**Link Layer and Local Area Networks (LANs)**

**Link Layer: Introduction**

For the Link Layer, we focus within a Subnet.

Terminology

* **Nodes:** Hosts and Routers
* **Links**: Communication channels that connect adjacent nodes along the path: Wired links, Wireless links, LANs
* **Layer-2 packet**: Frame, encapsulates datagram

The **Data-Link Layer** has responsibility of transferring datagram from one nod to a **physically adjacent** node over a link.

**Link Layer: Context**

Datagrams are transferred by different link protocols over different links:

* 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 over links

Travel analogy:

* Tourist = **datagram**
* Transport segment (location) = **communication link**
* Transportation mode (train, bus, airplane) = **link layer protocol**
* Travel agent = **routing algorithm**

**Link Layer: Services**

**Frame, link access**

* Encapsulates datagram into a frame, adding header and trailer (last bytes at end of block for error checking).
* Channel access if it’s a shared medium
* MAC addresses used in frame headers to identify the SOURCE/DEST (different from IP addresses)

**Reliably delivery between adjacent nodes**

* Recall reliable delivery in previous topics
* RDT service is seldom used low bit-error link
* High error rates in wireless links.

**Flow Control**

* Pacing between adjacent sending and receiving nodes

**Error Detection**

* Errors caused by signal attenuation (reduction of signal strength during transmission) and noise.
* Receiver detects presence of errors: signals sender for retransmission or drops the frame.

**Error Correction**

* Receiver identifies and corrects bit errors without resorting to re-transmission.

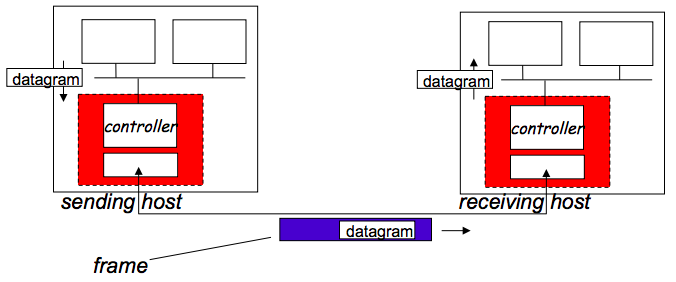
**Half-Duplex and Full-Duplex**

* Half-Duplex: nodes at both ends of link can retransmit, but not at the same time.

Where is the link layer implemented?

* In each and every host.
* Link layer is implemented in a **network interface card (adaptor)** or on a chip.
  + Ethernet card, 802.11 card, Ethernet chipset.
  + Implements the link, physical layer.
* Attaches into a host’s system bus
* Combination of hardware, software, firmware

Adaptors communicating:

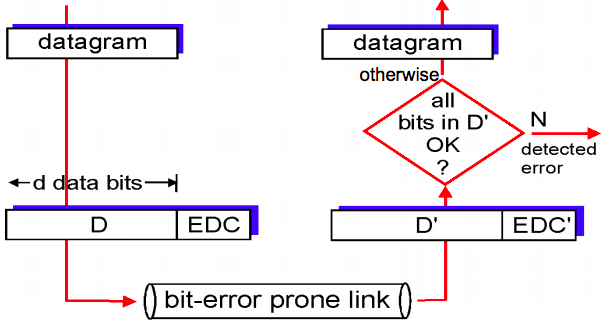


**Sending side:**

* Encapsulates datagram in frame
* Adds error checking bits, rdt, flow control etc.

**Receiving side:**

* Looks for errors, rdt, flow control
* Extracts datagram, passes to upper layer at receiving side.

**Error Correction**

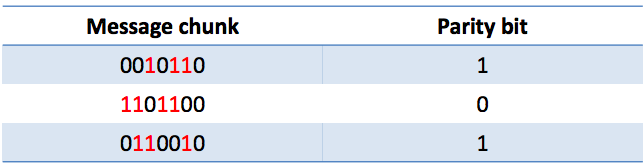
**EDC = Error Detection and Correction bits** (redundancy)  
**D = Data protected by error checking, may include error fields**

Error detection is not 100% reliable:

* Protocol may miss some errors (rare though)
* Larger EDC field yields better detection and correction

**Simple Parity – Sender**

* For every d\_bits add a parity bit:
  + Parity bit = 1 if the number of one’s are odd
  + Parity bit = 0 if the number of one’s are even
* Example (where d=7):

 result = 0010110**1**1101100**0**0110010**1**

**Simple Parity – Receiver**

* For each block of size d\_bits, count # of ones and compare with the following bit parity.
* If an ODD number of bits get flipped, it will be detected
* Cost: One extra bit for every d\_bits (in this example, 21 🡪 24 bits)

**Two Dimensional Parity**

* On top of Parity Bits, add an extra PARITY BYTES + compute parity on columns too.
* Can detect 1,2,3-bit (and some 4-bit) errors.

