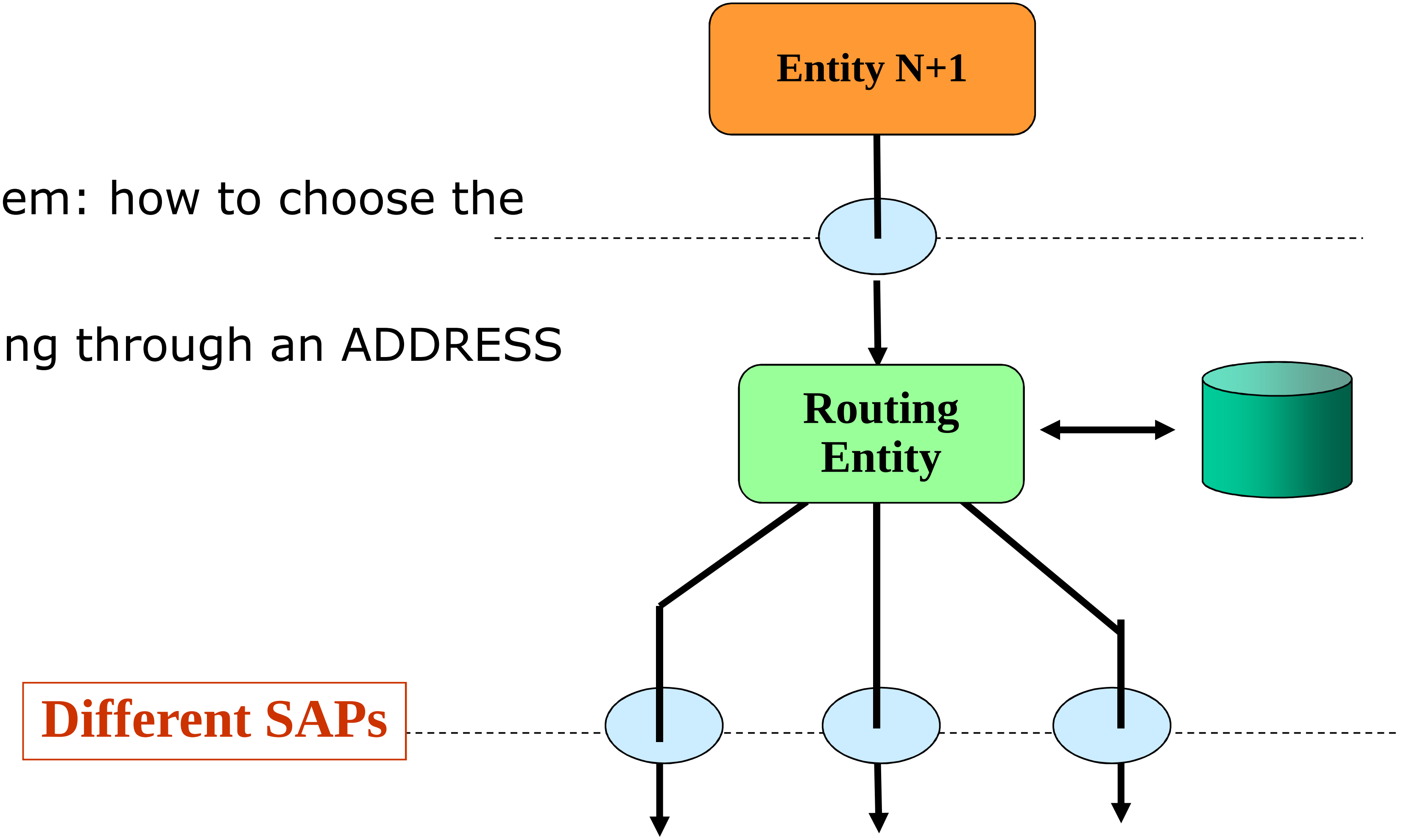
Networking Functions

- A given entity can communicate with multiple entities at the same level
- Need of Routing functionalities (SAP choice)



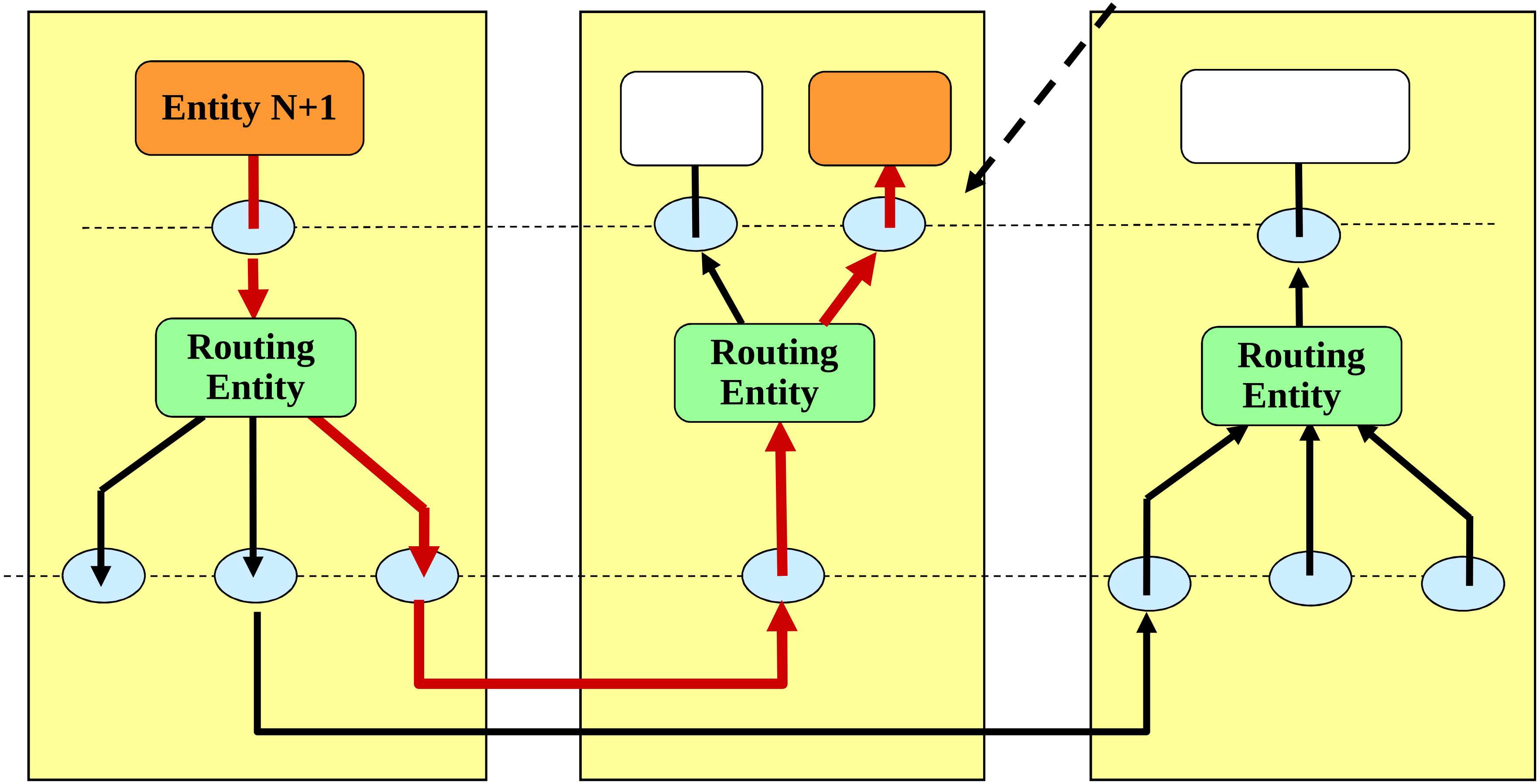
Routing

- Problem: how to choose the partner
- Routing through an ADDRESS



Addressing

- Identifies the destination N-SAP

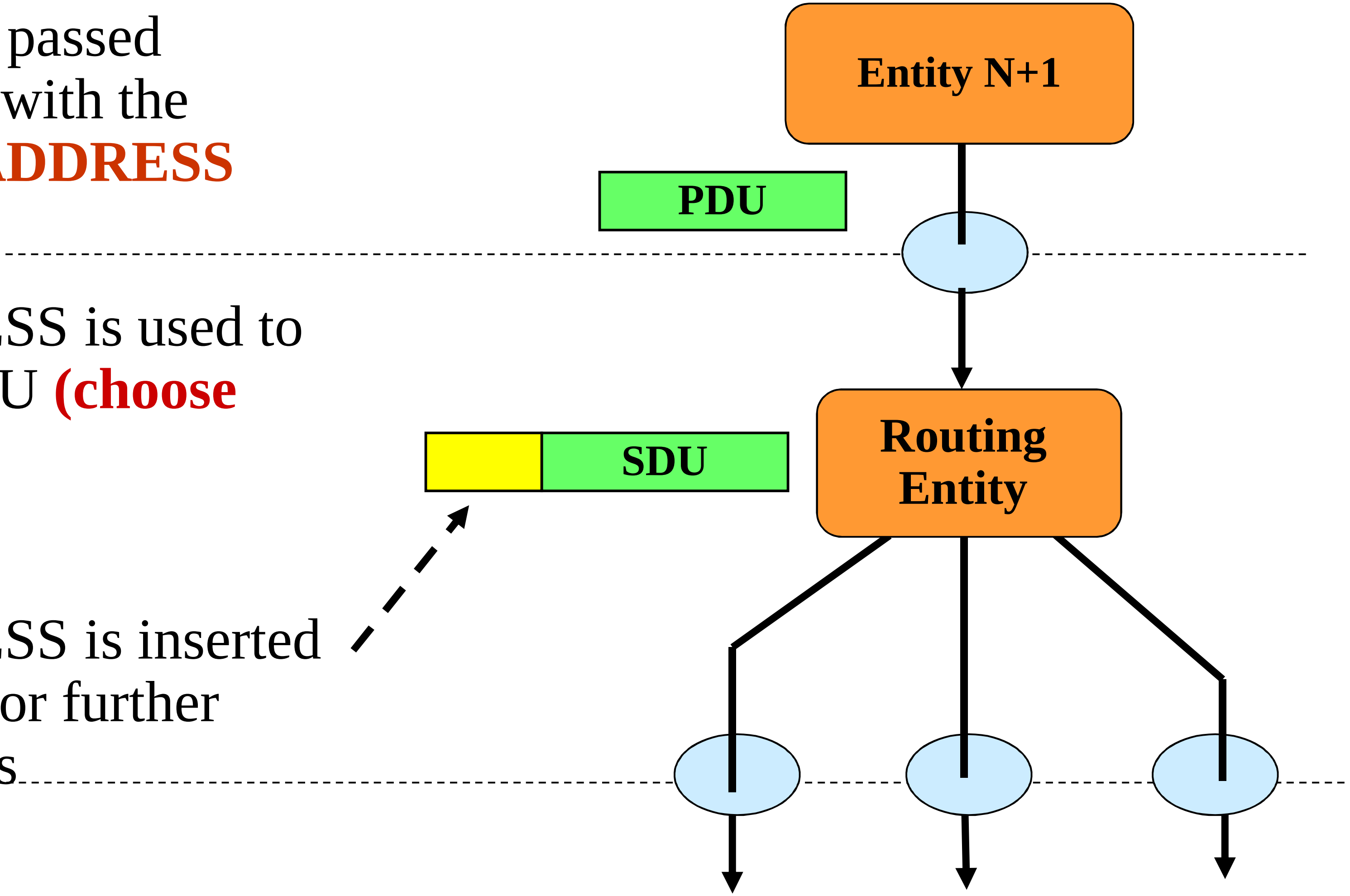


Addressing & Forwarding

The PDU is passed downwards with the parameter **ADDRESS**

The ADDRESS is used to route the PDU **(choose the SAP)**

The ADDRESS is inserted in the PDU for further routing needs

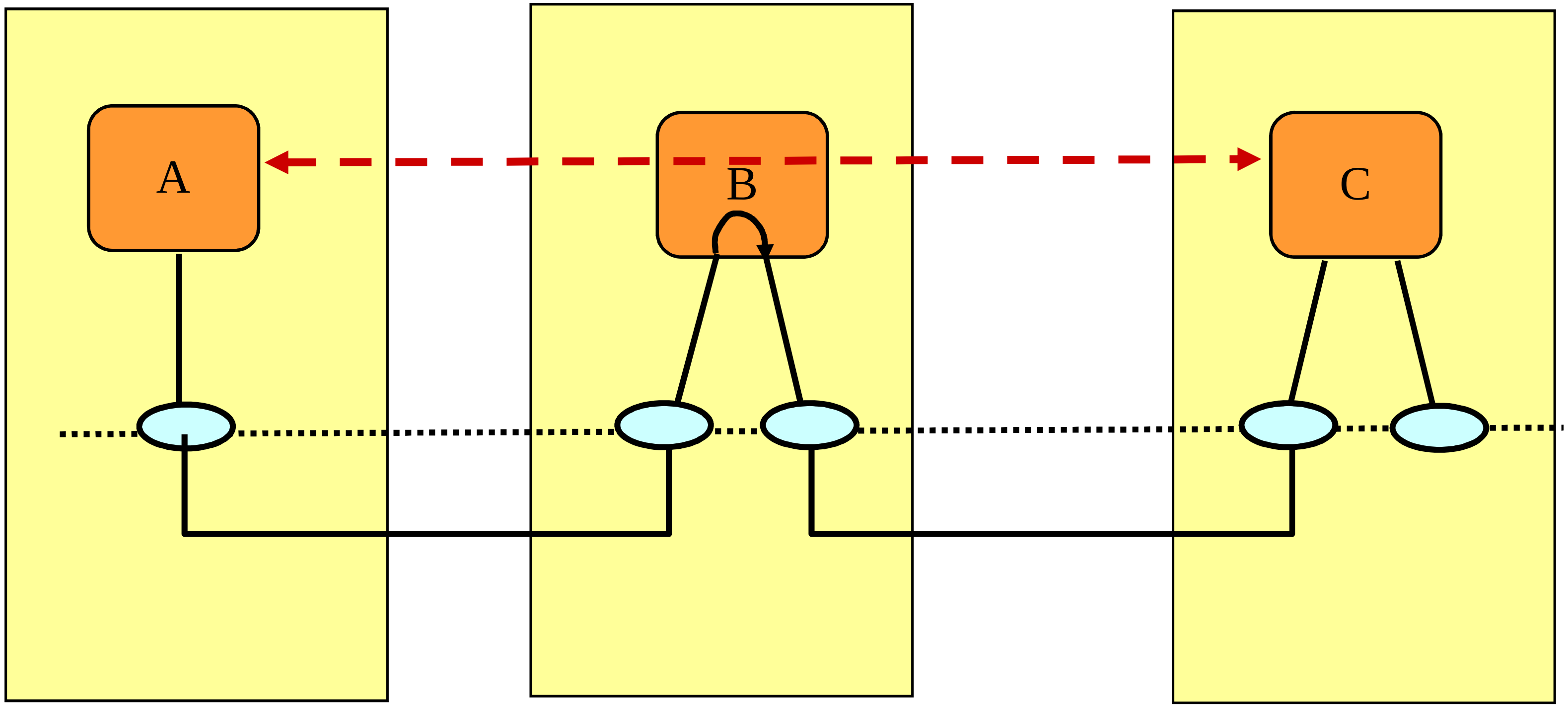


Addressing

- Address: SAP identifier, unique among those of the same level
- Types of addresses:
  - unicast: single SAP
  - multicast: groups of SAPs
  - broadcast: all the SAPs

Forwarding

- Once the SAP is chosen the PDU must be forwarded



Routing Tables

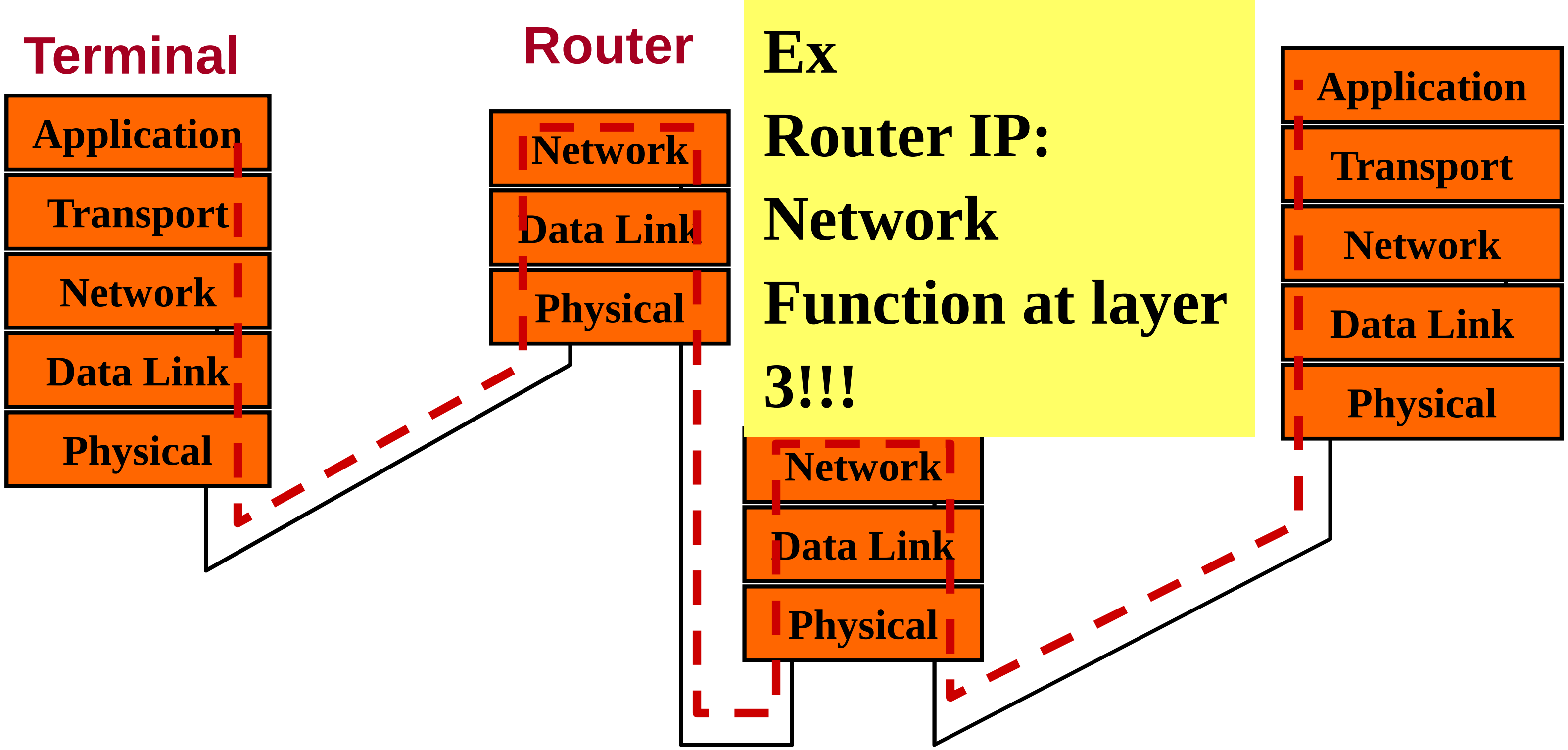
- Exit SAP chosen on the basis of the routing tables

Routing Table	
destination	Exit SAP

- Info gathering through routing protocols

Route To Destination (1)

- Through multiple nodes
- Network nodes go up to the network layer only



Ex

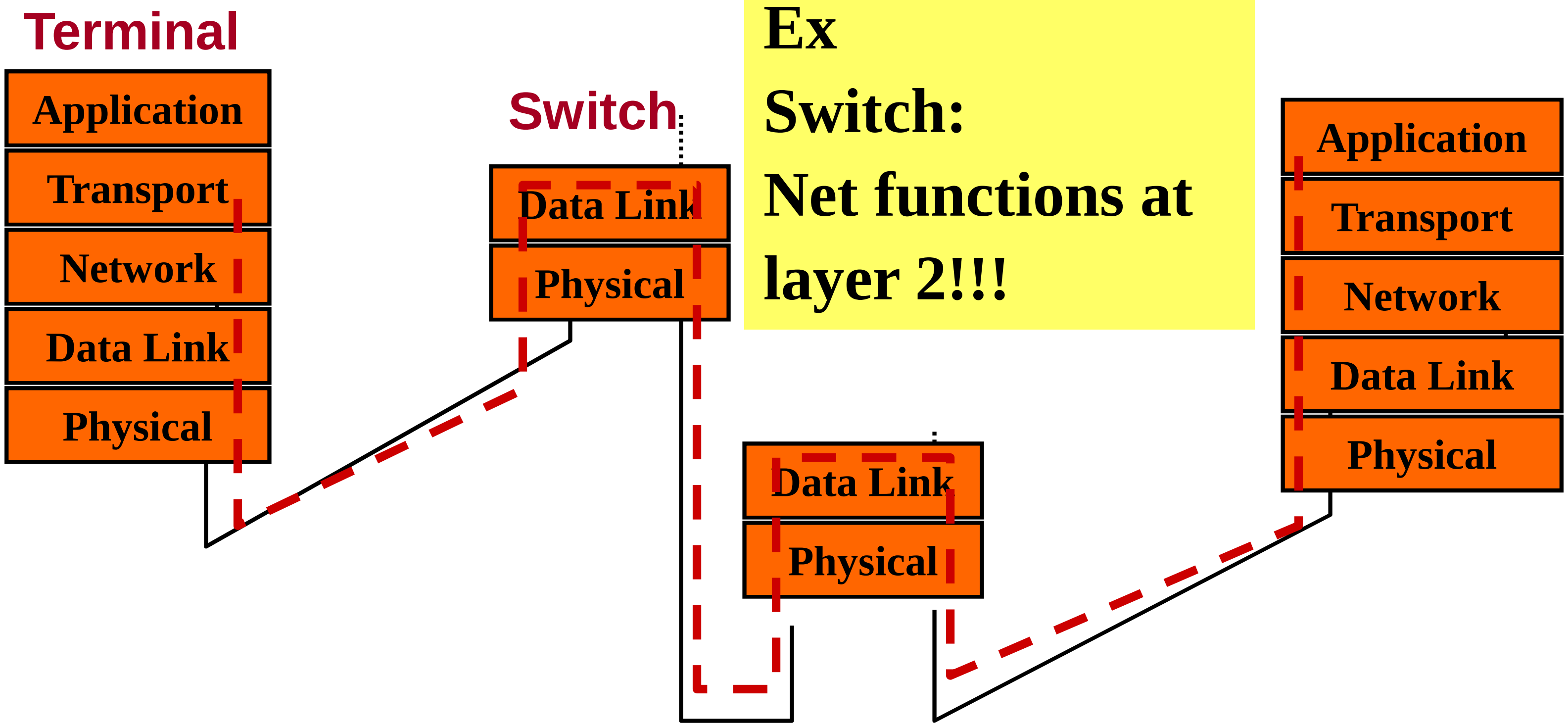
Router IP:

Network

Function at layer 3!!!

Route To destination (2)

- Routing function may be implemented at lower/upper layers
- LAN Switch



Terminal

Application

Transport

Network

Data Link

Physical

Switch

Data Link

Physical

Ex

Switch:

Net functions at

layer 2!!!

Data Link

Physical

Application

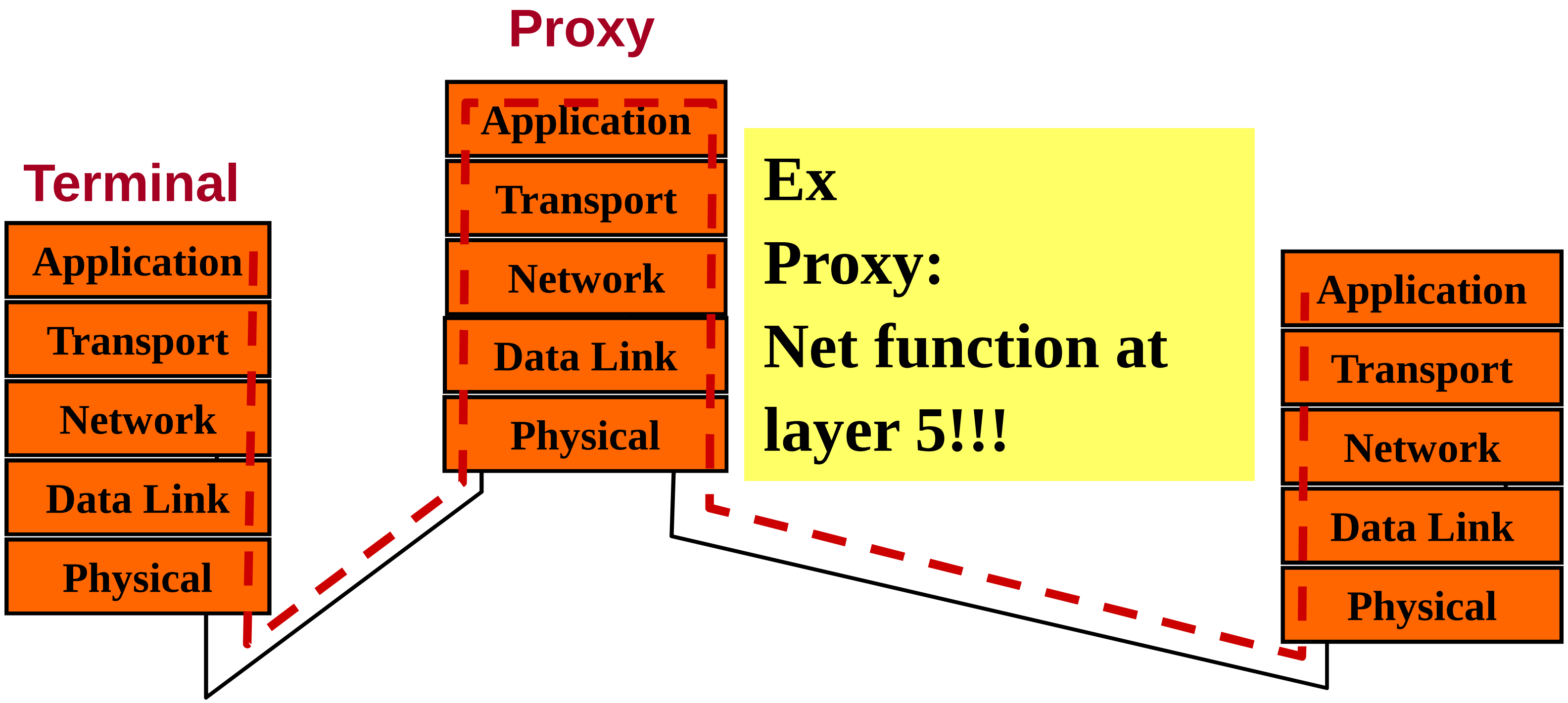
Transport

Network

Data Link

Physical

Route To destination (3)



Ex

Proxy:

Net function at

layer 5!!!