Chapter 5
Link Layer and LANs

A note on the use of these ppt slides:
We’re making these slides freely available to all (faculty, students, readers). They’re in PowerPoint form so you can add, modify, and delete slides (including this one) and slide content to suit your needs. They obviously represent a lot of work on our part. In return for use, we only ask the following:
- If you use these slides (e.g., in a class) in substantially unaltered form, that you mention their source (after all, we’d like people to use our book!)
- If you post any slides in substantially unaltered form on a www site, that you note that they are adapted from (or perhaps identical to) our slides, and note our copyright of this material.

Thanks and enjoy! JFK/KWR

All material copyright 1996-2009
J.F Kurose and K.W. Ross, All Rights Reserved

I am enormously grateful to Jim Kurose and Keith Ross for making these slides available for our use in this course. If you need a comprehensive networking introductory book, I highly recommend this one.
Carl Hauser

Computer Networking: A Top Down Approach
5th edition.
Jim Kurose, Keith Ross
Addison-Wesley, April 2009.
The Data Link Layer: Layer 2

Our goals:

- understand principles behind data link layer services:
  - error detection, correction
  - sharing a broadcast channel: multiple access
  - link layer addressing

- instantiation and implementation of various link layer technologies
Link Layer Roadmap

1. Introduction and services
2. Error detection and correction
3. Multiple access protocols
4. Link-layer Addressing
5. Ethernet
6. Link-layer switches
7. PPP
8. Link virtualization: ATM, MPLS
Link Layer: Introduction

Some terminology:
- hosts and routers are **nodes**
- communication channels that connect adjacent nodes along communication path are **links**
  - wired links
  - wireless links
  - LANs
- layer-2 packet is a **frame**, encapsulates datagram

**data-link layer** has responsibility of transferring datagram from one node to adjacent node over a link
Link layer: context

- Datagrams are transferred by different link protocols over different links:
  - e.g., Ethernet on first link, frame relay on intermediate links, 802.11 on last link
- each link protocol provides different services
  - e.g., may or may not provide reliable data transfer (rdt) over link

transportation analogy
- trip from Princeton to Lausanne
  - limo: Princeton to JFK
  - plane: JFK to Geneva
  - train: Geneva to Lausanne
- tourist = datagram
- transport segment = communication link
- transportation mode = link layer protocol
- travel agent = routing algorithm
Link Layer Services

- framing, link access:
  - encapsulate datagram into frame, adding header, trailer
  - channel access if shared medium
  - "MAC" addresses used in frame headers to identify source, dest
    - different from IP (destination, network layer) address!
Link Layer Services (more)

- **flow control:**
  - pacing between adjacent sending and receiving nodes

- **error detection:**
  - errors caused by signal attenuation, noise.
  - receiver detects presence of errors:
    - signals sender for retransmission or drops frame

- **error correction:**
  - receiver identifies *and corrects* bit error(s) without resorting to retransmission

- **half-duplex and full-duplex**
  - with half duplex, nodes at both ends of link can transmit, but not at same time
Where is the link layer implemented?

- in each and every host
- link layer implemented in “adaptor” (aka network interface card NIC)
  - Ethernet card, PCMCI card, 802.11 card
  - implements link, physical layer
- attaches into host’s system buses
- combination of hardware, software, firmware
Adaptors Communicating

- **sending side:**
  - encapsulates datagram in frame
  - adds error checking bits, rdt, flow control, etc.

- **receiving side**
  - looks for errors, rdt, flow control, etc
  - extracts datagram, passes to upper layer at receiving side
Link Layer Roadmap

1. Introduction and services
2. Error detection and correction
3. Multiple access protocols
4. Link-layer Addressing
5. Ethernet
6. Link-layer switches
7. PPP
8. Link virtualization: ATM, MPLS
Error Detection

EDC = Error Detection and Correction bits (redundancy)

\[ D = \text{Data protected by error checking, may include header fields} \]

- Error detection not 100% reliable!
  - protocol may miss some errors, but rarely
  - larger EDC field yields better detection and correction

\[ \text{datagram} \]

\[ \text{d data bits} \]

\[ \text{D EDC} \]

\[ \text{bit-error prone link} \]

\[ \text{all bits in D' OK?} \]

\[ N \text{ detected error} \]

\[ \text{datagram} \]

\[ \text{D' EDC'} \]
Link Layer Roadmap

1. Introduction and services
2. Error detection and correction
3. Multiple access protocols
4. Link-layer Addressing
5. Ethernet
6. Link-layer switches
7. PPP
8. Link virtualization: ATM, MPLS
Multiple Access Links and Protocols

Two types of “links“:

- **point-to-point**
  - PPP for dial-up access
  - point-to-point link between Ethernet switch and host

- **broadcast** (shared wire or medium)
  - old-fashioned Ethernet
  - upstream HFC
  - 802.11 wireless LAN

![Diagram showing different types of access links](image-url)
Multiple Access protocols

- single shared broadcast channel
- two or more simultaneous transmissions by nodes:
  - interference
    - collision if node receives two or more signals at the same time

multiple access protocol

- distributed algorithm that determines how nodes share channel, i.e., determine when node can transmit
- communication about channel sharing must use channel itself!
  - no out-of-band channel for coordination
Ideal Multiple Access Protocol

Broadcast channel of rate R bps

1. when one node wants to transmit, it can send at rate R.

2. when M nodes want to transmit, each can send at average rate $R/M$.

3. fully decentralized:
   - no special node to coordinate transmissions
   - no synchronization of clocks, slots

4. simple
MAC Protocols: a taxonomy

Three broad classes:

- **Channel Partitioning**
  - divide channel into smaller “pieces” (time slots, frequency, code)
  - allocate piece to node for exclusive use

- **Random Access**
  - channel not divided, allow collisions
  - “recover” from collisions

- **“Taking turns”**
  - nodes take turns, but nodes with more to send can take longer turns
Channel Partitioning MAC protocols: TDMA

TDMA: time division multiple access

- access to channel in "rounds"
- each station gets fixed length slot (length = pkt trans time) in each round
- unused slots go idle
- example: 6-station LAN, 1,3,4 have pkt, slots 2,5,6 idle
Channel Partitioning MAC protocols: FDMA

FDMA: frequency division multiple access

- channel spectrum divided into frequency bands
- each station assigned fixed frequency band
- unused transmission time in frequency bands go idle
- example: 6-station LAN, 1,3,4 have pkt, frequency bands 2,5,6 idle

[Diagram of FDM cable and frequency bands]
Random Access Protocols

- When node has packet to send
  - transmit at full channel data rate $R$.
  - no a priori coordination among nodes

- two or more transmitting nodes $\Rightarrow$ “collision”,

- random access MAC protocol specifies:
  - how to detect collisions
  - how to recover from collisions (e.g., via delayed retransmissions)

- Examples of random access MAC protocols:
  - slotted ALOHA
  - ALOHA
  - CSMA, CSMA/CD, CSMA/CA
CSMA (Carrier Sense Multiple Access)

**CSMA**: listen before transmit:
- If channel sensed idle: transmit entire frame
- If channel sensed busy, defer transmission

- human analogy: don’t interrupt others!
CSMA collisions

collisions can still occur:
propagation delay means two nodes may not hear each other’s transmission

collision:
entire packet transmission time wasted

note:
role of distance & propagation delay in determining collision probability
CSMA/CD (Collision Detection)

CSMA/CD: carrier sensing, deferral as in CSMA
- collisions detected within short time
- colliding transmissions aborted, reducing channel wastage

- collision detection:
  - easy in wired LANs: measure signal strengths, compare transmitted, received signals
  - difficult in wireless LANs: received signal strength overwhelmed by local transmission strength

- human analogy: the polite conversationalist
CSMA/CD collision detection
“Taking Turns” MAC protocols

Token passing:
- control token passed from one node to next sequentially.
- token message
- concerns:
  - token overhead
  - latency
  - single point of failure (token)
Summary of MAC protocols

- **channel partitioning**, by time, frequency or code
  - Time Division, Frequency Division

- **random access** (dynamic),
  - ALOHA, S-ALOHA, CSMA, CSMA/CD
  - carrier sensing: easy in some technologies (wire), hard in others (wireless)
    - CSMA/CD used in Ethernet
    - CSMA/CA used in 802.11

- **taking turns**
  - polling from central site, token passing
  - Bluetooth, FDDI, IBM Token Ring
Link Layer Roadmap

1. Introduction and services
2. Error detection and correction
3. Multiple access protocols
4. Link-layer Addressing
5. Ethernet
6. Link-layer switches
7. PPP
8. Link virtualization: ATM, MPLS
MAC Addresses and ARP

- 32-bit Network address:
  - used to get datagram to destination IP subnet

- MAC (or LAN or physical or Ethernet) address:
  - function: *get frame from one interface to another physically-connected interface (same network)*
  - 48 bit MAC address (for most LANs)
    - burned in NIC ROM, also sometimes software settable
LAN Addresses and ARP

Each adapter on LAN has unique LAN address

Broadcast address = FF-FF-FF-FF-FF-FF

= adapter

LAN (wired or wireless)

1A-2F-BB-76-09-AD

71-65-F7-2B-08-53

58-23-D7-FA-20-B0

0C-C4-11-6F-E3-98
LAN Address (more)

- **MAC address allocation** administered by IEEE
- manufacturer buys portion of MAC address space (to assure uniqueness)
- analogy:
  - (a) MAC address: like Social Security Number
  - (b) IP address: like postal address
- **MAC flat address** ➜ portability
  - can move LAN card from one LAN to another
ARP: Address Resolution Protocol

Question: how to determine MAC address of B knowing B’s IP address?

- Each IP node (host, router) on LAN has ARP table
- ARP table: IP/MAC address mappings for some LAN nodes
  - IP address; MAC address; TTL
  - TTL (Time To Live): time after which address mapping will be forgotten (typically 20 min)
ARP protocol: Same LAN (network)

- A wants to send datagram to B, and B's MAC address not in A's ARP table.
- A broadcasts ARP query packet, containing B's IP address
  - dest MAC address = FF-FF-FF-FF-FF-FF
  - all machines on LAN receive ARP query
- B receives ARP packet, replies to A with its (B's) MAC address
  - frame sent to A's MAC address (unicast)
- A caches (saves) IP-to-MAC address pair in its ARP table until information becomes old (times out)
  - soft state: information that times out (goes away) unless refreshed
- ARP is "plug-and-play":
  - nodes create their ARP tables without intervention from net administrator
Addressing: routing to another LAN

walkthrough: send datagram from A to B via R
assume A knows B’s IP address

- two ARP tables in router R, one for each IP network (LAN)
A creates IP datagram with source A, destination B
A uses ARP to get R’s MAC address for 111.111.111.110
A creates link-layer frame with R’s MAC address as dest, frame contains A-to-B IP datagram
A’s NIC sends frame
R’s NIC receives frame
R removes IP datagram from Ethernet frame, sees its destined to B
R uses ARP to get B’s MAC address
R creates frame containing A-to-B IP datagram sends to B

This is a really important example – make sure you understand!
Link Layer Roadmap

1. Introduction and services
2. Error detection and correction
3. Multiple access protocols
4. Link-layer Addressing
5. Ethernet
6. Link-layer switches
7. PPP
8. Link virtualization: ATM, MPLS
Ethernet

“dominant” wired LAN technology:
- cheap $20 for NIC
- first widely used LAN technology
- simpler, cheaper than token LANs and ATM
- kept up with speed race: 10 Mbps – 10 Gbps

Metcalfe’s Ethernet sketch
Star topology

- bus topology popular through mid 90s
  - all nodes in same collision domain (can collide with each other)

- today: star topology prevails
  - active switch in center
  - each “spoke” runs a (separate) Ethernet protocol (nodes do not collide with each other)
Ethernet: Unreliable, connectionless

- **connectionless**: No handshaking between sending and receiving NICs
- **unreliable**: receiving NIC doesn’t send acks or nacks to sending NIC
  - stream of datagrams passed to network layer can have gaps (missing datagrams)
  - gaps will be filled if app is using TCP
  - otherwise, app will see gaps
- **Ethernet’s MAC protocol**: unslotted **CSMA/CD**
Ethernet CSMA/CD algorithm

1. NIC receives datagram from network layer, creates frame
2. If NIC senses channel idle, starts frame transmission
   If NIC senses channel busy, waits until channel idle, then transmits
3. If NIC transmits entire frame without detecting another transmission, NIC is done with frame!
4. If NIC detects another transmission while transmitting, aborts and sends jam signal
5. After aborting, NIC enters exponential backoff: after mth collision, NIC chooses K at random from \( \{0,1,2,...,2^m-1\} \). NIC waits \( K \cdot 512 \) bit times, returns to Step 2
Ethernet's CSMA/CD (more)

Jam Signal: make sure all other transmitters are aware of collision; 48 bits

Bit time: .1 microsec for 10 Mbps Ethernet; for K=1023, wait time is about 50 msec

Exponential Backoff:
- **Goal:** adapt retransmission attempts to estimated current load
  - heavy load: random wait will be longer
- first collision: choose K from \{0,1\}; delay is $K \cdot 512$ bit transmission times
- after second collision: choose K from \{0,1,2,3\}...
- after ten collisions, choose K from \{0,1,2,3,4,...,1023\}
802.3 Ethernet Standards: Link & Physical Layers

- many different Ethernet standards
  - common MAC protocol and frame format
  - different speeds: 2 Mbps, 10 Mbps, 100 Mbps, 1Gbps, 10G bps
  - different physical layer media: fiber, cable

![Diagram showing Ethernet standards and physical layers]
Hubs

... physical-layer ("dumb") repeaters:

- bits coming in one link go out *all* other links at same rate
- all nodes connected to hub can collide with one another
- no frame buffering
- no CSMA/CD at hub: host NICs detect collisions
Switch

- link-layer device: smarter than hubs, take active role
  - store, forward Ethernet frames
  - examine incoming frame’s MAC address, selectively forward frame to one-or-more outgoing links when frame is to be forwarded on segment, uses CSMA/CD to access segment

- transparent
  - hosts are unaware of presence of switches

- plug-and-play, self-learning
  - switches do not need to be configured
Switch: allows multiple simultaneous transmissions

- hosts have dedicated, direct connection to switch
- switches buffer packets
- Ethernet protocol used on each incoming link, but no collisions; full duplex
  - each link is its own collision domain
- **switching**: A-to-A' and B-to-B' simultaneously, without collisions
  - not possible with dumb hub

![Switch Diagram](image-url)

- switch with six interfaces
  - (1,2,3,4,5,6)
Switch Table

- **Q:** how does switch know that A' reachable via interface 4, B' reachable via interface 5?
- **A:** each switch has a switch table, each entry:
  - (MAC address of host, interface to reach host, time stamp)
- looks like a routing table!
- **Q:** how are entries created, maintained in switch table?
  - something like a routing protocol?
Switch: self-learning

- switch *learns* which hosts can be reached through which interfaces
  - when frame received, switch “learns” location of sender: incoming LAN segment
  - records sender/location pair in switch table

<table>
<thead>
<tr>
<th>MAC addr</th>
<th>interface</th>
<th>TTL</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1</td>
<td>60</td>
</tr>
</tbody>
</table>

Switch table (initially empty)
Institutional network

to external network

router

mail server

web server

IP subnet
Switches vs. Routers

- both store-and-forward devices
  - routers: network layer devices (examine network layer headers)
  - switches are link layer devices
- routers maintain routing tables, implement routing algorithms
- switches maintain switch tables, implement filtering, learning algorithms

![Network Diagram]

```plaintext

<table>
<thead>
<tr>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bridge</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```
Link Layer Roadmap

1. Introduction and services
2. Error detection and correction
3. Multiple access protocols
4. Link-layer Addressing
5. Ethernet
6. Link-layer switches
7. PPP
8. Link virtualization: ATM, MPLS
Link Layer Roadmap

1. Introduction and services
2. Error detection and correction
3. Multiple access protocols
4. Link-layer Addressing
5. Ethernet
6. Link-layer switches
7. PPP
8. Link virtualization: ATM, MPLS
Virtualization of networks

Virtualization of resources: powerful abstraction in systems engineering:

- computing examples: virtual memory, virtual devices
  - Virtual machines: e.g., java
  - IBM VM os from 1960’s/70’s
  - VMware, Virtual Box, ...

- layering of abstractions: don’t sweat the details of the lower layer, only deal with lower layers abstractly
Cerf & Kahn’s Internetwork Architecture

- two layers of addressing: internetwork and local network
- new layer (IP) makes everything homogeneous at internetwork layer
- underlying local network technology
  - cable
  - satellite
  - 56K telephone modem
  - Ethernet
  - DSL/Cable Modem/Wireless/...
ATM and MPLS

- ATM, MPLS separate networks in their own right
  - different service models, addressing, routing from Internet
- viewed by Internet as logical link connecting IP routers
  - just like dialup link is really part of separate network (telephone network)
Asynchronous Transfer Mode: ATM

- 1990's/00 standard for high-speed (155Mbps to 622 Mbps and higher) Broadband Integrated Service Digital Network architecture
- **Goal:** integrated, end-end transport of carry voice, video, data
  - meeting timing/QoS requirements of voice, video (versus Internet best-effort model)
  - “next generation” telephony: technical roots in telephone world
  - packet-switching (fixed length packets, called “cells”) using virtual circuits
ATM architecture

- Adaptation layer: only at edge of ATM network
  - data segmentation/reassembly
  - roughly analogous to Internet transport layer
- ATM layer: “network” layer
  - cell switching, routing
- Physical layer
ATM: network or link layer?

**Vision:** end-to-end transport: “ATM from desktop to desktop”
- ATM is a network technology

**Reality:** used to connect IP backbone routers
- “IP over ATM”
- ATM as switched link layer, connecting IP routers
Multiprotocol label switching (MPLS)

- initial goal: speed up IP forwarding by using fixed length label (instead of IP address) to do forwarding
  - borrowing ideas from Virtual Circuit (VC) approach
  - but IP datagram still keeps IP address!

![MPLS header diagram]

- PPP or Ethernet header
- MPLS header
- IP header
- remainder of link-layer frame

- label: 20
- Exp: 3
- S: 1
- TTL: 5
MPLS capable routers

- a.k.a. label-switched router
- forwards packets to outgoing interface based only on label value (don’t inspect IP address)
  - MPLS forwarding table distinct from IP forwarding tables
- signaling protocol needed to set up forwarding
  - RSVP-TE
    - forwarding possible along paths that IP alone would not allow (e.g., source-specific routing) !!
    - use MPLS for traffic engineering
- must co-exist with IP-only routers
Chapter 5: Summary

- principles behind data link layer services:
  - error detection, correction
  - sharing a broadcast channel: multiple access
  - link layer addressing

- instantiation and implementation of various link layer technologies
  - Ethernet
  - switched LANS
  - PPP
  - virtualized networks as a link layer: ATM, MPLS