# Network Organization and Standards #4

Victor S. Frost
Dan F. Servey Distinguished Professor
Electrical Engineering and Computer Science
University of Kansas
2335 Irving Hill Dr.
Lawrence, Kansas 66045
Phone: (785) 864-4833 FAX:(785) 864-7789
e-mail: frost@eecs.ku.edu
http://www.ittc.ku.edu/



## Network Standards and Open Systems

- Standards Organization Objectives
- Standards Organizations
- Open System Network Requirements
- Layered Architecture
- Goal: Understand how networks are described

## Network Standards and Open Systems: Need for Standards

- Enable interoperability of equipment from different vendors
- Facilitate the building of a large market to reduce prices



## Network Standards and Open Systems: Problems with Standards

- Freezes technology
- Multiple standards evolve for same system
- Standards take a long time to be established
- Difficult to evolve to meet rapidly changing needs
- Often standards are complex
- De-facto standards often emerge

## Network Standards and Open Systems: Objectives for Standards

- Fulfill need for standards thru:
  - Development
  - Establishment
  - Promulgation
- Co-ordinate activity
- Assure consensus
- Information focal point
- Mechanism for management



## Network Standards and Open Systems: Standards Organizations

- American National Standards Institute
  - Manufacturers
  - Organizations
  - Government
  - Users

## Network Standards and Open Systems: International Standards Organizations

- Internet Engineering Task Force (IETF)
  - Request for Comment (RFC)
- Electronic Industries Association (EIA)
  - Electronic manufacturers
- International Telecommunications Union (ITU) [Formerly: Consultative Committee International Telegraph Telephone CCITT]
  - National PTT's
  - Scientific organizations
- IEEE
- Forums, e.g. Frame Relay and ATM
  - Vendors
  - Users



## Open Systems

- Standards lead to **"Open Systems"**
- With open systems customers are not locked into one vendor's solution
- Open systems lead to a *"seamless"* user environment, e.g., www

## Network architectures and the Reference Models

- Open systems are build upon a **Layered Architecture** of the network
- Layered Architecture is the *"structuring"* of network functions
- Note that a network protocols are one example of real-time distributed processing



## Network architectures and the Reference Models

- Reference models provide:
  - A conceptual framework to characterize networks
  - A mechanism to control/describe the complexity of networks
  - Required for open systems

## Network architectures and the Reference Models

- Layered Architectures must have
  - Structure
  - Symmetry
  - Peer protocols
- Structure is the collection of related processing functions into layers



## Network architectures and the Reference Models

- Symmetry requires compatible functions exist is source/destination systems
- Peer Protocols are the set of rules that govern the processing between peer entities, i.e., the source/destination

## Network architectures:
### Underlying Principles

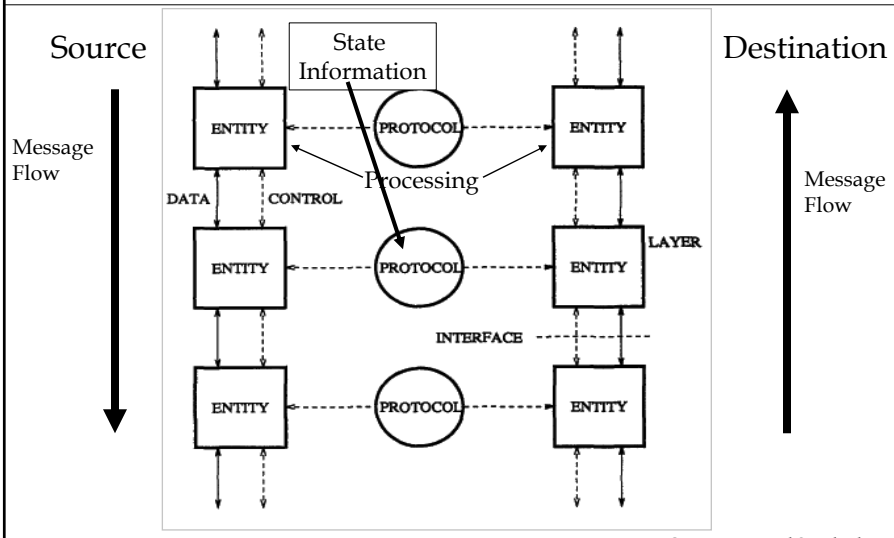
- Minimize the number of layers thus simplifying the tasks of describing and integrating different layers.
- Establish boundaries at points where the description of services is small and the number of interactions is minimum.
- Create layers that include different functions.



## Network architectures:
### Underlying Principles

- Establish boundaries where history demonstrates that the implementation can be partitioned.
- Engineer layers so that they can could be redesigned to take advantage of new technology without changing the services and interfaces of adjacent layers.
- Allow for the bypassing of sublayers.
- Each layer should add value

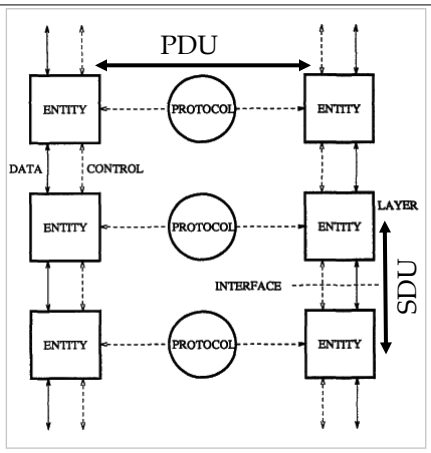
# Layered Architecture



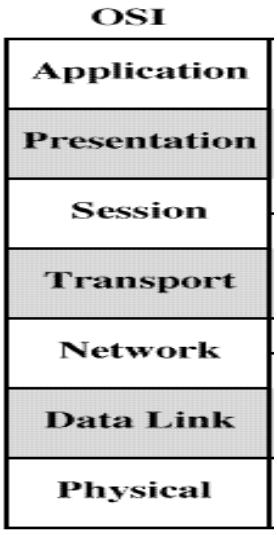



# PDUs and SDUs

- Protocol Data Units (PDU) = packets between Peer entities
- Service Data Units (SDU)= packets between layers



## Layered Architecture: International Organization for Standardization (ISO) Open Systems Interconnection Model (OSI)






## Physical Layer

- DTE/DCE interface
  - Data Terminal Equipment (PC)
  - Data Communications Equipment (Modem)
- Electrical/optics/radio connections
- Mechanical connections
- Functional Requirements
- Procedural protocol
- Bit transmission

## Data Link Layer

- Manage the link connection
- Supervise data interchange
- Synchronize and delimit
- Frame (block) sequencing
- Link flow control
- Link Error Control
- Abnormal condition recovery
- Identification and parameter exchange

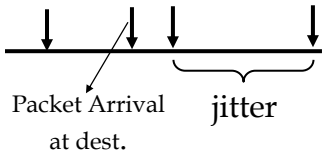


## Network Layer

- Routing and switching
- Network connections
- Logical channel control
- Segmenting and blocking
- Error recovery
- Sequencing and flow control

## Network Layer

- Guaranteed Delivery (eventually)
- Guaranteed Delivery with delay bound
- For packet flows if defined:
  - In-order delivery
  - Guaranteed minimal data rate (e.g. in b/s)
  - Guaranteed minimal jitter
  - Security






## Transport Layer

- Mapping
- Multiplexing
  - Multiple sessions on one transport pipe
- End-to-end error control
- Flow regulation
- Manage concatenated networks

## Session Layer

- Administrative services
  - Binding connections
  - Unbinding connections
- Dialog Services
  - Control data exchange
  - Interaction and synchronization
  - Exception reporting



## Presentation Layer

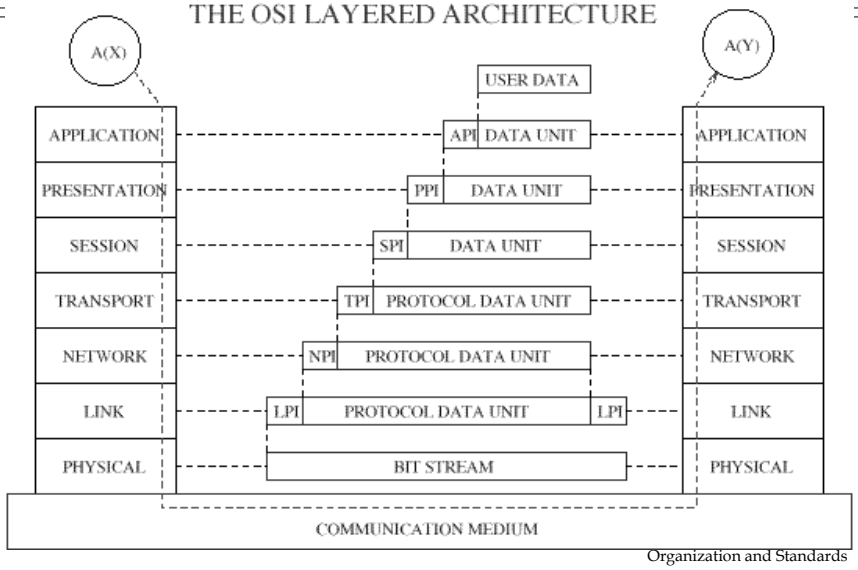
- Interpretation of data
- Data transformation
- Data formatting
- Syntax selection
- Structuring of data

# Application Layer

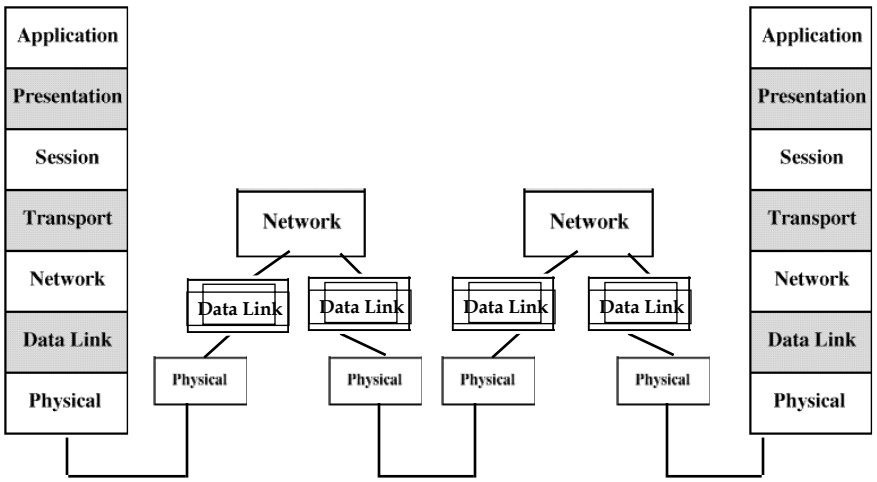
- Highest layer
- Serves as window to OSI
- Functions to provide all services
- Comprehensible to the user e.g.
  - Identification
  - Availability of resources
  - Authority
  - Authentication
  - Agreement on syntax
- Layer management function



# Layered Architecture: End-to-End Perspective



# Layered Architecture:

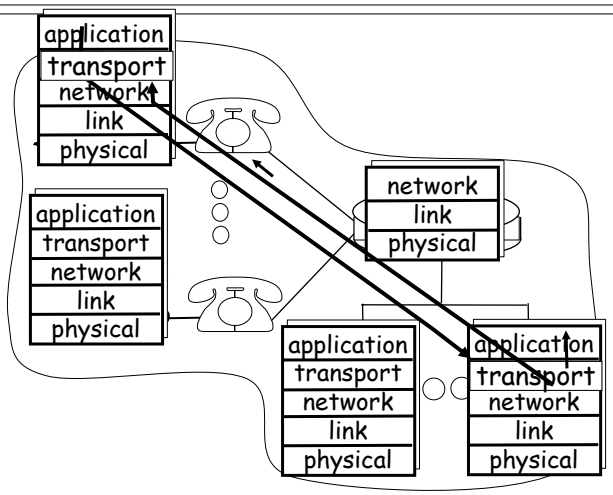





# Layering: *logical* communication

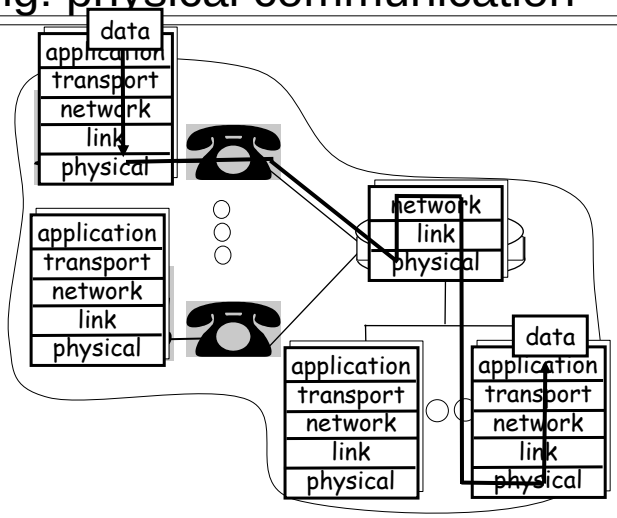
E.g.: transport

- 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




From: *Computer Networking: A Top Down Approach Featuring the Internet*, 2nd edition. Jim Kurose, Keith Ross Addison-Wesley, July 2002.

# Layering: physical communication



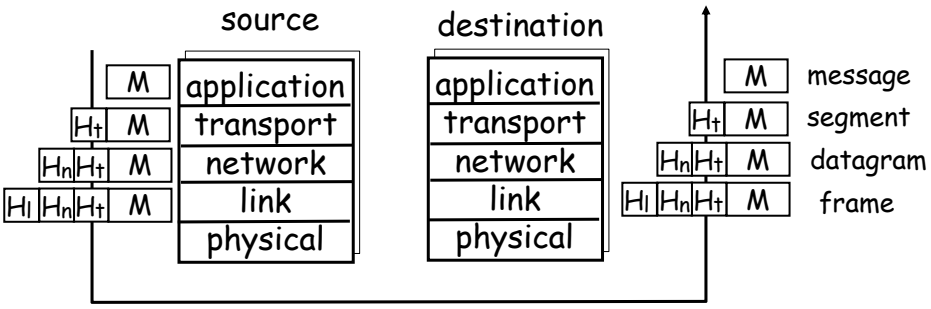

From: *Computer Networking: A Top Down Approach Featuring the Internet*, 2nd edition. Jim Kurose, Keith Ross Addison-Wesley, July 2002.



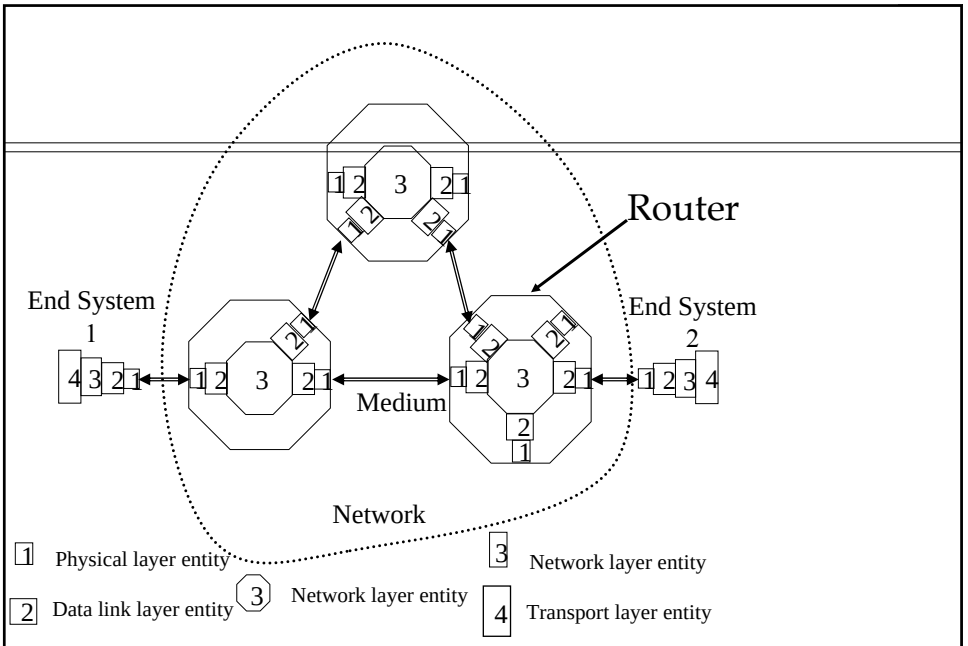
# Protocol layering and data

Each layer takes data from above

- adds header information to create new data unit
- passes new data unit to layer below




From: *Computer Networking: A Top Down Approach Featuring the Internet*, 2nd edition. Jim Kurose, Keith Ross Addison-Wesley, July 2002.




Modified from: Leon-Garcia & Widjaja: *Communication Networks*

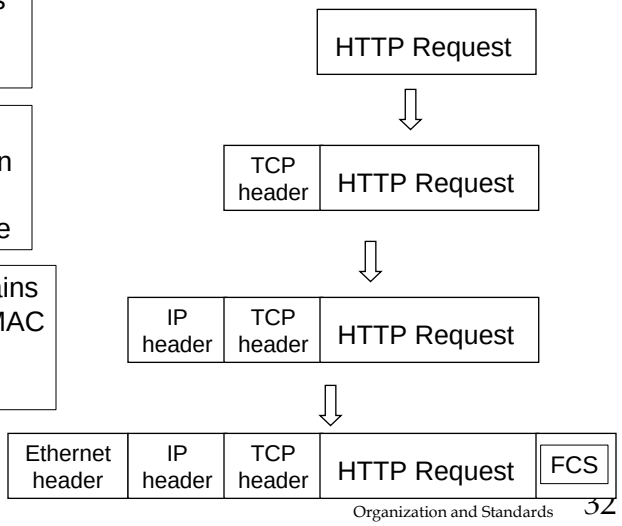


# Encapsulation

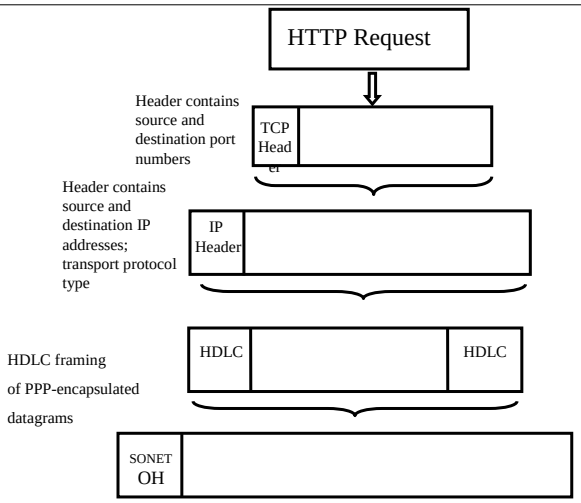
TCP Header contains source & destination port numbers

IP Header contains source and destination IP addresses; transport protocol type

Ethernet Header contains source & destination MAC addresses; network protocol type



## Example of Encapsulation: IP over SONET over WDM-Packet over SONET

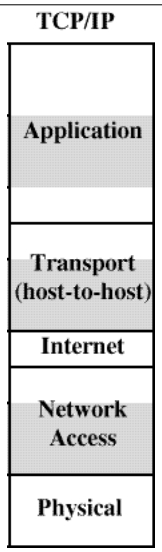





## Layered Architecture:

- Presentation: What does the peer look like?
- Sessions: Who is the Peer?
- Transport: Where is the Peer?
- Network: What is the route to the peer?
- Link: How is each step along the rout taken?
- Physical: How is the transmission medium used?

# Layered Architecture: TCP/IP






# Layered Architecture: TCP/IP

- Physical layer is same as in OSI
- Network Layer:
  - Interaction between end-systems and network
  - Source provides destination address through network layer
  - Makes higher layer software "independent" of underlying networking technology

## Layered Architecture: TCP/IP:

- Internet Layer
  - Routing between networks
  - Implemented in end systems
  - Implemented in routers
  - Internet Protocol (IP)



## Layered Architecture: TCP/IP:

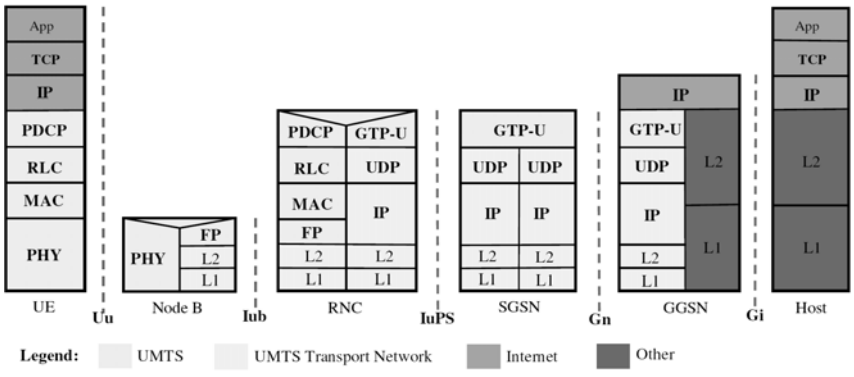
- Transport Layer
  - Reliable end-to end- transport
    - Transport Control Protocol (TCP)
  - User datagram protocol (UDP)
  - Others, e.g., Real Time Protocol (RTP)

# Layered Architecture: TCP/IP:

- Application Layer
  - ftp
  - telnet
  - Mail
  - www



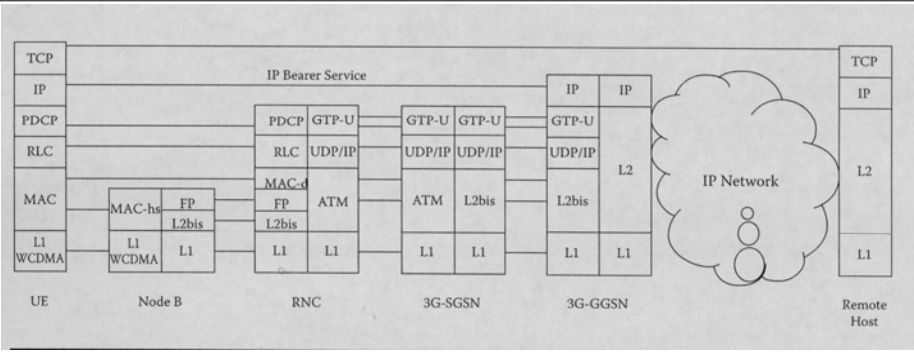
# Example Protocol Stack: Universal Mobile Telecommunications System (UMTS) Protocol Architecture - User Plane




FP= Framing Protocol GTP-U= GPRS Tunneling Protocol-User PDCP =Packet Data convergence Protocol

From: Geert Heijenk, wwwhome.cs.utwente.nl/~heijenk/mwn/slides/Lecture-5%206%20slides%20per%20page.pdf

# Example Protocol Stack: High Speed Data Packet Access(HSDPA)

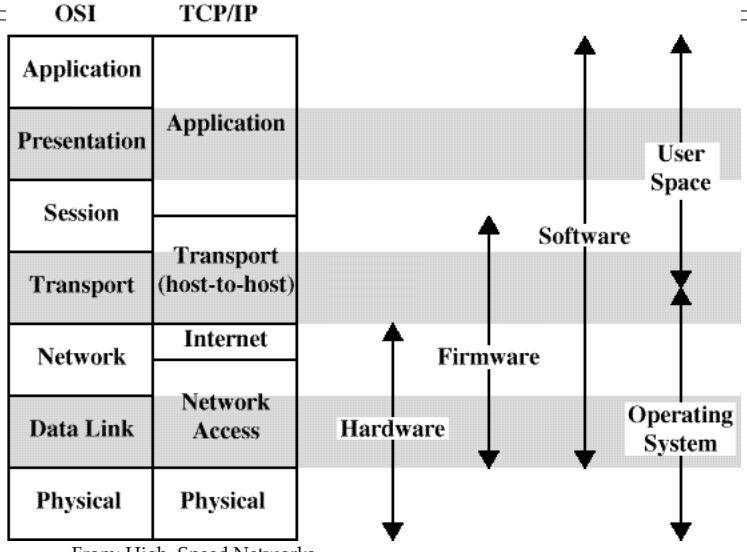



GTP-U = GPRS tunneling protocol: GTP-U is used for carrying user data within the GPRS core network
and between the Radio Access Network and the core network.

From: Mohamad Assaad, Zeghlache Djamal TCP Performance Over UMTS-HSDPA Systems, CRC Press, 2007



# Layered Architecture: OSI and TCP/IP




From: High -Speed Networks, W. Stallings,Prentice Hall, 1998