**I. SERVICES**:

* **Package/ truy cập kênh**: Nhận từ network layer, package vào frame (add header và trailer) rồi chuyển xuống physical layer/ quản lý việc access vào kênh truyền (kênh chung như LAN nơi có nhiều devices muốn concurrently access).
* **Reliable delivery:** rdt in chapter 3, sử dụng nhiều ở wireless vì nhiều bugs hơn wire => sử dụng reliable delivery
* **Flow control**: control rate giữa 2 các thiết bị kết nối trực tiếp -> không bị overload
* **Error detection and correction** (EDC)
* **Parity checking** (kiểm tra chẵn lẻ): thêm 1 bit để mark parity (có thể là 1/ 2 dimensions, có thể là là odd/even dạng)
* **Internet checksum:** chỉ được dùng ở transport layer (TCP/UDP) -> tính tổng -> cộng thêm redundant bit -> ra sum -> đảo bit -> ra checksum
* **CRC** (Cyclic redundancy check): -> xác định bao nhiêu redundant bits -> xor đến chết.

**II. GIAO THỨC ĐA TRUY CẬP:** phân loại theo MAC:

* **Chia kênh**: TDMA và FDMA
* **Truy cập ngẫu nhiên**:
* Slotted ALOHA: Truyền trong khe thời gian cố định nếu phát hiện collision -> truyền lại đến khi thành công
* Pure/ unslotted ALOHA: Truyền ngay khi có dữ liệu: bị overlap giữa frame t0 - 1 và t0 + 1 của frame gửi vào thời điểm t0
* CSMA (có chữ S tức sense) -> lắng nghe -> tránh đụng độ
* CSMA/CD: (cũng có chữ S nhưng CD) -> lắng nghe -> phát hiện collision sớm
* Algo của CSMA/CD: package -> sense -> send -> collision -> cancel ->binary (exponential) backoff tức là wait và re-send
* **Xoay vòng**:
* Polling: master node mời slave node truyền respectively.
* Chuyển token: chuyển token giữa các node

**III. MAC, ETHERNET, SWITCH:** 3 cái này sử dụng cục bộ (tức là cùng IP) để chuyển frame từ 1 interface (cổng vật lý) này sáng 1 interface khác

* **ARP** (address resolution protocol):
* Là 1 cái bảng trong mỗi host, router (router còn có forwarding table)
* Ở trong mạng LAN
* Dùng để map giữa IP và MAC
* Có TTL check xem cái **<IP, MAC, TTL>** trong table đã expire chưa
* **Cách hoạt động của ARP:** thì node muốn gửi sẽ biết IP des (của node gửi) và src (của node nhận) và cả MAC src (của node gửi), nếu có MAC des rồi thì chỉ việc gửi, còn không thì đi tìm MAC des:

1. Đầu tiên, A broadcasts cho **TẤT CẢ** các node trong LAN với MAC des là FF-FF-FF-FF-FF-FF (vì đã biết MAC des đâu)
2. Sau đó, node nào trong những node được gửi có thể là B, ok B sẽ nhận ARP packet rồi sẽ gửi cho **MỖI** mình A thông tin về MAC des.

* **NOTE**: còn khác subnet thì IP des và src trong quá trình truyền luôn giữ nguyên, MAC des và src thay đổi qua lại ở router, vì router chia làm 2 khu vực LAN cách hoạt động tương tự
* **Ethernet:** hình bus thì dễ collision -> sử dụng hình star, cấu trúc của frame ethernet gồm:
* Preamble: 7 bytes 10101010 theo sau 1 byte 10101011 dùng để sync rate
* Addresses: 6 bytes MAC src và des
* Type: chỉ ra protocol của tầng trên
* CRC: dùng để check lỗi
* **NOTE**: ethernet thì unreliable (lỗi là bỏ luôn chứ không ack hay nack) và connectionless (không shakehand) và CSMA/CD (binary backoff)
* **Switch:** thiết bị ở tầng link
* Lưu và chuyển tiếp các frame ethernet
* Chọn lựa các frame để forward (việc forward này sử dụng CSMA/CD)
* Mấy cái switch này thì host coi như không tồn tại
* Switch cho phép truyền đồng thời, sử dụng full-duplex, **mỗi cái đường truyền là 1 collision domain của chính nó (tức có thể truyền đồng thời như A-A’ và B-B’ cùng lúc nhưng A-A’ và C-A’ là tạch)**
* **Switch tự học:** để biết máy (host) nào kết nối tới cổng (interface) nào?
* Kiểu **<MAC, cổng, TTL>**
  1. Đầu tiên, khi nhận frame từ 1 host nào đó, nó ghi vào forwarding table (switch cũng có cái table này như router) ví dụ <địa chỉ MAC A, cổng 1, 60>
  2. Check trong forwarding table xem có địa chỉ MAC des (để mò ra cổng nào có cái MAC des đó) chưa? Có 2TH là:
* TH1: Không có cái thông tin nào về MAC des (lưu ý ở đây đang truyền frame ethernet nên sẽ có thông tin về MAC des) => flood (gửi frame đến tất cả interface trừ interface nhận vào) => Gửi frame thực hiện tất cả các bước => có được thông tin về MAC và cổng des
* TH2: Tìm thấy thông tin trong forwarding table:
* TH2.1: cái MAC des nằm trên phân đoạn mạng từ frame gửi đến (tức là nằm trong chung 1 cái cổng) => Bỏ frame đó đi không cần sử dụng switch để forward
* TH2.2: chuyển frame đến interface (cổng) trong forwarding table tương ứng với MAC