Computer Networking

Mooc 1: Fundamentals of Network Communication

Digital Communication:

In digital transmission, long-distance digital communications require the use of a generator to recover original data sequence and re-transmits on next segment   
(Trong dẫn truyền kĩ thuật số, truyền thông kĩ thuật số trong khoảng cách xa yêu cầu sử dụng một trình tạo để khôi phục chỗi dữ liệu gốc và truyền lại trong đoạn tiếp theo.)

Digital communication technologies may introduce errors in communication, which of the following can be used to provide reliable communication: TCD & HDLC   
(Công nghệ truyền thông kĩ thuật số có thể gây ra lỗi trong truyền thông)

Commonly used as Digital Communication medium: Coaxial cable, Optical fibel, Twisted pair

Network Architecture Evolution

Telegraph networks 🡪 Telephone networks 🡪 Internet, Optical & Wireless networks 🡪 Next Generation Internet

1. Telegraph networks
   * Electric Telegraph Networks: Message switching & Store-and-Forward operation
   * Elements of Telegraph Networks:
     1. Digital Transmission:
        1. Text messages converted into symbols
        2. Transmission system designed to convey symbols
           1. Receives energy from medium
           2. Converts received signal into a form suitable for delivery to user
     2. Multiplexing:
        1. *Framing* needed to recover text characters
     3. Message Switching:
        1. Messages contain source & destination *addresses*
        2. *Store-and-Forward: m*essages forwarded hop-by-hop across network
        3. *Routing* according to destination address
2. Telephone networks:
   * Circuit Switching and connection oriented
3. Internet, Optical & Wireless networks (Computer Networks and the Internet)
   * Packet switching
   * Virtual circuit switching
4. Next Generation Internet: ??

Packet switching & Message switching: Supporting multiple applications

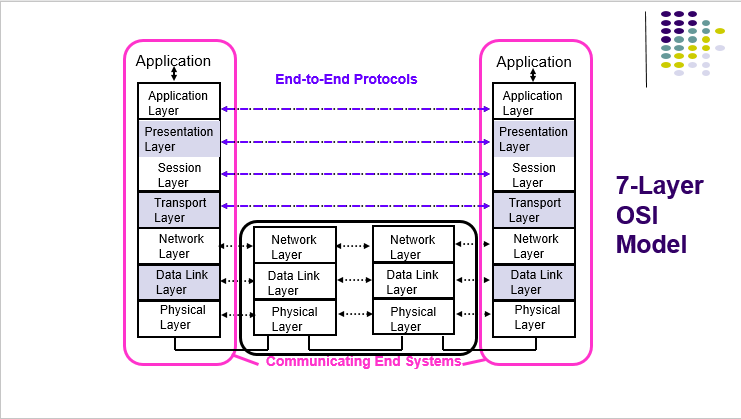
Computer Network Evolution

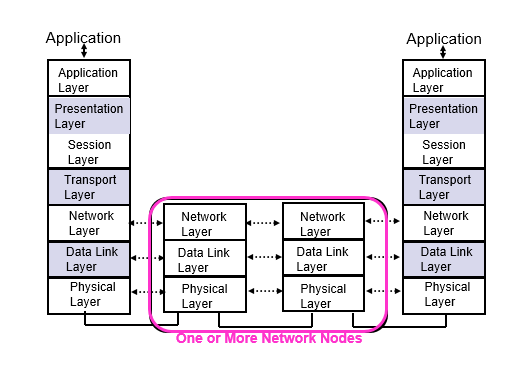
* 1960s: Terminals access shared host computer
* 1970s: Computers connect directly to each other
  + ARPANET packet switching network
  + TCP/IP Internet protocols
  + Ethernet local area network
* 1980s - 2000s: New applications and Internet growth

Services & Applications

1. Layers, Services & Protocols
   * The overall communications process between machines connected across one or more networks is very complex ( Quá trình giao tiếp tổng thể giữa các máy được kết nối thông qua 1 hoặc nhiều mạng rất phức tạp)
   * Layering partitions related communications functions into groups that are manageable (Sắp xếp phân vùng chức năng truyền thông liên quan đến nhau thành một nhóm để dễ dàng quản lý)
   * Each layer provides a service to the layer above (Mỗi layer cung cấp 1 dịch dụ cho layer ở bên trên)
   * Each layer operates according to a protocol (Mỗi layer được thực thi dựa trên 1 giao thức)
2. DNS (Domain Name System)
   * network device is primarily used to translate a domain name to the associated IP address
   * If it does not have an entry for a requested URL, a DNS server take the server checks with another DNS server to see if it has an entry.
   * DNS is a domain-name-service that responds to queries of domain name to IP address or IP address to domain name. DNS uses services provided by UDP
   * DNS protocol is an application layer protocol
   * DNS is a distributed database that resides in multiple machines in the Internet
   * DNS protocol allows queries of different types
   * DNS usually involves short messages and so uses service provided by UDP
   * Well-known port 53
   * DNS servers are one primary target of cyber attacks
3. TCP (Transmission Control Protocol)
   * HTTP needs a Transmission Control Protocol (TCP) connection between the HTTP client and HTTP server to transfer messages reliably
4. HTTP
   * HTTP built upon from TCP
   * Is an application layer protocol
   * Retrieves documents on behalf of a browser application program
   * HTTP specifies fields in request messages and response messages
     1. Request types; Response codes
     2. Content type, options, cookies, …
   * HTTP specifies actions to be taken upon receipt of certain messages
5. Protocol
   * A protocol is a set of precise & unambiguous rules that governs ( Giao thức là một tập hợp các quy tắc chính xác và rõ ràng để quản lý)
     1. how two or more communicating entities in a layer are to interact
     2. Messages that can be sent and received
     3. Actions that are to be taken when a certain event occurs
   * **The purpose of a protocol is to provide a service to the layer above**

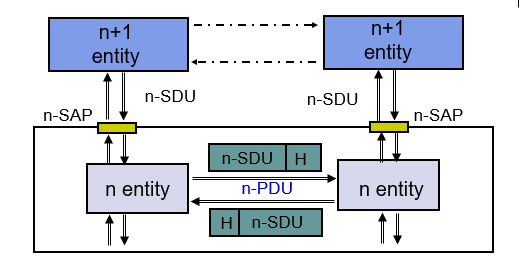
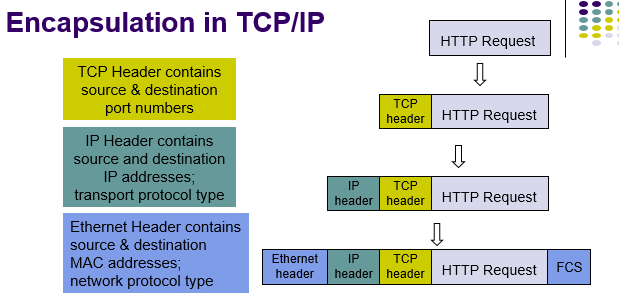
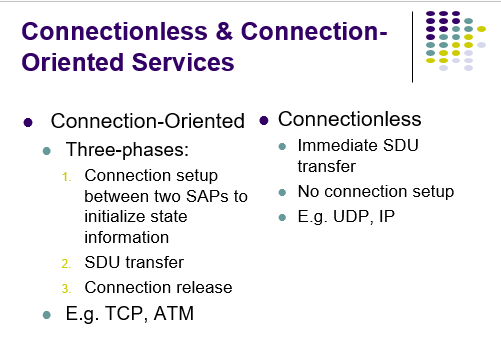
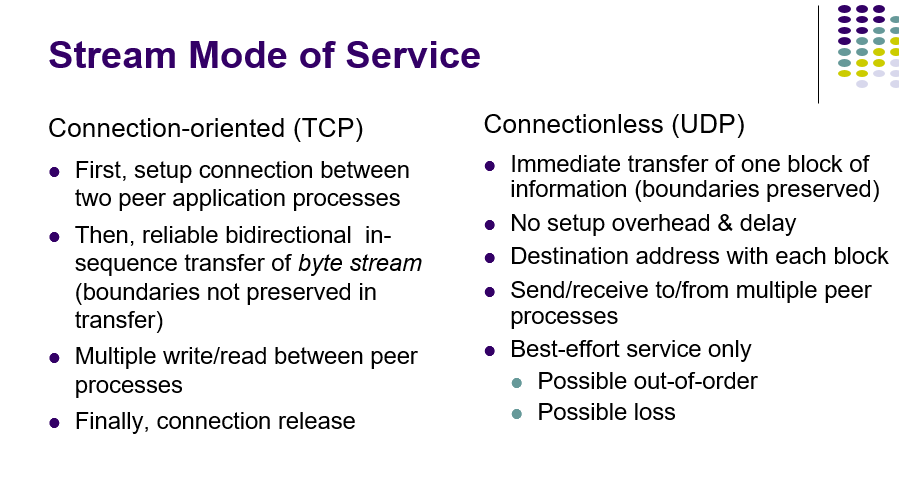
Layer OSI Model





1. Physical Layer
   1. Transfers bits across a link
   2. Definition & specification of the physical aspects
      1. Mechanical: cable, plugs, pins...
      2. Electrical/optical: modulation, signal strength, voltage levels, bit times, …
      3. functional/procedural: how to activate, maintain, and deactivate physical links…
   3. Ethernet, DSL, cable modem, telephone modems…
   4. Twisted-pair cable, coaxial cable optical fiber, radio, …
2. Data Link Layer
   1. Transfers *frames* across *direct* connections
      1. Groups bits into frames
      2. Detection of bit errors; Retransmission of frames
   2. Activation, maintenance of data link connections
   3. Medium access control for local area networks
   4. *Node-to-node* flow control
   5. Data Link Layer responsible for dividing the transmitted bit stream into frames
   6. The Medium is the physical path over which a message travels
   7. Data link layer is concerned with framing and the transport layer is not
      1. Insert framing information into the transmitted stream to indicate the boundaries that define frames
      2. Provide error control to ensure reliable transmission
      3. Provide flow control to prevent the transmitter from overrunning the receiver buffer
3. Network Layer
   1. Transfers *packets* across multiple links and/or multiple networks
      1. *Addressing* must scale to large networks
      2. Nodes execute *routing* algorithm to determine paths across the network
      3. Routing protocol means the procedure used to select routing paths
      4. *Forwarding* transfers packet across a node
      5. *Congestion control* to deal with traffic surges
      6. Most complex layer in the OSI reference model
   2. Network Layer responsible for determining which route through the network to use
   3. In internet, switching is done by using datagram approach to packet switching at the Network Layer
   4. Packet Network
      1. Individual packet streams are highly bursty
      2. User demand can undergo dramatic change
      3. Internet structure is highly decentralized
   5. It is possible for a network layer to provide a choice of services to the user of the network. the IP network layer offers Best-effort connectionless service
4. Internetworking
   1. Internetworking is part of network layer and provides transfer of packets across multiple and possibly dissimilar networks
   2. Gateways (routers) direct packets across networks
5. Transport Layer
   1. Transfers segments end-to-end from process in a machine to process in another machine
      1. *Reliable* stream transfer or quick-and-siBmple single-block transfer
      2. Message segmentation and reassembly
      3. Connection setup, maintenance, and release
   2. Transport Layer responsible for providing end-to-end communication with reliable service
   3. The data link layer and transport layer have in common Flow control feature
6. Application & Upper Layer
   1. Application Layer: Provides services that are frequently required by applications: DNS, HTTP web access, file transfer, email…
   2. Incorporates into Application Layer

OSI Unified View

* Layer
  1. A layer is a set of related communication functions managed and grouped together
  2. Layer n in one machine interacts with layer n in another machine to provide a service to its upper layer n +1
  3. The entities comprising the corresponding layers on different machines are called peer processes. (các thực thể bao gồm các lớp tương ứng trên các máy khác nhau được gọi là peer process(quy trình ngang hàng) )
  4. The processes at layer n are referred to as layer n entities.
  5. The IP Layer provides an unreliable, connectionless delivery system
* Protocol
  1. The machines at the same layer use a set of precise and unambiguous rules called the *layer-n protocol*.
  2. Layer-*n* peer processes communicate by exchanging *Protocol Data Units* (PDUs)
* Services
  1. Communication between peer processes is virtual and actually indirect
  2. Layer *n+1* transfers information by invoking the services provided by layer *n*
  3. Services are available at *Service Access Points (*SAP’s)
  4. Each layer passes data & control information to the layer below it until the physical layer is reached and transfer occurs
  5. The data passed to the layer below is called a *Service Data Unit* (SDU); SDU’s are *encapsulated* in PDU’s
* Layer, Services & Protocols
  1. 
* Bandwidth Utilization
  1. Utilization =
* Encapsulation in TCP/IP
  1. 
  2. The ways in which the OSI reference model and TCP/IP reference model differ.
     1. They differ in the number of layers
     2. TCP/IP model does not have presentation layer, but OSI model has
     3. TCP/IP model does not have session layer, but OSI model has
* Connectionless & Connection Oriented Services
  1. 
  2. 
* Internet Protocol (IP)
  1. IP packets transfer information across Internet
  2. *Host A IP → router→ router…→ router→ Host B IP*
  3. *Routers (gateways)* interconnect different networks
  4. Host computers prepare IP packets and transmit them over their attached network
  5. Routers forward IP packets across networks
  6. *Best-effort* IP transfer service
  7. IP glues the network of networks together as the Internet
* Transport Protocols
  1. Host computers run two transport protocols on top of IP to enable process-to-process communications
  2. *User Datagram Protocol* (UDP) enables best-effort connectionless transfer of individual block of information
  3. *Transmission Control Protocol* (TCP) enables connection-oriented reliable transfer of a stream of bytes. Web browsing and File transfer is applications would you select TCP protocol for.
  4. Two services though Sockets: connection-oriented and connection-less
* Client & Server Differences
  1. Server
     1. Specifies well-known port # when creating socket
     2. May have multiple IP addresses (net interfaces)
     3. Waits passively for client requests
  2. Client
     1. Assigned ephemeral port #
     2. Initiates communications with server
     3. Needs to know server’s IP address & port #
        1. DNS for URL & server well-known port #
     4. Server learns client’s address & port #
* Stream mode of Service
  1. Connectionless (UDP)
     1. Immediate transfer of one block of information (boundaries preserved)
     2. No setup overhead & delay
     3. Destination address with each block
     4. Send/receive to/from multiple peer processes
     5. Best-effort service only
        1. Possible out-of-order
        2. Possible loss
* In a LAN, Physical address is used to transfer frames to appropriate destination

BSD Sockets API

* SOCK\_STREAM(TCP): used to create a TCP socket
* SOCK\_DGRAM(UDP): used to create a UDP socket
* TCP socket is stream oriented
* UDP socket is block oriented
* System call:
  + Connect(): Establishes connection between the endpoints on stream sockets.
  + Bind(): used to assign a network address to the socket
  + Listen(): Sets the socket in a listen mode.
  + Gethostbyname(): if a client knows the server name but not server's network address, the client use to get server's network address
  + Sendto(): usually used for transmitting data in the connectionless mode

C1-M3-L3

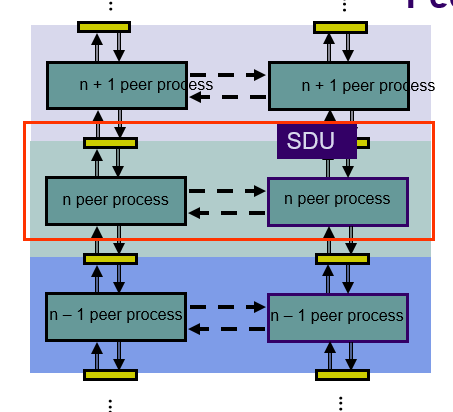
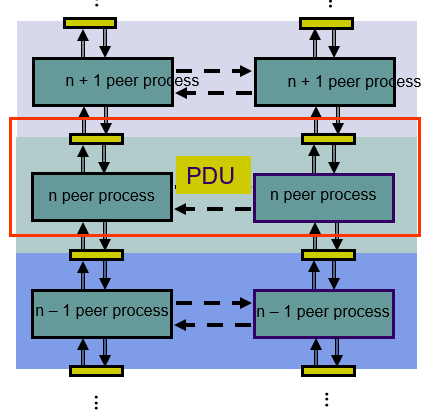
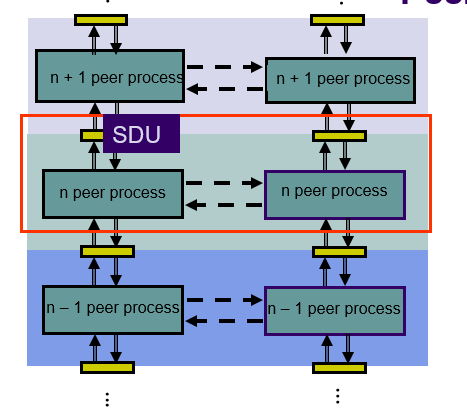
C1-M4-L1

C1-M4-L2

C1-M4-L3

MOOC 2: Peer-to-Peer Protocols and Local Area Networks

Peer-to-Peer Protocols



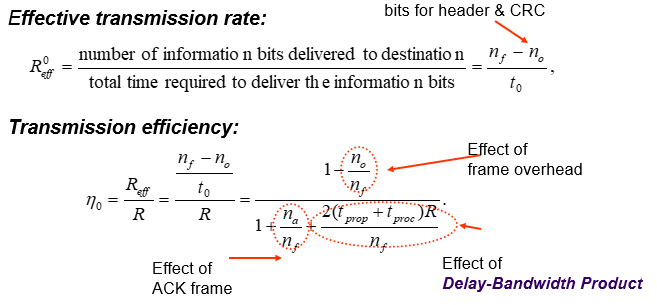
* *Peer-to-Peer processes* execute layer-n protocol to provide service to layer-(n+1)
* Layer-(n+1) peer calls layer-n and passes Service Data Units (SDUs) for transfer
* Layer-n peers exchange Protocol Data Units (PDUs) to effect transfer
* Layer-n delivers SDUs to destination layer-(n+1) peer

1. Service Models
   1. The *service model* specifies the information transfer service layer-n provides to layer-(n+1)
   2. Quality-of-Service (QoS) requirement that specifies a level of performance that can be expected in the transfer of information.
   3. Example of Services: Service model posible features:
      1. Arbitrary message size or structure: kích thước và cấu trúc của tin nhắn tùy ý
         1. Different services impose restrictions on size & structure of data it will transfer Single bit? Block of bytes? Byte stream?
      2. Sequencing & Reliability: Độ tin cậy, sự sắp xếp
         1. *ARQ protocols* = error detection + retransmission + sequence numbering 🡪 provide reliability
         2. Provide reliable communication: TCP, HDLC
      3. Timing: Thời gian (căn thời gian)
      4. Flow control: kiểm soát lưu lượng
         1. Messages can be lost if receiving system does not have sufficient(đủ) buffering(bộ nhớ đệm) to store arriving messages
         2. If destination layer-(n+1) does not retrieve its information fast enough, destination layer-n buffers may overflow
         3. *Flow Control* provide backpressure mechanisms that control transfer according to availability of buffers at the destination
         4. Examples: TCP and HDLC
      5. Multiplexing: ghép kênh
         1. Applications involving voice and video generate units of information that are related temporally(liên quan đến thời gian): Ứng dụng liên quan đến giọng nói và video tạo ra các đơn vị thông tin có liên quan theo thời gian.
         2. Destination application must reconstruct temporal relation in voice/video units: Ứng dụng đích phải tái tạo lại mối quan hệ thời gian trong các đơn vị giọng nói hoặc video
         3. Network transfer introduces delay & jitter(độ trễ & nhiễu, bất ổn)
         4. Timing Recovery protocols use timestamps(tem thời gian) & sequence numbering(đánh số trình tự ) to control the delay & jitter in delivered information
         5. Examples: RTP & associated protocols in Voice over IP
      6. Privacy, integrity, and authentication: bảo mật, toàn vẹn và xác thực
         1. *Privacy*: ensuring that information transferred cannot be read by others
         2. *Integrity*: ensuring that information is not altered(thay đổi, biến đổi) during transfer
         3. *Authentication*: verifying(xác minh, xác định) that sender and/or receiver are who they claim to be: xác minh người gửi hoặc người nhận là người mà họ đã claim
         4. Examples: IPSec, SSL
   4. End-to-End vs Hop-by-Hop
      1. A service feature can be provided by implementing a protocol
         1. end-to-end across the network
         2. across every hop in the network
      2. Examples:
         1. Perform error control at every hop in the network or only between the source and destination?
         2. Perform flow control between every hop in the network or only between source & destination?
      3. Hop-by-Hop: khởi động phục hồi lỗi nhanh hơn( faster recovery), có thể cung cấp dịch vụ đáng tin cậy hơn. Lỗi có khả năng xảy ra(likely), complex in node 🡪 hop by hop
      4. End-to-End: simple inside network, ưa chuộng hơn, lỗi không thường xuyên( infrequent) 🡪 end-to-end
   5. Error control in Data Link Layer
      1. Data Link operates over *wire-like*, directly-connected systems
      2. Frames can be corrupted or lost, but arrive in order
      3. Data link performs error-checking & retransmission
      4. Ensures error-free packet transfer between two systems

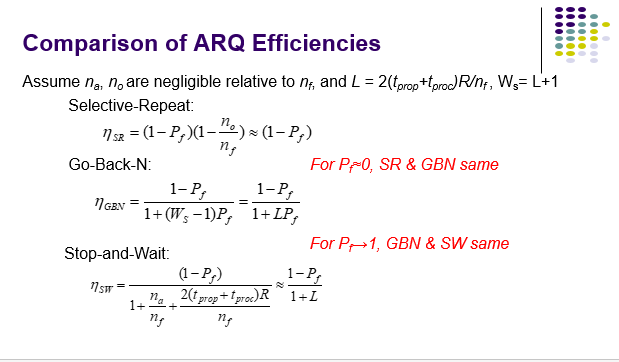
Automatic Repeat Request (ARQ): Yêu cầu lặp lại tự động

* *Purpose* **:** to ensure a sequence of information packets is delivered in order and without errors or duplications despite transmission errors & losses **(để đảm bảo rằng các gói thông tin trên quá trình truyền, đúng thứ tự, không xảy ra lỗi hoặc lặp mặc dù có thể xảy ra lỗi đường truyền hoặc mất đường truyền)**
* Sliding window: a set of Seq.# corresponding to frames permitted to send or receive. (**một tập hợp các Seq tương ứng với các khung được phép gửi hoặc nhận**)

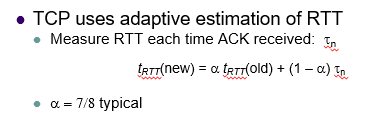
1. Three ARQ protocol:
   1. Stop-and-Wait ARQ
   2. Go-Back N ARQ
   3. Selective Repeat ARQ
2. Basic elements of ARQ:
   1. *Error-detecting code* with high error coverage: mã phát hiện lỗi với phạm vi tìm thấy lỗi cao)
   2. *ACKs* (positive acknowledgments): thừa nhận tích cực
   3. *NAKs* **(**negative acknowledgments): thừa nhận tiêu cực
   4. *Timeout mechanism: Cơ chế thời gian chờ*

* Stop-and-Wait ARQ
  + All ARQ protocols require the frame to contain a header which can store information that is needed for it's proper operation.
  + Stop-and-wait is only efficient if the link delay-bandwidth product is small
  + In Stop-and-Wait protocol, sequence number are required
  + Delay-bandwidth product is a key element in performance evaluation of network protocols
  + Disadvantage of Stop-and-Wait protocol:
    - Error free communication channel does not exist
    - Acknowledgement may get lost
    - Deadlock situation may occur
  + **Applications of Stop-and-Wait ARQ**
    - IBM Binary Synchronous Communications protocol (Bisync): character-oriented data link control
    - Xmodem: modem file transfer protocol
    - Trivial File Transfer Protocol (RFC 1350): simple protocol for file transfer over UDP
  + **S&W Efficiency on Error-free channel**
    - 
* Go-Back-N ARQ
  + Improve Stop-and-Wait by not waiting!
  + Keep channel busy by continuing to send frames
  + A procedure where the transmission of a new frame is begun before the completion time of the previous frame transmission is said to be **pipelining**.
  + Allow a window of up to Ws outstanding frames
    - Receiver’s window size is often 1
  + The window size must be larger than the delay-bandwidth product to ensure that the channel is kept full
  + If ACK for oldest frame arrives before window is exhausted, continue transmitting
  + If window is exhausted, pull back and retransmit all outstanding frames

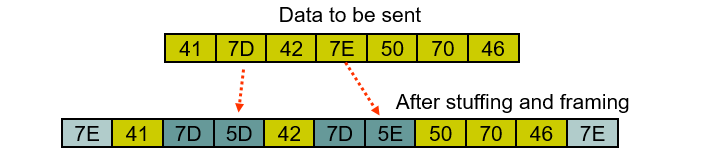
Piggybacking: Trong quá trình truyền tải dữ liệu 2 chiều, các data frame và xác nhận của các frame xen kẽ nhau. Nhưng Piggybacking cho phép người nhận có thể chèn các ACK vào the next departing frame

* Quest:
  + Consider a situation where an interactive application produces a packet to send each keystroke from the client and the server echoes each keystroke that it receives from the client. Which of following strategies for sending ACK frames in a Go-Back-N is appropriate for the situation? send an ACK frame when the next piggyback opportunity arises
  + Consider a bulk data transfer application where a server sends a large file that is segmented in a number of full-size packets that are to be transferred to the client. Assume the channel has a low probability of error. Which of following strategies for sending ACK frames in a Go-Back-N is appropriate for the situation? send an ACK frame after every other frame is received
  + If the probability of error is very low in a communication link, Go-back-N ARQ and Selective Repeat ARQ protocols have similar performance
* Selective Repeat ARQ
  + because *multiple* frames are resent when errors or losses occur (GBN). Correct but out-of-sequence frames would be discarded (because receiver buffer only 1)
  + Selective Repeat retransmits *only an individual frame*
    - Timeout causes individual corresponding frame to be resent
    - NAK causes retransmission of oldest un-acked frame
  + Receiver maintains a *receive window* of sequence numbers that can be accepted
    - Receiver window is increased larger
    - Error-free, but out-of-sequence frames with sequence numbers within the receive window are buffered
    - Arrival of frame with Rnext causes window to slide forward by 1 *or more*
  + **Applications of Sel. Repeat ARQ**
    - *TCP* (Transmission Control Protocol): transport layer protocol uses variation of selective repeat to provide reliable stream service
    - *Service Specific Connection Oriented Protocol*: error control for signaling messages in ATM networks
* Comparison of ARQ Efficiencies:
  + 
* TCP ARQ Model
  + TCP reliable stream service
    - Connection-oriented
    - Error free, without duplication, in order of sequence
  + TCP uses *Selective Repeat ARQ*
    - Transfers byte stream without preserving boundaries
* Application layer write bytes into send buffer through socket (can fragment or reassemble)
* TCP ARQ Environment
  + Operates over best effort service of IP that is not wirelike
    - Packets can arrive with errors or be lost
    - Packets can arrive out-of-order
    - Packets can arrive after very long delays
  + Old segments from previous connections may arrive, so detection and elimination of duplicates is hard
* TCP ARQ Sequence #
  + Sequence Numbers
    - Seq. # is number of first byte in segment payload
    - Very long Seq. #s (32 bits) to deal with long delays
    - Initial sequence numbers negotiated during connection setup (to deal with very old duplicates) ( Thiết lập số thứ tự ban đầu trong khi thiết lập kết nối)
    - Accept segments within a receive window 🡪 likelihood accepting a very old message is very low ( Bất cứ lúc nào cũng chấp nhận segments number từ của nhỏ rất nhiều 🡪 nên khả năng nhận 1 thông điệp rất cũ là rất thấp)
    - Timeout at the end of connection to clear old segments
  + TCP Connection
    - Use 3 way handshake ( bắt tay 3 chiều)
      * Identified uniquely by Send IP Address, Send TCP Port #, Receive IP Address, Receive TCP Port #
      * Data transfer: exchange segment carring data
      * Graceful close: Close each direction separatly

|  |  |
| --- | --- |
| * + Transmitter | * + Receiver |
| * + Ready state   + Wait request from above layer   + Update Slast & CRC   + Go to wait state   + Wait state   + Wait for ACK or time to expire block request from higher layer   + If timeout expires 🡪 retransmit frame, rest timer   + If ACK received   + Sequence number incorrect/ error   + Deleted: Ignose ACK   + Sequenced: Correct (Rnext = Slast + 1)   + Accept frame, go to Ready State | * + Always in Ready State   + Wait for arrive of new frame   + No error & Sequence number correct( Slast = Rnext)   + +Accept frame, update Rnext   + +Send ACK frame with Rnext   + +Deliver packet higher layer   + No error, squence number incorrect   + + discard frame   + + snde ACK frame with Rnext   + Error detected   + Discard frame |

* + TCP flow control
    - Enssure sender close not too much data that may overwhelm the receiver
    - WA = WR  - (Rnew – Rlast) 🡪 Receiver
    - Srecent – Slast <= WA
  + TCP retransmission Time out
    - TCP retransmits a segment after timeout period
      * Timeout too short: excessive number of retransmissions (số lượng truyền lại quá nhiều)
      * Timeout too long: recovery too slow
      * Timeout depends on RTT: time from when segment is sent to when ACK is received
  + Adaptive RTT
    - 

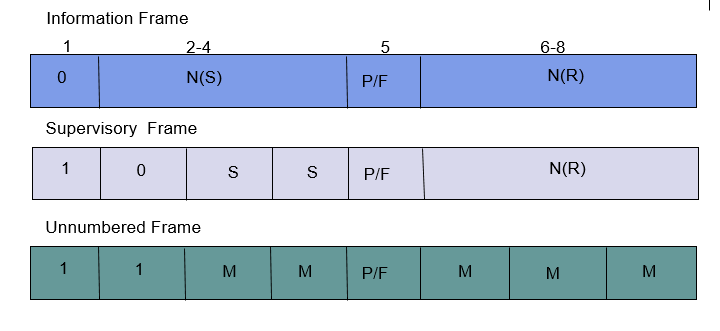
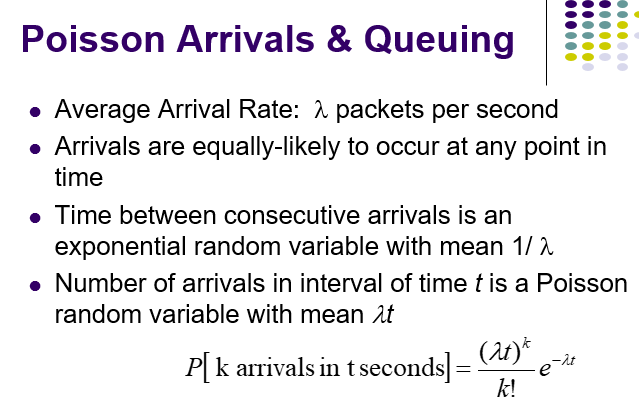
Data Link Protocols:

* **Data Links Services**
  + - Framing
    - Error control
    - Flow control
    - Multiplexing
    - Link Maintenance
    - Security: Authentication & Encryption
* Example
  + - PPP
    - HDLC
    - Ethernet LAN
    - IEEE 802.11 (Wi Fi) LAN
* Framing
  + Mapping stream of physical layer bits into frames
  + Mapping frames into bit stream
  + Frame boundaries can be determined using:
    - Character Counts
    - Control Characters
    - Flags
    - CRC Checks
* Character-Oriented Framing (Byte Stuffing = nhồi byte)
  + Frames consist of integer number of bytes
  + Special 8-bit patterns used as control characters
  + Byte used to carry non-printable characters in frame
    - DLE (data link escape) = 0x10,là dữ liệu liên kết thoát khỏi điều khiển với giá trị hệ thập lục phân một số 0 được giới thiệu
    - DLE STX (DLE ETX) used to indicate beginning (end) of frame, STX được sử dụng để chỉ sự bắt đầu của 1 frame, ETX biểu thị sự kết thức của 1 khung
    - Insert extra DLE in front of occurrence of DLE STX (DLE ETX) in frame
    - All DLEs occur in pairs except at frame boundaries
  + Flag-base Framing & Bit Stuffing
    - Bit Stuffing
      * Frame delineated by flag character
      * It uses *bit stuffing* to prevent occurrence of flag 01111110 (HEX 7E) inside the frame
      * Transmitter inserts extra 0 after each consecutive five 1s *inside* the frame
      * Receiver checks for 5 consecutive 1s
        + if next bit = 0, it is removed
        + if next two bits are 10, then flag is detected
        + If next two bits are 11, then frame has errors
* PPP: Point-to-Point Protocol
  + Data link protocol for point-to-point lines in Internet
    - Router-router; dial-up to router
  + 1. Provides *Framing and Error Detection*
    - Character-oriented HDLC-like frame structure
  + 2. *Link Control Protocol*
    - Bringing up, testing, bringing down lines; negotiating options
    - ***Authentication***: key capability in ISP access
  + 3. A family of *Network Control Protocols* specific to different network layer protocols
    - IP, OSI network layer, IPX (Novell), Appletalk
* PPP Frame
  + PPP uses similar frame structure as HDLC, except
    - Protocol type field
    - Payload contains an *integer* number of bytes
  + PPP uses the same flag, but uses *byte stuffing*
  + Problems with PPP byte stuffing
    - Size of frame varies unpredictably due to byte insertion ( kích thước của frame không đoán được do chèn byte)
    - Malicious users can inflate bandwidth by inserting 7D & 7E (người dùng độc hại có thể thổi phồng băng thông bằng cách đảo ngược 7D & 7E)
* Byte – Stuffing in PPP
  + Is character-oriented version of HDLC
  + Flag is 0x7E (01111110)
  + Control escape 0x7D (01111101)
  + Any occurrence of flag or control escape inside of frame is replaced with 0x7D followed by original octet XORed with 0x20 (00100000)
  + 
* PPP Applications
  + PPP used in many point-to-point applications
  + Telephone Modem Links 30 kbps
  + Packet over SONET 600 Mbps to 10 Gbps
  + IP→PPP→SONET
  + shared links such as Ethernet to provide LCP, NCP, and authentication features
* PPP Authentication
  + Password Authentication Protocol
    - Initiator must send ID & password
    - Authenticator replies with authentication success/fail
    - After several attempts, LCP closes link
    - Password, ID are transmitted unencrypted(không được mã hóa), susceptible to eavesdropping( nghe trộm)
  + Challenge-Handshake Authentication Protocol (CHAP)
    - Initiator & authenticator share a secret key
    - Authenticator sends a challenge (random # & ID)
    - Initiator computes cryptographic checksum of random # & ID using the shared secret key
    - Authenticator calculates cryptocgraphic checksum & compares to response
    - Authenticator can reissue challenge during session

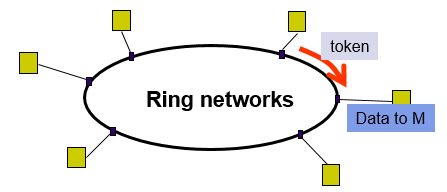
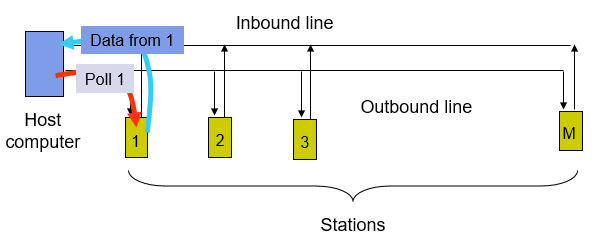
High-Level Data Link Control (HDLC)

* Bit-oriented data link control 🡪 Khác vs PPP: Character-oriented
* Derived from IBM Synchronous Data Link Control (SDLC)
* Related to Link Access Procedure Balanced (LAPB)
  + LAPD in ISDN
  + LAPM in cellular telephone signaling
* HDLC Data Transfer Modes
  + Normal Response Mode (NRM)
    - Used in polling multi-drop lines
  + Asynchronous Balanced Mode (ABM)
    - Used in full-duplex point-to-point links
* HDLC Frame Format

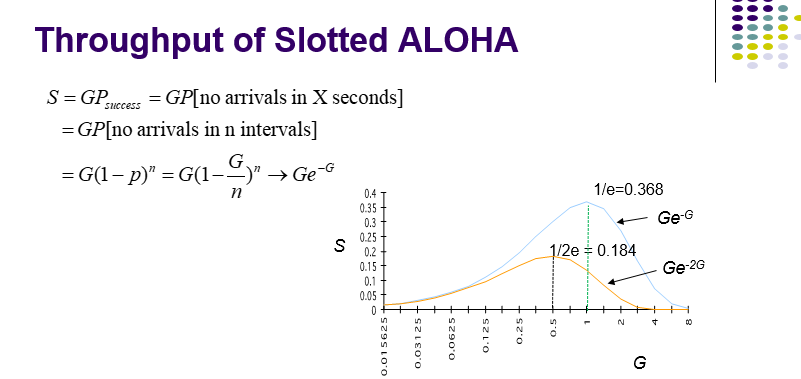
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Flag | Address | Control | Information | FCS | Flag |

* + Control: define purpose & functions of frame (1 or 2 octets)
  + Flag: delineate frame boundaries
  + Address: identify *secondary* station (1 or more octets)
    - In ABM mode, a station can act as primary or secondary 🡪 address can changes accordingly
  + Information: contains user data; length not standardized, but implementations impose maximum ( chứa dữ liệu người dùng, độ dài không được tiêu chuẩn hóa, nhưng được áp đặt tối đa)
  + Frame Check Sequence(FCS): 16- or 32-bit CRC
* Control Field Format
  + 
* **Error Detection & Loss Recovery**
  + Frames lost due to loss-of-synch or receiver buffer overflow ( Khung bị mất do không thể đồng bộ hoặc bộ nhớ đệm thu bị tràn)
  + Frames may undergo errors in transmission ( Frame có thể trải qua các lỗi trong khi truyền tải 🡪 Well)
  + CRCs detect errors and such frames treated as lost (CRCs phát hiện lỗi hoặc sửa chữa những frame bị mất)
  + Recovery through ACKs, timeouts & retransmission (Phục hồi thông qua ACKs, timeouts, retransmisstion)
  + Sequence numbering to identify out-of-sequence & duplicate frames
  + HDLC provides for options that implement several ARQ methods
* **Multiplexers inherent in Packet Switches**
  + Concentrates bursty traffic onto a shared line
  + Greater effiency and lower cost
  + Packet/frame forward to buffer( queue) prior to trans from switch
  + Multiplexers occur in buffer, use FIFO & priority scheduling
* **Delay and Utilization Tradeoff**
  + Buffering introduces packet delay
  + Buffer overflow introduces packet loss
  + End-to-end protocols deals with loss
* Multiplexer Modeling
  + Packet arrive pattern: What is the packet inter-arrival pattern?
  + Packet service time: How long are the packets?
  + Buffer discipline: If buffer is full, which packet is dropped?
  + Service Discipline: What is order of transmission?
  + Performance Measures:
    - *Delay Distribution;*
    - *Packet Loss Probability;*
    - *Line Utilization*
  + Delay = Waiting time( from arrrival start to begin of sercie) + Service Time( time to trans packet)
  + 🡪 Queueing involves sophisticated modeling and analysis multiplexing make god trade off of delay & utilization
  + 

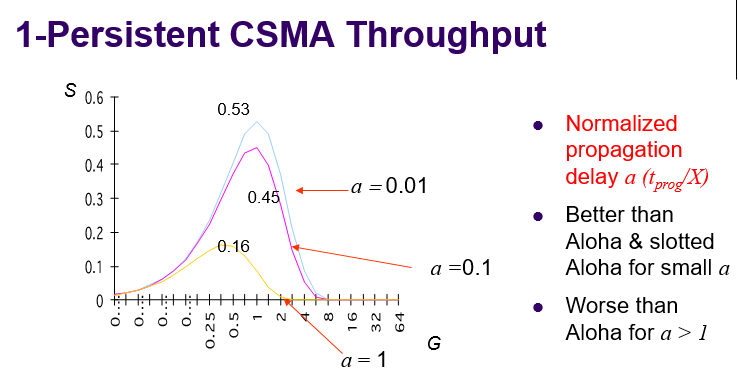
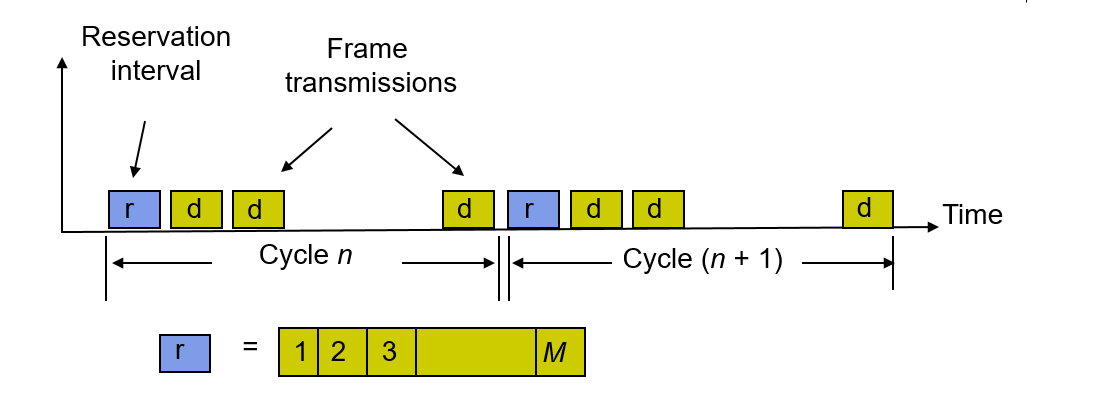
Medium Access Control

* Broadcast LANs
  + Simple and Cheap
  + All information sent to all users
  + No routing
* Primary function: minimize/ eliminate the instance of collision to achieve a reasonable utilization of medium ( giảm thiểu/ loại bỏ sự va chạm để đạt được sử dụng hợp lý)
* Media Sharing
  + Media Sharing techniques:
    - Static channelization: (Dành riêng cho người dùng cụ thể)
      * Partition medium: Phân vùng phương tiện
      * Dedicated allocation to users
      * Satellite transmission: Vệ tinh
      * Cellular Telephone
    - Dynamic medium access control
      * Scheduling:
        + Polling: take turns
        + Request for slot in transmission schedule
        + Token ring
        + Wireless LANs
      * Random access:
        + Loose coordination
        + Send, wait, retry if necessary
        + Aloha
        + Ethernet
    - Channelization
      * Advantage: suitable for staing generate strady stream ( để sử dụng hiệu quả kênh chuyên dụng (dedicate channer))
      * Disadvantage:
        + Inflexible in allocation of languager to users with diffent require
        + Inefficient for bursty trafffic
    - **Channelization Approaches**
      * *Frequency Division Multiple Access* (FDMA)
        + Frequency band allocated to users
        + Broadcast radio & TV, analog cellular phone
      * *Time Division Multiple Access* (TDMA)
        + Periodic time slots allocated to users
        + Telephone backbone, GSM digital cellular phone
      * *Code Division Multiple Access* (CDMA)
        + Code allocated to users
        + Cellular phones, 3G cellular
* Token – Passing
  + Ring: hold token trans 🡪 Passed to next station
    - 
* Polling
  + Use in multi job line to connect a number of terminal to a cantral computer
    - 
* Random access:
  + Transmit when ready 🡪 collision coccur 🡪 need retransmision
* Delay-Bandwidth Product:
  + A =
  + Tprop: propagation delay
  + L/R: Time to trans a frame
    - R: transmission bit rate( bandwidth)
    - L bits/frame

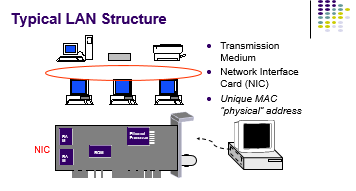
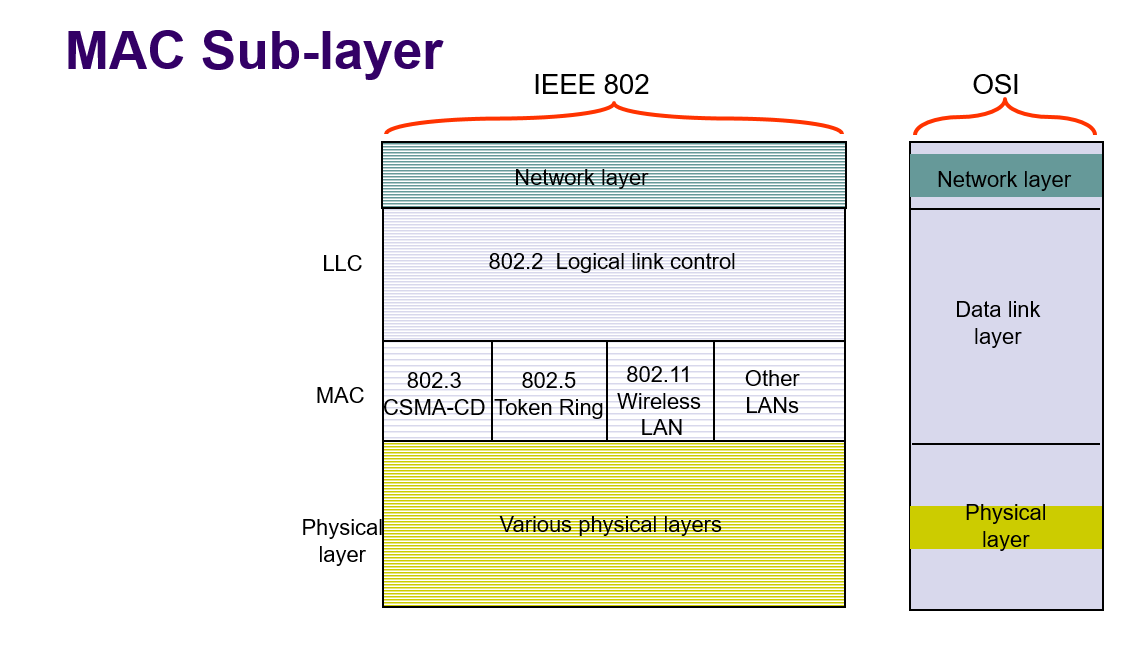
ALOHA

1. Wireless link to provide data transfer between campuses
   1. A station transmits whenever it has data to transmit
   2. If >1 frames are transmitted🡪 collide with each other and lost
   3. If ACK not received within timeout 🡪 a station picks random backoff
   4. Station retransmits frame after backoff time
   5. Max throughput = 18,4% / vulnerable period time = 2 tprop
2. Slotted ALOHA / vulnerable period time = 1 tprop
   1. ALOHA performance depends on probability of collisions
   2. Reducing vulnerable period 🡪reduce collision probability
   3. 🡪 by constraining the stations to transmit in synchronized manner. ( hạn chế các trạm truyền tải bằng cách đồng bộ)
   4. Max throughput: 36,8%
   5. 

Carrier Sensing Multiple Access(CSMA) (1 tprop)

1. A station senses the channel before it starts transmission
   1. If busy🡪 wait/ schedule backoff (different options)
   2. If idle 🡪 start transmission
2. CSMA options: 1-persistent CSMA
   1. 
3. Non-persisten CSMA (Least Gready)
   1. If busy 🡪 Wait a backoff period 🡪 sense carrier again
   2. High Delay & High efficiency
4. P-persistent CSMA (guecly)
   1. Wait till channel become idle, trans with probability p or wait one tpop & resense with probability(1-p)
   2. Delay = efficiency
5. CSMA/CD: CSMA with collision Detection
   1. Monitor for collision & about trans ( giám sát va chạm và hủy bỏ truyền)
   2. First: sensing 🡪 transmit
   3. Begin trans station: Continue listenint to the medium to detect collision
   4. If detected 🡪 about trans, reschedule random backoff time try again at scheduls time
   5. 🡪 quickly terminate a damaged frame help save time & bandwidth
   6. Contension resolution: if exactly 1 station trans in a slot
   7. Throughput: luana phieen (alternate): giữa contention period & frame trans time
6. Summary throughput for MAC Random Access
   1. Small a: CSMA-CD max throughput
   2. Large a: ALOHA & slotted ALOHA has max (a: normalizer propagation delay)
7. Scheduling for MAC
   1. Purpose: avoid collision in shared medium
   2. 🡪 More efficient channel utilization
      1. Less variability in delay
      2. Provide fairness to station. Increase computational or procedural amplexity (tăng độ phức tạp tính toán hoặc thủ tục)
8. Collision-free Reservation System
   1. Trans organized into cycle
   2. Cycle = a reservation interval + frame trans
   3. 
      1. Each slot for each station
   4. Option:
      1. Centralized: a central controller listen to reservation myor, decides order of trans, issue grant
      2. Distributed: Each station determines its slot for transmission from the reservation information
   5. Reservation System Options:
      1. Single or Multiple Frames
      2. Channelized or Random Access Reservations
         1. *Channelized (typically TDMA) reservation*: Reservation messages from different stations are multiplexed without any risk of collision
         2. *Random access reservation*: Each station transmits its reservation message randomly until the message goes through
   6. Polling system
      1. *Centralized systems*: A central controller accepts requests from stations and issues grants to transmit
      2. *Distributed systems*: Stations implement a decentralized algorithm to determine transmission order
      3. Options:
         1. Service Limits: How much is a station allowed to transmit per poll?
            1. *Exhaustive*: until station’s data buffer is empty (including new frame arrivals)
            2. *Gated*: all data in buffer when poll arrives
            3. *Frame-Limited*: one frame per poll
            4. *Time-Limited*: up to some maximum time
         2. Priority mechanisms
            1. More bandwidth and lower delay for stations that appear multiple times in the polling list
            2. Issue polls for stations with message of priority k or higher
   7. Comparison of MAC approaches
      1. Channelization
         1. Feasible if traffic is steady
      2. Aloha & Slotted Aloha
         1. Simple & quick transfer at very low load
         2. Accommodates large number of low-traffic bursty users
         3. Highly variable delay at moderate loads
         4. Efficiency does not depend on *a*
      3. CSMA & CSMA-CD
         1. Quick transfer and high efficiency for low delay-bandwidth product
         2. Can accommodate large number of bursty users
         3. Variable and unpredictable delay
      4. Reservation
         1. On-demand transmission of bursty or steady streams
         2. Accommodates large number of low-traffic users with slotted Aloha reservations
         3. Can incorporate QoS
         4. Handles large delay-bandwidth product via delayed grants
      5. Polling
         1. Generalization of time-division multiplexing
         2. Provides fairness through regular access opportunities
         3. Can provide bounds on access delay
         4. Performance deteriorates with large delay-bandwidth product

Local Area Network

1. Short distance (~1km) between computers
   1. low cost
   2. high-speed, relatively error-free communication
   3. complex error control unnecessary
   4. ***Broadcast*** *messages to all machines in the LAN*
2. LAN structure
   1. 
   2. 
      1. Connectionless from transfer service
      2. Machine identify = MAC/Physical add
3. LLC Logical Link Control
   1. Between Network & MAC sub
   2. Provide 3 service
      1. Type 1: Unacknowledged connectionless service
         * Unnumbered frame mode of HDLC
      2. Type 2: Reliable connection-oriented service
         * Asynchronous balanced mode of HDLC
      3. Type 3: Acknowledged connectionless service
4. Wireless LAN