Electrical Engineering E6761
Computer Communication Networks
Lecture 7
Multicast + Link Layer

Professor Dan Rubenstein
Tues 4:10-6:40, Mudd 1127
Course URL:
http://www.cs.columbia.edu/~danr/EE6761

Overview

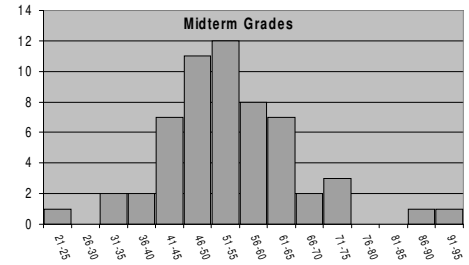
- Midterm results (on-campus)
  - CVN tests still being graded
- Project
  - form groups
  - groups should meet with me this or next week - You contact me!
- Mid-course evaluations:

http://oracle.seas.columbia.edu

- Lecture
  - Multicast
    - Review Multicast Group Concept
    - Theory
    - Example protocols (DVMRP, CBT, PIM, EXPRESS)
    - Reliability
  - Link Layer
    - Error detection / correction
    - Multiple Access Protocols
    - PPP
    - If time: ATM, Frame Relay, X25

Midterm Results

- Mean: 53.8
- Median: 53



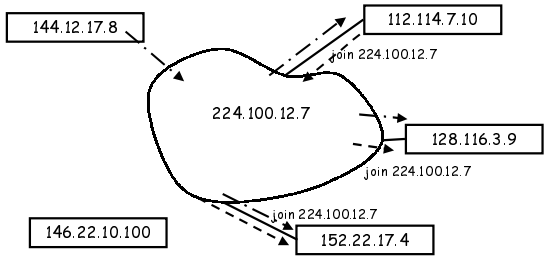
Transport Layer Multicast

Requires Multicast IP addressing

- class D addresses (224.0.0.0 - 239.255.255.255) reserved for multicast
- each address identifies a **multicast group**
  - address not explicitly associated with any host
  - hosts must join to the group to receive data sent to the group
- Any sender that sends to the multicast group will have its transmission delivered to all receivers joined to the multicast group

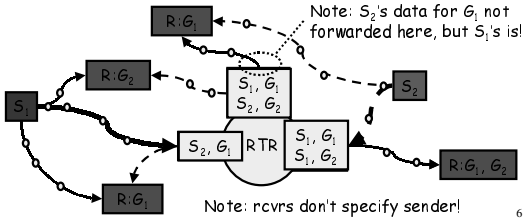
  (Note: delivery is UDP-like: unreliable, no order guarantees, etc.)
- joins accomplished through a socket interface

Multicast Example



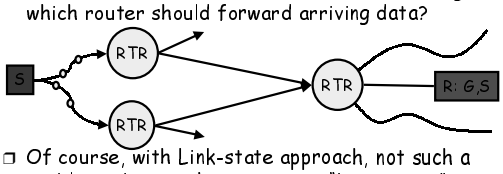
Router State for Multicast

- For each interface, router maintains (Source, Group) pairs
- (S,G) exists at an interface *i* if packets originating at S destined for multicast group G should be forwarded through *i*. Why distinguish source?



Multicast Routing vs. Unicast Routing

- In Multicast (using distance-vector):
  - A packet can be routed on multiple outgoing interfaces
  - The packet's final destination(s) are unknown by intermediate routers
- As a result, can't do destination-based routing, so which router should forward arriving data?



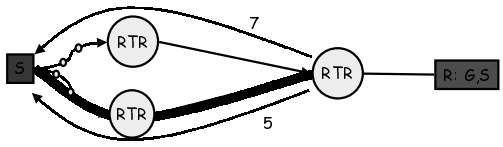
- Of course, with Link-state approach, not such a problem, since each router sees "big picture"

2 Distance Vector Issues for Multicast

- 1: How should the direction of routes be decided?
  - i.e., which router should be a parent?
- 2: How / when should this direction info be propagated?
  - You have a sender that wants to reach receivers, but doesn't know where the receivers are
  - You have receivers that would want to get data from a sender, but might not know sender existence

Choosing Route: Reverse Path Routing

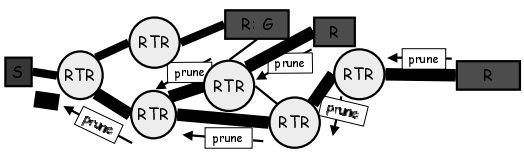
- Router takes a packet from the previous hop on its shortest path back to the source
- Assumption needed for shortest path routing: paths in reverse directions have same (or proportional) distance as fwd direction



9

Propagation method #1: Flood-and-prune

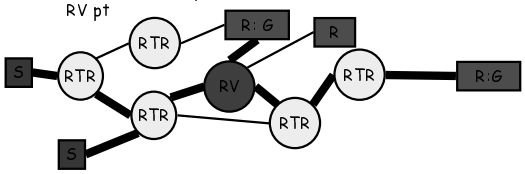
- Initially, assume a receiver downstream wants information
- Routers that receive a packet and "know" that it need not be forwarded downstream request a prune to their upstream router
- Routers do not forward down a pruned interface until the prune state times out (& prune process repeats)



10

Prop method #2: Rendez-Vous Points

- Connect to "special router" (i.e., the rendez-vous point) in the network
  - Sender's transmissions go to rendez-vous point, and then "branch out"
  - receiver join requests head toward rendez-vous point
  - Can renegotiate path after contact established to avoid RV pt



11

Prop method #3: Sender-specific joins

- Session model: multicast session has a single sender and receivers know identity (e.g., IP address) of the sender

12

Pros & Cons

Cons

- Reverse-Path Flooding:
  - requires symmetric paths for optimal shortest path routing
- Flood-and-prune:
  - bandwidth waste during flooding stage
- Rendez-vous points
  - not shortest paths
  - single-point of failure
- Sender-specific joins
  - limited to single sender

Pros

- Reverse-Path Flooding:
  - no loops
- Flood-and-prune:
  - rcvr wanting data doesn't miss any
- Rendez-vous points
  - no flooding
- Sender-specific joins
  - simple
  - often sessions have only one sender

13

Protocol Examples

- DVMRP (Distance Vector Multicast Routing Protocol), PIM (Protocol Independent Multicast) Dense Mode:
  - multi-source, flood & prune
- CBT (Core-Based Trees), PIM Sparse Mode
  - multi-source
  - rendez-vous points
- EXPRESS
  - single-source

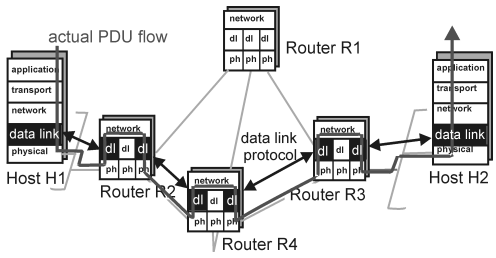
14

Reliable Multicast (Transport Layer)

- Problem: How to guarantee many receivers reliably receive data
  - Need ACK from every receiver?
  - Just NAKs are sufficient, but with many receivers and high loss rates, still too much sender processing
- Solution: NAK-based protocols +
  - hierarchy (ACK trees)
  - rcvrs wait random time, then broadcast NAKs (if rcv other NAK before broadcast, suppress own broadcast)
  - Forward Error Correction (FEC) techniques

15

Link Layer Protocols



16

Link Layer Services

- ***Framing and link access***:
  - encapsulate datagram into frame adding header and trailer,
  - implement channel access if shared medium,
  - 'physical (MAC) addresses' are used in frame headers to identify source and destination of frames on broadcast links
- ***Reliable Delivery***:
  - seldom used on fiber optic, co-axial cable and some twisted pairs too due to low bit error rate (not worth the overhead).
  - Used on wireless links, where the goal is to reduce errors thus avoiding end-to-end retransmissions

17

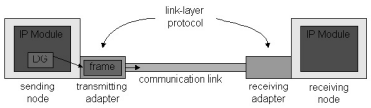
Link Layer Services (more)

- ***Flow Control***:
  - pacing between senders and receivers
- ***Error Detection***:
  - errors are caused by signal attenuation and noise.
  - Receiver detects presence of errors:
  - it signals the sender for retransmission or just drops the corrupted frame
- ***Error Correction***:
  - mechanism for the receiver to locate and correct the error without resorting to retransmission
  - Note: can't guarantee repair (w/ finite set of bits)

18

Link Layer Protocol Implementation

- Link layer protocol entirely implemented in the adapter (eg,PCMCIA card). Adapter typically includes: RAM, DSP chips, host bus interface, and link interface
- Adapter **send** operations: encapsulates (set sequence numbers, feedback info, etc.), adds error detection bits, implements channel access for shared medium, transmits on link
- Adapter **receive** operations: error checking and correction, interrupts host to send frame up the protocol stack, updates state info regarding feedback to sender, sequence numbers, etc.

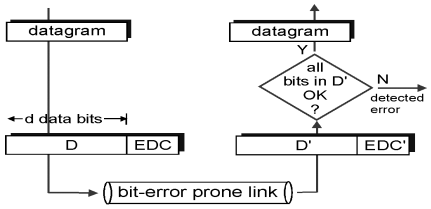


19

Error Detection

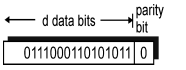
EDC= Error Detection and Correction bits (redundancy)
D = Data protected by error checking, may include some header fields

- Error detection is not 100%;
- protocol may miss some errors, but rarely
- Larger EDC field yields better detection and correction

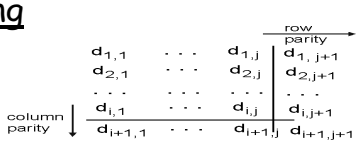


20

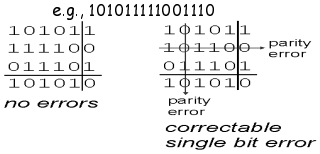
Parity Checking



Single Bit Parity:
Detect single bit errors:
sum of bits + parity = 0 (mod 2)

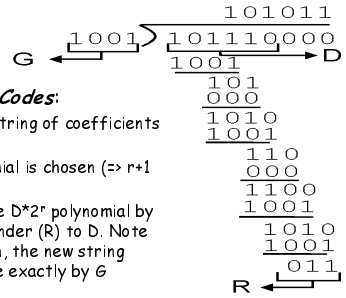


e.g., 101011111001110



Two Dimensional Bit Parity:
Detect and correct single bit errors
Note: 4 bit errors may go undetected

CRC

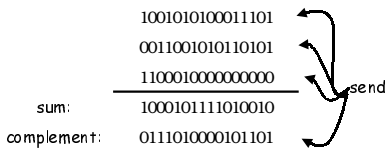


- ***Cyclic Redundancy Codes***:
  - Data is viewed as a string of coefficients of a polynomial (D)
  - A *Generator* polynomial is chosen (=> r+1 bits), (G)
  - Divide (modulo 2) the D*2^r polynomial by G. Append the remainder (R) to D. Note that, by construction, the new string <D,R> is now divisible exactly by G

$D * 2^r$ XOR R *mathematical formula*

Checksumming Methods

- ***Internet Checksum***: View data as made up of 16 bit integers; add all the 16 bit fields (one's complement arithmetic) and append the frame with the resulting sum; the receiver repeats the same operation and matches the checksum sent with the frame



The sum of sent vectors is a vector of 1's

CRC Implementation (cont)

- The sender carries out on-line, in hardware the division of the string D by the polynomial G and appends the remainder R to it
- The receiver divides < D,R> by G; if the remainder is non-zero, the transmission was corrupted
- International standards for G polynomials of degrees 8, 12, 15 and 32 have been defined
- ARPANET was using a 24 bit CRC for the alternating bit link protocol
- ATM is using a 32 bit CRC in ALL 5
- HDLC uses a 16 bit CRC

Multiple Access Links and Protocols

Three types of links:

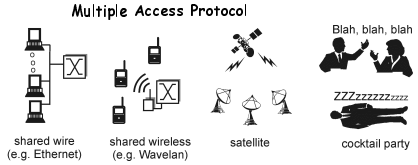
(a) Point-to-point (single wire)

(b) Broadcast (shared wire or medium; eg, E-net, wireless, etc.)

(c) Switched (eg, switched E-net, ATM etc)

We start with **Broadcast** links. Main challenge:

Multiple Access Protocol



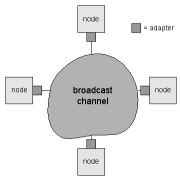
Q: How should multiple senders / receivers share a common transmission medium?

25

Multiple Access Control (MAC) Protocols

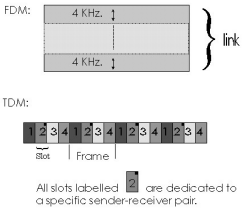
- MAC protocol: coordinates transmissions from different stations in order to minimize/avoid collisions
- (a) **Channel Partitioning** MAC protocols
- (b) **Random Access** MAC protocols
- (c) "**Taking turns**" MAC protocols

- Goals: **efficient, fair, simple, decentralized**



26

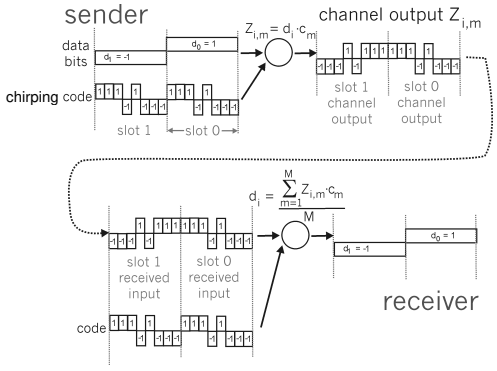
Channel Partitioning MAC protocols



- TDM (Time Division Multiplexing): channel divided into N time slots, one per user; inefficient with low duty cycle users and at light load.
- FDM (Frequency Division Multiplexing): frequency subdivided.

27

CDMA (Code division) Encode/Decode

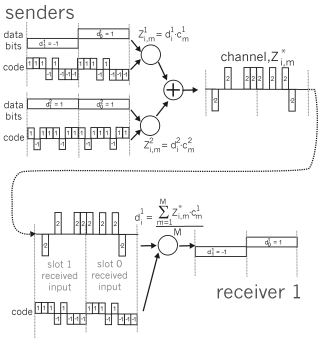


28

Channel Partitioning (CDMA)

- **CDMA (Code Division Multiple Access)**: exploits **spread spectrum** (DS or FH) encoding scheme
- unique "code" assigned to each user; ie, **code set** partitioning
- Used mostly in **wireless** broadcast channels (cellular, satellite,etc)
- All users share the **same frequency**, but each user has **own "chipping" sequence** (ie, code)
- Chipping sequence like a **mask**: used to **encode** the signal
- **encoded signal** = (original signal) X (chipping sequence)
- **decoding**: innerproduct of encoded signal and chipping sequence (note, the innerproduct is the sum of the component-by-component products)
- To make CDMA work, chipping sequences must be chosen orthogonal to each other (i.e., innerproduct = 0)

CDMA: two-sender interference



CDMA (cont'd)

CDMA Properties:

- protects users from interference and jamming (used in WW II)
- protects users from radio multipath fading
- allows multiple users to "coexist" and transmit simultaneously with minimal interference (if codes are "orthogonal")
- Pf: Let A & B be two orthogonal chirping codes (A·B = 0), D be data. Signal = (A+B) D

  A·(A+B) D = (A·A) D + (A·B) D = (A·A)D = |A|D

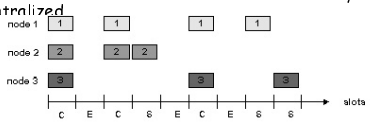
Random Access protocols

- A node transmits **at random** (ie, no a priory coordination among nodes) at **full** channel data rate R.
- If two or more nodes "**collide**", they retransmit later with random time between transmission
- The **random access MAC** protocol specifies how to detect collisions and how to recover from them (via delayed retransmissions, for example)
- Examples of random access MAC protocols:

(a) SLOTTED ALOHA
(b) ALOHA
(c) CSMA and CSMA/CD

Slotted Aloha

- Time is divided into equal size slots (= time to deliver full packet across unbridged part of LAN)
- a newly arriving station transmits a the beginning of the next slot
- if collision occurs (assume channel feedback, eg the receiver informs the source of a collision), the source retransmits the packet at each slot with probability P, until successful.
- Success (S), Collision (C), Empty (E) slots
- S-ALOHA is channel utilization efficient; it is fully decentralized



Slotted Aloha efficiency

If N stations have packets to send, and each transmits in each slot with probability *p*, the probability of successful transmission *S* is:

For a particular node, $S = p\,(1-p)^{(N-1)}$

For an arbitrary node of the N,

S = Prob (only one transmits) = $N\,p\,(1-p)^{(N-1)}$

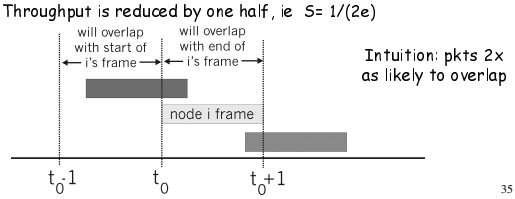
Optimal value of P: P = 1/N

For example, if N=2, S= .5

For N very large one finds S= 1/e (approximately, .37)

Pure (unslotted) ALOHA

- Slotted ALOHA requires slot synchronization
- A simpler version, pure ALOHA, does not require slots
- A node transmits without awaiting for the beginning of a slot
- Collision probability increases (packet can collide with other packets which are transmitted within a window twice as large as in S-Aloha)
- Throughput is reduced by one half, ie S= 1/(2e)

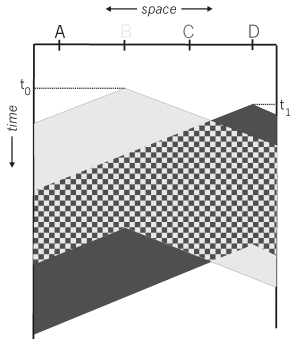


Intuition: pkts 2x as likely to overlap

CSMA (Carrier Sense Multiple Access)

- **CSMA**: listen before transmit. If channel is sensed busy, defer transmission
- **Persistent** CSMA: retry immediately when channel becomes idle (this may cause instability)
- **Non persistent CSMA**: retry after random interval
- Note: collisions may still exist, since two stations may sense the channel idle at the same time (or better, within a "vulnerable" window = round trip delay)
- In case of collision, the entire pkt transmission time is wasted

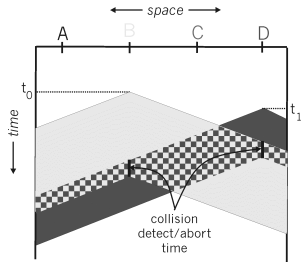
CSMA collisions



CSMA/CD (Collision Detection)

- **CSMA/CD**: carrier sensing and deferral like in CSMA. But, collisions are detected within a few bit times.
- Transmission is then aborted, reducing the channel wastage considerably.
- Typically, **persistent** retransmission is implemented
- Collision detection is **easy in wired LANs** (eg, E-net): can measure signal strength on the line, or code violations, or compare tx and receive signals
- Collision detection **cannot be done in wireless LANs** (the receiver is shut off while transmitting, to avoid damaging it with excess power)
- CSMA/CD can approach channel utilization =1 in LANs (low ratio of propagation over packet transmission time)

CSMA/CD collision detection



CSMA/CD

A: sense channel, **if** idle
then {
transmit and monitor the channel;
If detect another transmission
then {
abort and send jam signal;
update # collisions;
delay as required by exponential backoff algorithm;
goto A
}
else {done with the frame; set collisions to zero}
}
else {wait until ongoing transmission is over and goto A}

CSMA/CD (more)

- **Jam Signal**: to make sure all other transmitters are aware of the collision; 48 bits
  (transmitters either see collision or else they receive intact jam signal)
- **Exponential Backoff**:
  - Goal is too adapt the offered rate by transmitters to the estimated current load (ie backoff when load is heavy)
  - After the first collision Choose K from {0,1}; delay is K x 512 bit transmission times
  - After second collision choose K from {0,1,2,3}…
  - After ten or more collisions, choose K from {0,1,2,3,4,…,1023}

CSMA/CD (more)

- Note that under this scheme a new frame has a chance of sneaking in in the first attempt, even in heavy traffic
- **Ethernet Efficiency**: under heavy traffic and large number of nodes:

$$Efficiency = \frac{1}{1 + (5 * \frac{t_{prop}}{\cdot})}$$

"Taking Turns" MAC protocols

- So far we have seen that **channel partitioning** MAC protocols (TDM, FDM and CDMA) can share the channel fairly; but a single station cannot use it all
- **Random access** MAC protocols allow a single user full channel rate; but cannot share the channel fairly (in fact, **capture** is often observed)
- Also there are "taking turns" protocols…

"Taking Turns" MAC protocols

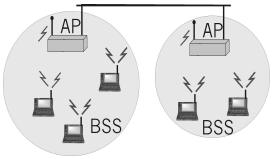
- **Taking Turns** MAC protocols achieve both fairness and full rate, at the expense of some extra control overhead

  (a) Polling: a Master station on a LAN in turn "invites" the slave stations to transmit their packets (up to a Max). Problems: Request to Send/Clear to Send overhead, latency, single point of failure (Master)

  (b) Token passing: the control **token** is passed from one node to the next sequentially. Can alleviate the latency and improve fault tolerance (in a token bus configuration). Still, elaborate procedures to recover from **lost token,** etc.

IEEE 802.11 Wireless LAN

- Wireless LANs are becoming popular for mobile Internet access
- Applications: nomadic Internet access, portable computing, ad hoc networking (multihopping)
- IEEE 802.11 standards defines MAC protocol; unlicensed frequency spectrum bands: 900Mhz, 2.4Ghz
- **Basic Service Sets + Access Points => Distribution System**
- Like a bridged LAN (flat MAC address)



45

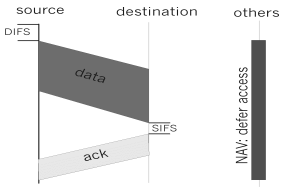
IEEE 802.11 MAC Protocol

CSMA Protocol:

- sense channel idle for **DISF** sec (Distributed Inter Frame Space)
  - transmit frame (no Collision Detection)
  - receiver returns ACK after **SIFS** (Short Inter Frame Space)

-if channel sensed busy then expo. backoff

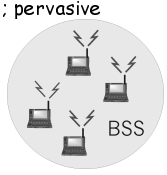
NAV: Network Allocation Vector (min time of deferral)



47

Ad Hoc Networks

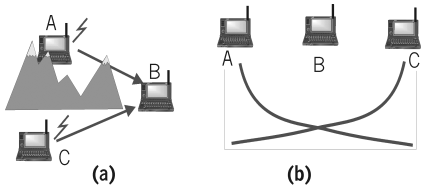
- IEEE 802.11 stations can dynamically form a group without AP
- Ad Hoc Network: no pre-existing infrastructure
- Applications: "laptop" meeting in conference room, car, airport; interconnection of "personal" devices (see bluetooth.com); battlefield; pervasive computing (smart spaces)
- IETF MANET (Mobile Ad hoc Networks) working group



46

Hidden Terminal effect

- CSMA inefficient in presence of hidden terminals
- Hidden terminals: A and B cannot hear each other because of obstacles or signal attenuation; so, their packets collide at B
- Solution? **CSMA/CA**
- **CA** = Collision Avoidance



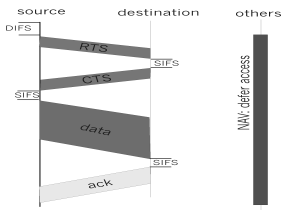
48

Collision Avoidance: RTS-CTS exchange

- Sender sends short RTS (request to send) request
- Rcvr chooses 1 sender and sends it CTS (clear to send)
- **CTS** "freezes" stations within range of receiver (but possibly hidden from transmitter); this prevents collisions by hidden station during data

•**RTS** and **CTS** are very short: collisions during data phase are thus very unlikely (the end result is similar to Collision Detection)

•Note: IEEE 802.11 allows **CSMA, CSMA/CA** and "**polling**" from AP



Point to Point protocol (PPP)

- Point to point, wired data link easier to manage than broadcast link: no Media Access Control
- Several Data Link Protocols: PPP, HDLC, SDLC, Alternating Bit protocol, etc
- PPP (Point to Point Protocol) is very popular: used in dial up connection between residential Host and ISP; on SONET/SDH connections, etc
- PPP is extremely simple (the simplest in the Data Link protocol family) and very streamlined

PPP Requirements

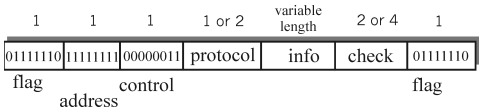
- Pkt framing: encapsulation of packets
- bit transparency: must carry any bit pattern in the data field
- error detection (no correction)
- multiple network layer protocols
- connection liveness
- Network Layer Address negotiation: Hosts/nodes across the link must learn/configure each other's network address

Not Provided by PPP

- error correction/recovery
- flow control
- sequencing
- multipoint links (e.g., polling)

PPP Data Frame

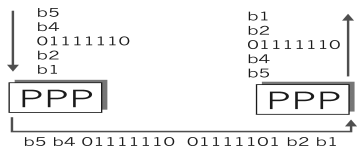
- Flag: delimiter (framing)
- Address: does nothing (only one option)
- Control: does nothing; in the future possible multiple control fields
- Protocol: upper layer to which frame must be delivered (eg, PPP-LCP, IP, IPCP, etc)



53

Byte Stuffing

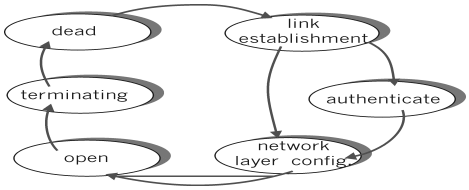
- For "data transparency", the data field must be allowed to include the pattern <01111110> ; ie, this must not be interpreted as a flag
- to alert the receiver, the transmitter "stuffs" an extra < 01111110> byte after each < 01111110> data byte
- the receiver discards each 01111110 followed by another 01111110, and continues data reception



54

PPP Data Control Protocol

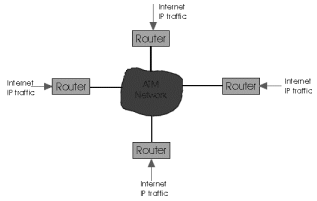
- PPP-LCP establishes/releases the PPP connection; negotiates options
- Starts in DEAD state
- Options: max frame length; authentication protocol
- Once PPP link established, IPCP (Control Protocol) moves in (on top of PPP) to configure IP network addresses etc.



55

ATM

- **ATM** (Asynchronous Transfer Mode) is the switching and transport technology of the B-ISDN (Broadband ISDN) architecture (1980)
- Goals: **high speed** access to business and residential users (155Mbps to 622 Mbps); **integrated services** support (voice, data, video, image)



56

ATM VCs

- Focus on **bandwidth allocation** facilities (in contrast to IP best effort)
- ATM main role today: "**switched**" link layer for **IP-over-ATM**
- ATM is a **virtual circuit** transport: cells (53 bytes) are carried on VCs
- in IP over ATM: **Permanent VCs** (PVCs) between IP routers;
- scalability problem: *N(N-1)* VCs between all IP router pairs

57

ATM VCs

- **Switched VCs** (SVCs) used for short lived connections
- **Pros of ATM VC** approach:
  - Can guarantee QoS performance to a connection mapped to a VC (bandwidth, delay, delay jitter)
- **Cons of ATM VC** approach:
  - Inefficient support of **datagram** traffic; PVC solution (one PVC between each host pair) does **not scale**;
  - SVC introduces excessive **latency** on short lived connections
  - High **SVC processing Overhead**

58

ATM Address Mapping

- Router interface (to ATM link) has **two addresses**: IP and ATM address.
- To route an IP packet through the ATM network, the IP node:

(a) inspects own routing tables to find **next IP router** address
(b) then, using **ATM ARP table**, finds ATM addr of next router
(c) passes packet (with ATM address) to ATM layer

- At this point, the ATM layer takes over:

(1) it determines the **interface and VC** on which to send out the packet
(2) if no VC exists (to that ATM addr) a **SVC** is set up

59

ATM Physical Layer

- Two **Physical sublayers**:
- (a) **Physical Medium Dependent** (PMD) sublayer
  - (a.1) **SONET/SDH**: transmission frame structure (like a container carrying bits);
    - bit synchronization;
    - bandwidth partitions (TDM);
    - several speeds: OC1 = 51.84 Mbps; OC3 = 155.52 Mbps; OC12 = 622.08 Mbps
  - (a.2) **TI/T3**: transmission frame structure (old telephone hierarchy): 1.5 Mbps/ 45 Mbps
  - (a.3) **unstructured**: just cells (busy/idle)

60

ATM Physical Layer (more)

- Second physical sublayer

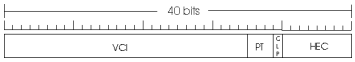
  (b) **Transmission Convergence Sublayer (TCS)**: it adapts PMD sublayer to ATM transport layer

- TCS Functions:
  - Header **checksum** generation: 8 bits CRC; it protects a 4-byte header; can correct all single errors.
  - Cell **delineation**
  - With "unstructured" PMD sublayer, transmission of **idle cells** when no data cells are available in the transmit queue

61

ATM Layer

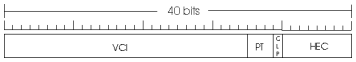
- **ATM layer** in charge of transporting cells across the ATM network
- ATM layer protocol defines ATM cell **header format** (5bytes);
- **payload** = 48 bytes; total cell length = 53 bytes



62

ATM Layer

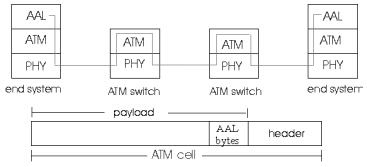
- **VCI** (virtual channel ID): translated from link to link;
- **PT** (Payload type): indicates the type of payload (eg mngt cell)
- **CLP (**Cell Loss Priority) bit: CLP = 1 implies that the cell is low priority cell, can be **discarded** if router is congested
- **HEC** (Header Error Checksum) byte



63

ATM Adaptation Layer (AAL)

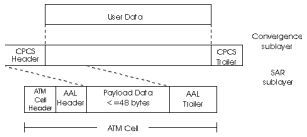
- ATM **Adaptation Layer** (AAL): "adapts" the ATM layer to the upper layers (IP or native ATM applications)
- AAL is present **only in end systems**, not in switches
- The AAL layer has its **header/trailer fields**, carried in the ATM cell



64

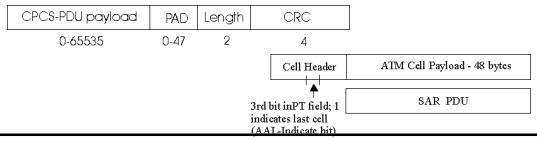
ATM Adaption Layer (AAL) [more]

- **Different versions** of AAL layers, depending on the service to be supported by the ATM transport:
  - **AAL1**: for CBR (Constant Bit Rate) services such as circuit emulation
  - **AAL2**: for VBR (Variable Bit Rate) services such as MPEG video
  - **AAL5**: for data (eg, IP datagrams)



AAL5 - Simple And Efficient AL (SEAL)

- **AAL5**: **low overhead** AAL used to carry IP datagrams
  - SAR header and trailer eliminated; CRC (4 bytes) moved to CPCS
  - PAD ensures payload multiple of 48bytes (LENGTH = PAD bytes)
  - At destination, cells are **reassembled based on VCI number**; AAL **indicate bit** delineates the CPCS-PDU; if CRC fails, PDU is dropped, else, passed to Convergence Sublayer and then IP

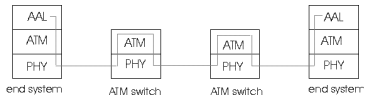


ATM Adaption Layer (AAL) [more]

- **Two sublayers** in AAL:
  - **(Common Part) Convergence Sublayer**: encapsulates IP payload
  - **Segmentation/Reassembly Sublayer**: segments/reassembles the CPCS (often quite large, up to 65K bytes) into 48 byte ATM segments

Datagram Journey in IP-over-ATM Network

- **At Source Host**:
  - (1) IP layer finds the mapping between IP and ATM exit address (using ARP); then, passes the datagram to AAL5
  - (2) AAL5 **encapsulates** datg and it **segments** to cells; then, down to ATM
- **In the network**, the ATM layer moves cells from switch to switch, along a **pre-established VC**
- **At Destination Host**, AAL5 **reassembles** cells into original datg;
  - if CRC OK, datgram is passed up the IP protocol.



ARP in ATM Nets

- ATM can route cells only if it has the ATM address
  - Thus, IP must **translate** exit IP address to ATM address
- The IP/ATM addr translation is done by **ARP** (Addr Recogn Protocol)
- Generally, ATM ARP table does not store **all ATM addresses**: it must **discover** some of them

- **Two techniques**:
  - broadcast
  - ARP servers

69

ARP in ATM Nets (more)

- (1) **Broadcast** the ARP request to **all destinations**:
  - (1.a) the **ARP Request** msg is broadcast to all ATM destinations using a special broadcast VC;
  - (1.b) the ATM destination which can match the IP address returns (via unicast VC) the IP/ATM address map;
- Broadcast overhead prohibitive for large ATM nets.

70

ARP in ATM Nets (more)

- (2) **ARP Server**:
  - (2.a) **source IP router** forwards ARP request to server on dedicated VC (Note: all such VCs from routers to ARP have same ID)
  - (2.b) **ARP server** responds to source router with IP/ATM translation
- Hosts must **register** themselves with the ARP server

Comments: more **scaleable** than ABR Broadcast approach (no broadcast storm). However, it requires an **ARP server**, which may be swamped with requests

71

X.25 and Frame Relay

- Wide Area Network technologies (like ATM); also, both **Virtual Circuit** oriented , like ATM
- **X.25** was born in mid '70s, with the support of the**Telecom Carriers**, in response to the ARPANET datagram technology (religious war..)
- **Frame relay** emerged from ISDN technology (in late '80s)
- Both X.25 and Frame Relay can be used to **carry IP datagrams**; thus, they are viewed as **Link Layers** by the IP protocol layer (and are thus covered in this chapter)

72

X.25

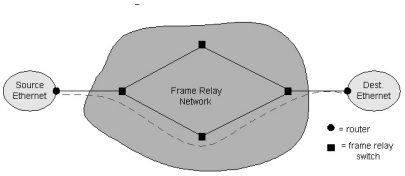
- X.25 builds a VC between source and destination for **each user connection**
- Along the path, **error control** (with retransmissions) on each hop using LAP-B, a variant of the HDLC protocol
- Also, on each VC, hop by hop **flow control** using **credits**;
  - congestion arising at an intermediate node propagates to source via **backpressure**

X.25

- As a result, packets are delivered **reliably** and in **sequence** to destination; per flow credit control guarantees **fair sharing**
- Putting "**intelligence into the network**" made sense in mid 70s (dumb terminals without TCP)
- Today, TCP and practically error free fibers favor pushing the "**intelligence to the edges**"; moreover, gigabit routers cannot afford the X.25 processing overhead
- As a result, X.25 is rapidly becoming **extinct**

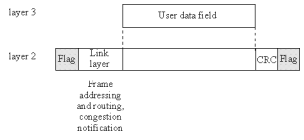
Frame Relay

- Designed in late '80s and widely deployed in the '90s
- FR VCs have **no error control**
- **Flow (rate) control is end to end**; much less processing O/H than hop by hop credit based flow control



Frame Relay (more)

- Designed to **interconnect** corporate customer LANs
- Each VC is like a "**pipe**" carrying aggregate traffic between two routers
- Corporate customer **leases** FR service from a public Frame Relay network (eg, Sprint or ATT)
- Alternative, large customer may build **Private** Frame Relay network.



Frame Relay (more)

- Frame Relay implements mostly **permanent VCs** (aggregate flows)
- 10 bit VC ID field in the Frame header
- If IP runs on top of FR, the VC ID corresponding to destination IP address is looked up in the **local VC table**
- FR switch simply **discards** frames with bad CRC (TCP retransmits..)

77

Frame Relay -VC Rate Control

- **CIR** = Committed Information Rate, defined for each VC and negotiated at VC set up time; customer **pays** based on CIR
- **DE bit** = Discard Eligibility bit in Frame header
  - DE bit = 0: **high priority**, rate compliant frame; the network will try to deliver it at "all costs"
  - DE bit = 1: **low priority**, "marked" frame; the network discards it when a link becomes congested (ie, threshold exceeded)

78

Frame Relay - CIR & Frame Marking

- **Access Rate**: rate **R** of the access link between **source router** (customer) and **edge FR switch** (provider); 64Kbps < **R** < 1,544Kbps
- Typically, **many VCs** (one per destination router) multiplexed on the same access trunk; each VC has own **CIR**
- Edge FR switch **measures** traffic rate for each VC; it **marks**
- (ie DE <= 1) frames which **exceed** CIR (these may be later dropped)

79

Frame Relay - Rate Control

- Frame Relay provider "almost" guarantees CIR rate (except for **overbooking**)
- No **delay** guarantees, even for high priority traffic
- Delay will in part depend on rate **measurement interval Tc**; the larger Tc, the burstier the traffic injected in the network, the higher the delays
- Frame Relay provider must do **careful traffic engineering** before committing to CIR, so that it can back up such commitment and prevent overbooking
- Frame Relay CIR is the first example of traffic rate dependent **charging model** for a packet switched network

80