Data Link Layer

**Textbook & (edited) slides:** *Computer Networking: A Top-Down Approach*. James F. Kurose, Keith W. Ross

**Reference:** *Computer Networks: A Systems Approach*. Larry Peterson, Bruce Davie, Morgan Kaufmann
Link Layer

- **Error detection, correction**
  - Multiple access protocols
  - LANs
    - Addressing, ARP
    - Ethernet
    - Switches
    - VLANs
  - Link virtualization: MPLS
  - Data center networking
Error Detection

• EDC = Error Detection and Correction bits (redundancy)
• D = 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
Parity Checking

Single bit parity:
Detect single bit errors

Two-dimensional bit parity:
Detect and correct single bit errors

0111000110101011 0

101011
111100
011101
001010
no errors

101011
101100
011101
010100
parity error
correctable single bit error
### Internet Checksum

**Goal:** Detect errors (e.g. flipped bits) in transmitted packet  
(note: used at transport layer only)

**Sender:**
- Treat segment contents as sequence of 16-bit integers
- Checksum: addition (1’s complement sum) of segment contents
- Sender puts checksum value into UDP checksum field
Cyclic Redundancy Check

- More powerful error-detection coding
- View data bits, $D$, as a binary number
- Choose $r+1$ bit pattern (generator), $G$
- Goal: choose $r$ CRC bits, $R$, such that
  - $<D,R>$ exactly divisible by $G$ (modulo 2)
  - Receiver knows $G$, divides $<D,R>$ by $G$. If non-zero remainder: error detected!
  - Can detect all burst errors less than $r+1$ bits
- Widely used in practice (Ethernet, 802.11 WiFi, ATM)
Example: CRC

Want: \( D.2^r \text{ XOR } R = nG \)

Equivalently: \( D.2^r = nGXOR R \)

Equivalently: If we divide \( D.2^r \)

By \( G \), want remainder \( R \) to satisfy:

\[
R = \text{remainder}\left[\frac{D.2^r}{G}\right]
\]
Link Layer

• Error detection, correction
✓ Multiple access protocols
• LANs
  • Addressing, ARP
  • Ethernet
  • Switches
  • VLANS
• Link virtualization: MPLS
• Data center networking
Multiple Access Links & Protocols

Two types of links

- **Point-to-point**
  - PPP for dial-up access
  - Point-to-point link between Ethernet switch, host
- **Broadcast (shared wire or medium)**
  - Old-fashioned Ethernet
  - Upstream HFC
  - 802.11 wireless LAN
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

**Given:** broadcast channel of rate R bps

**Desiderata**

- When one node wants to transmit, it can send at rate R.
- When M nodes want to transmit, each can send at average rate R/M.
- Fully decentralized
  - No special node to coordinate transmissions
  - No synchronization of clocks, slots
- Simple
MAC Protocols: 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 = packet transmission time) in each round
- Unused slots go idle
- Example: 6-station LAN, 1,3,4 have packets to send, 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 packet to send, frequency bands 2,5,6 idle
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
Slotted ALOHA

Assumptions

• All frames same size

• Time divided into equal size slots (time to transmit one frame)

• Nodes start to transmit only slot beginning

• Nodes are synchronized

• If two or more nodes transmit in slot, all nodes detect collision
Slotted ALOHA

Operation

• When node obtains fresh frame, transmits in next slot
  • **If no collision:** Node can send new frame in next slot
  • **If collision:** Node retransmits frame in each subsequent slot with probability $p$
    until success
Slotted ALOHA

Pros
• Single active node can continuously transmit at full rate of channel
• Highly decentralized: only slots in nodes need to be in sync
• Simple

Cons
• Collisions, wasting slots
• Idle slots
• Nodes may be able to detect collision in less than time to transmit packet
• Clock synchronization
Slotted ALOHA: Efficiency

- **Efficiency**: long-run fraction of successful slots
  (many nodes, all with many frames to send)

- **Suppose**: $N$ nodes with many frames to send, each transmits in slot with probability $p$

- Probability that given node has success in a slot $= P(1 - P)^{n-1}$

- Probability that any node has a success $= NP (1 - p)^{n-1}$
Slotted ALOHA: Efficiency

• Max efficiency: find $p^*$ that maximizes $NP(1-P)^{N-1}$
• For many nodes, take limit of $NP^*(1-P^*)^{N-1}$
• As $N$ goes to infinity, gives: $\frac{1}{e} = .37$

• At best: channel used for useful transmissions 37% of time!
Pure ALOHA

- Unslotted Aloha: simpler, no synchronization
- When frame first arrives
  - Transmit immediately
- Collision probability increases:
  - Frame sent at $t_0$ collides with other frames sent in $[t_0-1,t_0+1]$

Pure ALOHA Efficiency

\[ P(\text{success by given node}) = P(\text{node transmits}) \]
\[ P(\text{no other node transmits in } [t_0-1,t_0]) = p \cdot (1-p)^{2(N-1)} \]

… choosing optimum \( p \) and then letting \( n \to \infty \)
\[ = 1 / (2e) = 0.18 \]

even worse than slotted Aloha!
CSMA (Carrier Sense Multiple Access)

**CSMA:** listen before transmit:

- **If channel sensed idle:** transmit entire frame
- **If channel sensed busy,** defer transmission
- Human analogy: do not 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
  • Distance & propagation delay play role in determining collision probability
CSMA 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
CSMA Collision Detection

- Human analogy
  - The polite conversationalist
 Ethernet CSMA/CD Algorithm

• NIC receives datagram from network layer, creates frame
• If NIC senses channel idle, starts frame transmission. If NIC senses channel busy, waits until channel idle, then transmits.
• If NIC transmits entire frame without detecting another transmission, NIC is done with frame!
• If NIC detects another transmission while transmitting, aborts and sends jam signal
• After aborting, NIC enters **binary (exponential) backoff**:  
  • after $m$th collision, NIC chooses $K$ at random from $\{0, 1, 2, \ldots, 2^m - 1\}$.  
    • NIC waits $K \cdot 512$ bit times, returns to Step 2  
    • Longer backoff interval with more collisions
Taking-Turns MAC Protocols

• Channel partitioning MAC protocols:
  • Share channel **efficiently and fairly** at high load
  • Inefficient at low load: delay in channel access, 1/N bandwidth allocated even if only 1 active node!

• Random access MAC protocols
  • Efficient at low load: single node can fully utilize channel
  • High load: collision overhead

• They look for the best of both worlds!
Taking-Turns MAC Protocols

Polling
• Master node **invites** slave nodes to transmit in turn
• Typically used with dumb slave devices
• Concerns
  • Polling overhead
  • Latency
  • Single point of failure (master)
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)
Link Layer

• Error detection, correction
• Multiple access protocols
✓ LANs
  ✓ Addressing, ARP
  • Ethernet
  • Switches
  • VLANS
• Link virtualization: MPLS
• Data center networking
MAC Addresses & ARP

• 32-bit IP address:
  • Network-layer address for interface
  • Used for layer 3 (network layer) forwarding

• MAC (or LAN or physical or Ethernet) address:
  • Function: used locally to get frame from one interface to another physically-connected interface (same network, in IP-addressing sense)
  • 48 bit MAC address (for most LANs) burned in NIC ROM, also sometimes software settable
  • E.g.: 1A-2F-BB-76-09-AD
MAC Addresses & ARP

• Each adapter on LAN has unique LAN address
LAN Address

• MAC address allocation administered by IEEE
• Manufacturer buys portion of MAC address space (to assure uniqueness)

• Analogy
  • MAC address: like Social Security Number
  • IP address: like postal address
• MAC flat address → portability
  • Can move LAN card from one LAN to another
• IP hierarchical address not portable
  • Address depends on IP subnet to which node is attached
**ARP: Address Resolution Protocol**

**Question:** how to determine interface’s MAC address, knowing its IP address?

**ARP table:** each IP node (host, router) on LAN has 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

• A wants to send datagram to B
  • B’s MAC address not in A’s ARP table.

• A broadcasts ARP query packet, containing B’s IP address
  • Destination MAC address = FF-FF-FF-FF-FF-FF
  • All nodes 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)
ARP Protocol: Same LAN

• 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**
- Focus on addressing – at IP (datagram) and MAC layer (frame)
- Assume A knows B’s IP address
- Assume A knows IP address of first hop router, R (how?)
- Assume A knows R’s MAC address (how?)

![Diagram showing network topology with nodes A, R, and B, and their respective IP and MAC addresses.](image-url)
Addressing: Routing to Another LAN

- A creates IP datagram with IP source A, destination B
- A creates link-layer frame with R’s MAC address as destination address, frame contains A-to-B IP datagram
Addressing: Routing to Another LAN

• Frame sent from A to R
• Frame received at R, datagram removed, passed up to IP
Addressing: Routing to Another LAN

- R forwards datagram with IP source A, destination B
- R creates link-layer frame with B’s MAC address as destination address, frame contains A-to-B IP datagram
Addressing: Routing to Another LAN

• R forwards datagram with IP source A, destination B
• R creates link-layer frame with B's MAC address as dest, frame contains A-to-B IP datagram
Link Layer

• Error detection, correction
• Multiple access protocols
✓ LANs
  • Addressing, ARP
  ✓ Ethernet
  • Switches
  • VLANS
• Link virtualization: MPLS
• Data center networking
Ethernet

• Dominant wired LAN technology
  • Single chip, multiple speeds (e.g., Broadcom BCM5761)
  • First widely used LAN technology
  • Kept up with speed race: 10 Mbps – 10 Gbps

• **Bus**: popular through mid 90s
  • All nodes in same collision domain (can collide with each other)

• **Star**: prevails today
  • Active **switch** in center
  • Each **spoke** runs a (separate) Ethernet protocol (nodes do not collide with each other)
Ethernet: Physical Topology

- **Bus:** popular through mid 90s
  - All nodes in same collision domain (can collide with each other)
- **Star:** prevails today
  - Active *switch* in center
  - Each *spoke* runs a (separate) Ethernet protocol (nodes do not collide with each other)
Ethernet Frame Structure

• Sending adapter encapsulates IP datagram (or other network layer protocol packet) in **Ethernet frame**

**Preamble:**
• 7 bytes with pattern 10101010 followed by one byte with pattern 10101011
• Used to synchronize receiver, sender clock rates
Ethernet Frame Structure

- **Addresses**: 6 byte source, destination MAC addresses
  - If adapter receives frame with matching destination address, or with broadcast address (e.g. ARP packet), it passes data in frame to network layer protocol
  - Otherwise, adapter discards frame

- **Type**: Indicates higher layer protocol (mostly IP but others possible, e.g., Novell IPX, AppleTalk)
- **CRC**: Cyclic redundancy check at receiver
  - Error detected: frame is dropped
Ethernet: Unreliable, Connectionless

- **Connectionless**: No handshaking between sending and receiving NICs

- **Unreliable**: receiving NIC doesn't send acks or NACKs to sending NIC
  - Data in dropped frames recovered only if initial sender uses higher layer RDT (e.g., TCP), otherwise dropped data lost

- Ethernet’s MAC protocol: Unslotted **CSMA/CD with binary backoff**
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, 10 Gbps, 40 Gbps
  • Different physical layer media: fiber, cable
Link Layer

- Error detection, correction
- Multiple access protocols

✓ **LANs**
  - Addressing, ARP
  - Ethernet

✓ **Switches**
  - VLANS

- Link virtualization: MPLS
- Data center networking
Ethernet Switch

• **Link-layer device: takes an 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: 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’ can transmit simultaneously, without collisions

---

Switch Forwarding Table

Q: how does switch know 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
Switch: Frame Filtering/Forwarding

When frame received at switch

- Record incoming link, MAC address of sending host
- Index switch table using MAC destination address
- If entry found for destination

```java
then {
  if destination on segment from which frame arrived then drop frame
  else forward frame on interface indicated by entry
}
else flood /* forward on all interfaces except arriving interface */
```
Self-Learning, Forwarding: Example

• Frame destination, A’, location unknown: **flood**
• Destination A location known: **selectively send on just one link**

<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>
<tr>
<td>A’</td>
<td>4</td>
<td>60</td>
</tr>
</tbody>
</table>

Switch table (initially empty)
Interconnecting Switches

• Self-learning switches can be connected together:

Q: sending from A to G - how does $S_1$ know to forward frame destined to G via $S_4$ and $S_3$?
A: self learning! (works exactly the same as in single-switch case!)
Switches vs. Routers

Both are store-and-forward:

- **Routers**: network-layer devices (examine network-layer headers)
- **Switches**: link-layer devices (examine link-layer headers)

Both have forwarding tables:

- **Routers**: compute tables using routing algorithms, IP addresses
- **Switches**: learn forwarding table using flooding, learning, MAC addresses
Link Layer

• Error detection, correction
• Multiple access protocols

✓ **LANs**
  • Addressing, ARP
  • Ethernet
  • Switches

✓ **VLANS**
  • Link virtualization: MPLS
  • Data center networking
VLANs: Motivation

Consider

• CS user moves office to EE, but wants connect to CS switch?

• Single broadcast domain:
  • All layer-2 broadcast traffic (ARP, DHCP, unknown location of destination MAC address) must cross entire LAN
  • Security/privacy, efficiency issues
VLANs

Virtual Local Area Network

• Switch(es) supporting VLAN capabilities can be configured to define multiple virtual LANS over single physical LAN infrastructure.

• Port-based VLAN: switch ports grouped (by switch management software) so that single physical switch ......
VLANs

... operates as multiple virtual switches
Port-Based VLAN

- **Traffic isolation**: frames to/from ports 1-8 can only reach ports 1-8
  - Can also define VLAN based on MAC addresses of endpoints, rather than switch port

- **Dynamic membership**: ports can be dynamically assigned among VLANs

- **Forwarding between VLANS**: done via routing (just as with separate switches)
  - In practice vendors sell combined switches plus routers

![Diagram of port-based VLAN with traffic flow and router integration]
• **Trunk port**: carries frames between VLANs defined over multiple physical switches
  * Frames forwarded within VLAN between switches can’t be vanilla 802.1 frames (must carry VLAN ID info)
  * 802.1q protocol adds/removed additional header fields for frames forwarded between trunk ports
802.1Q VLAN Frame Format

![Diagram of 802.1Q VLAN Frame Format]

- **preamble**
- **dest. address**
- **source address**
- **data (payload)**
- **CRC**
- **2-byte Tag Protocol Identifier (value: 81-00)**
- **Tag Control Information (12 bit VLAN ID field, 3 bit priority field like IP TOS)**
- **R recomputed CRC**

802.1 frame

802.1Q frame
Link Layer

• Error detection, correction
• Multiple access protocols
• LANs
  • Addressing, ARP
  • Ethernet
  • Switches
  • VLANS

✓ Link virtualization: MPLS
• Data center networking
MPLS

• Multiprotocol Label Switching
• Initial goal: high-speed IP forwarding using fixed length label (instead of IP address)
  • Fast lookup using fixed length identifier (rather than shortest prefix matching)
  • Borrowing ideas from Virtual Circuit (VC) approach
  • But IP datagram still keeps IP address!
MPLS Capable Routers

• A.k.a. label-switched router

• Forward packets to outgoing interface based only on label value (don’t inspect IP address)
  • MPLS forwarding table distinct from IP forwarding tables

• **Flexibility:** MPLS forwarding decisions can differ from those of IP
  • Use destination and source addresses to route flows to same destination differently (traffic engineering)
  • Re-route flows quickly if link fails: pre-computed backup paths (useful for VoIP)
MPLS Versus IP Paths

- **IP routing**: path to destination determined by destination address alone
MPLS Versus IP Paths

- **IP routing**: path to destination determined by destination address alone
- **MPLS routing**: path to destination can be based on source and destination address
  - **Fast reroute**: precompute backup routes in case of link failure
MPLS Signaling

- Modify OSPF, IS-IS link-state flooding protocols to carry info used by MPLS routing
  - E.g., link bandwidth, amount of **reserved** link bandwidth
- Entry MPLS router uses RSVP-TE signaling protocol to set up MPLS forwarding at downstream routers
MPLS Forwarding Tables

<table>
<thead>
<tr>
<th>in label</th>
<th>out label</th>
<th>dest</th>
<th>out interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>A</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>D</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>A</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>in label</th>
<th>out label</th>
<th>dest</th>
<th>out interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>6</td>
<td>A</td>
<td>1</td>
</tr>
<tr>
<td>12</td>
<td>9</td>
<td>D</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>in label</th>
<th>out label</th>
<th>dest</th>
<th>out interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>6</td>
<td>A</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

R6
R5
R4
R3
R2
D
A
Link Layer

- Error detection, correction
- Multiple access protocols
- LANs
  - Addressing, ARP
  - Ethernet
  - Switches
  - VLANs
- Link virtualization: MPLS

✓ Data center networking
Data Center Networks

• 10’s to 100’s of thousands of hosts, often closely coupled, in close proximity
  • E-business (e.g. Amazon)
  • Content-servers (e.g., YouTube, Akamai, Apple, Microsoft)
  • Search engines, data mining (e.g., Google)

• Challenges
  • Multiple applications, each serving massive numbers of clients
  • Managing/balancing load, avoiding processing, networking, data bottlenecks

Inside a 40-ft Microsoft container, Chicago data center
Data Center Networks

- **Load balancer**: application-layer routing
  - Receives external client requests
  - Directs workload within data center
  - Returns results to external client
    (hiding data center internals from client)
Data Center Networks

• Rich interconnection among switches, racks:
  • Increased throughput between racks (multiple routing paths possible)
  • Increased reliability via redundancy
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, VLANs
  • Virtualized networks as a link layer: MPLS

• Synthesis: a day in the life of a web request
Acknowledgements

• The following materials have been used in preparation of this slide set:

7th Edition
James Kurose, Keith Ross
Pearson
2016

5th Edition
Larry Peterson, Bruce Davie
Morgan Kaufmann
2011