


# **Chapter 2**

## **네트워크 모델**

### **(Network Models)**

Two horizontal decorative bars are located below the title. The top bar is a solid dark green color, and the bottom bar is a solid light green color. Both bars span the entire width of the slide.

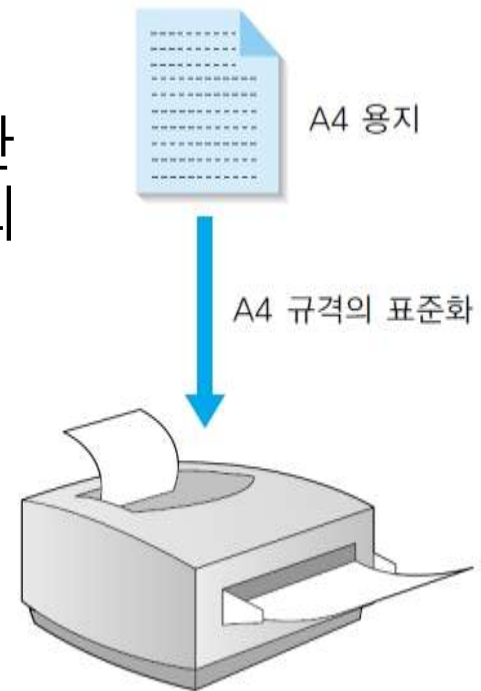
## ■ 네트워크 모델 개념과 TCP/IP 프로토콜 소개

- ✓ 네트워크 표준화
- ✓ 프로토콜 계층화 개념
- ✓ OSI 모델 소개
- ✓ TCP/IP 프로토콜 그룹

# § 1. 네트워크 표준화

## ■ 배경

- ✓ 정보통신기술은 자사의 모델을 중심으로한 통신 위주로 발전 → 타사 제품 컴퓨터 간의 통신문제 대두
- ✓ 네트워크 사용의 효율성을 높이고 컴퓨터 네트워크 구조를 통일시키는 것이 필요



## ■ 1983년 ISO에서 OSI 모형을 제시

- ✓ ISO (International Standardization Organization)  
국제표준화기구
- ✓ OSI (Open Systems Interconnection) 개방형 시스템간 상호접속
  - 서로 다른 통신 시스템간에 원활한 통신을 이루기 위해 ISO가 제안한 통신 H/W, S/W에 대한 표준안

## § 2. 프로토콜과 프로토콜 계층화

---

### ■ 프로토콜

- ✓ 통신에 참여하는 장치들이 효과적인 통신을 위해 지켜야 할 규칙
  - 송신기, 수신기, 중간 노드들
- ✓ 통신 프로토콜
  - 통신 원하는 두 개체 간에 정보교환을 위해 정한 통신 규칙
  - 상호 간의 접속이나 절단 방식, 통신 방식, 자료의 형식, 오류 검출 방식, 코드 변환 방식, 전송 속도 등

# 프로토콜: 첫 번째 시나리오

## ■ 공통적인 생각을 가진 이웃 간의 의사전달

✓ 단일 계층에서만 이루어짐



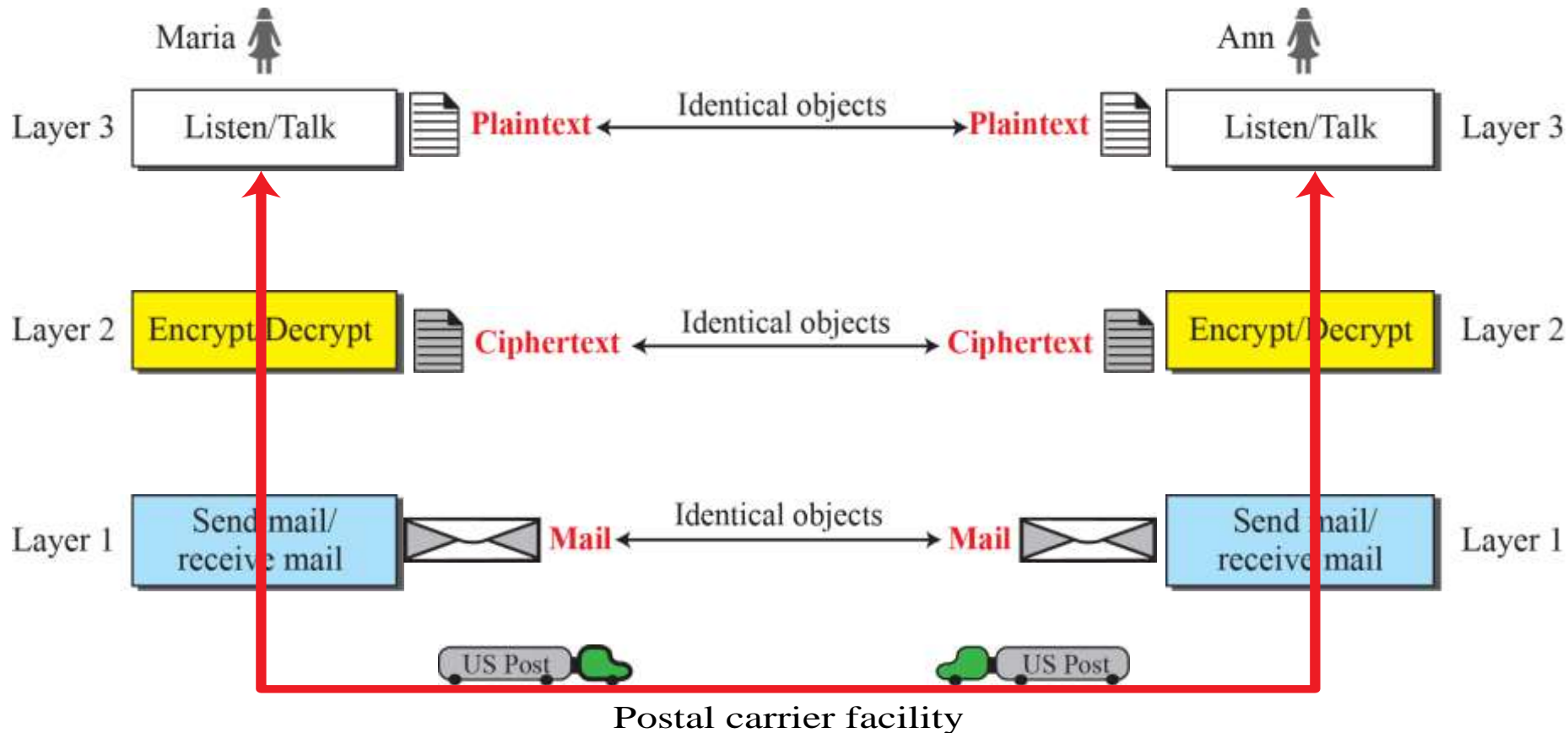
✓ 규칙

- 만날 때 서로 인사
- 서로 간에 사용하는 단어를 제한(친구간,..)
- 상대방이 말할 때 자신은 경청
- 독백이 아니라 대화식으로 대화
- 헤어질 땐 기분 좋은 단어를 교환

# 프로토콜: 두 번째 시나리오

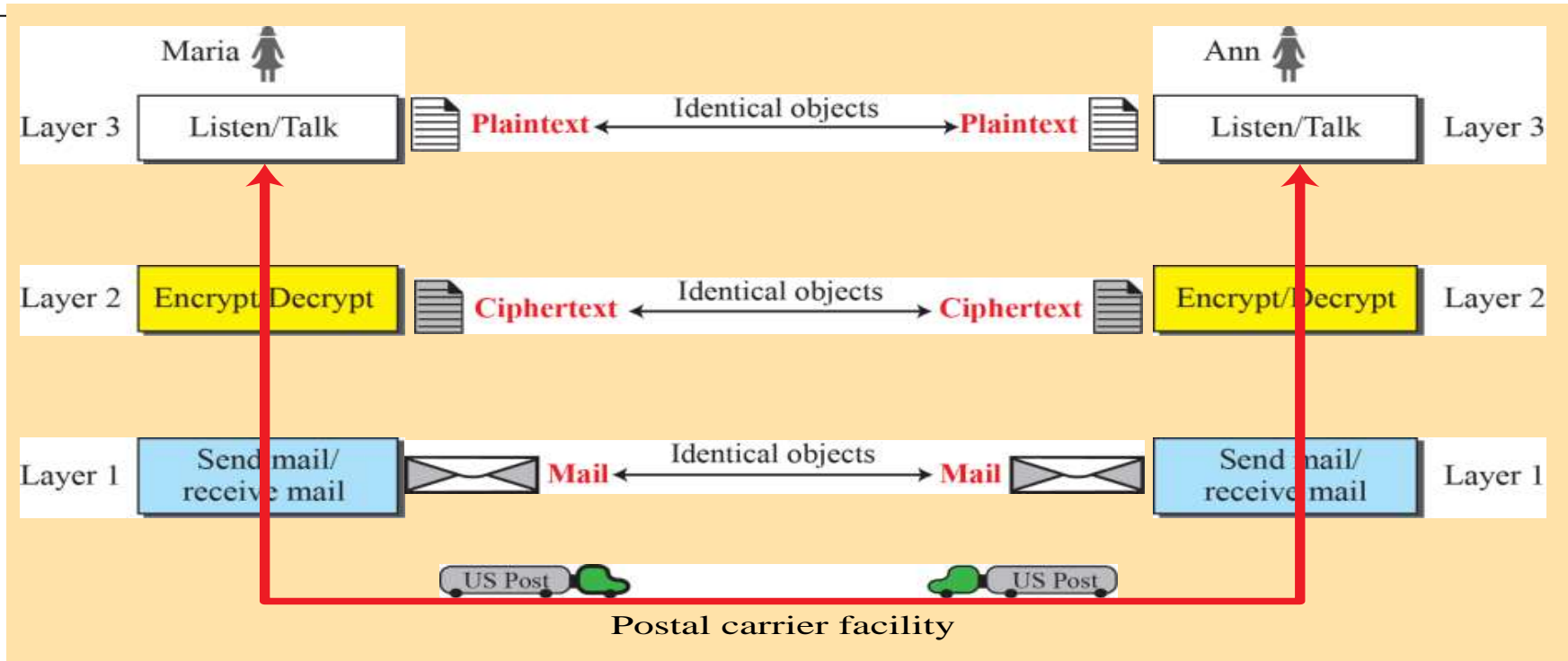
## ■ 먼 지역에 거주하는 친구와의 의사전달

- ✓ 이 통신이 3계층에서 이루어진다고 가정
- ✓ 각 층의 작업 처리하는 기계(로봇)가 있다고 가정



< 계층 프로토콜 >

# 프로토콜의 계층화의 특징



- 중간 노드에서 모든 계층을 다 사용하지 않을 수도 있음

## ■ 서비스를 구현으로 부터 분리 가능

- ✓ 한 계층은 하위 계층으로부터 서비스를 받고, 상위 계층에게 서비스를 제공해야 함
- ✓ 그 계층이 어떻게 구현되는지는 신경 쓰지 않아도 됨

# 프로토콜의 계층화의 원칙

---

## ■ 프로토콜 계층화의 원칙

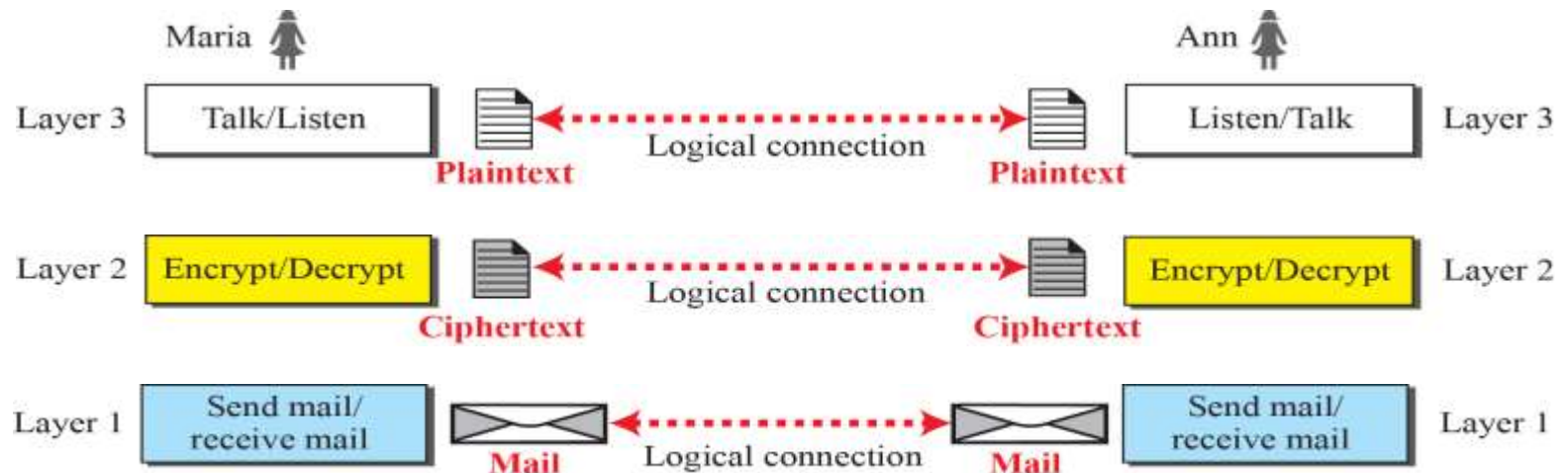
- ✓ **원칙1**: 양방향 통신이라면, 각 계층은 각 방향으로 한 가지씩, 상반되는 두가지 기능 수행해야 함
  - 예: 말하기/듣기(쓰기/읽기), 암호화/복호화, 보내기/받기
  
- ✓ **원칙2**: 양측의 각 계층의 객체는 동일
  - 예: 평문 처리, 암호문 처리, 편지



# 대등 대 대등 프로세스(Peer-to-Peer Process)

## ■ 논리적인 연결

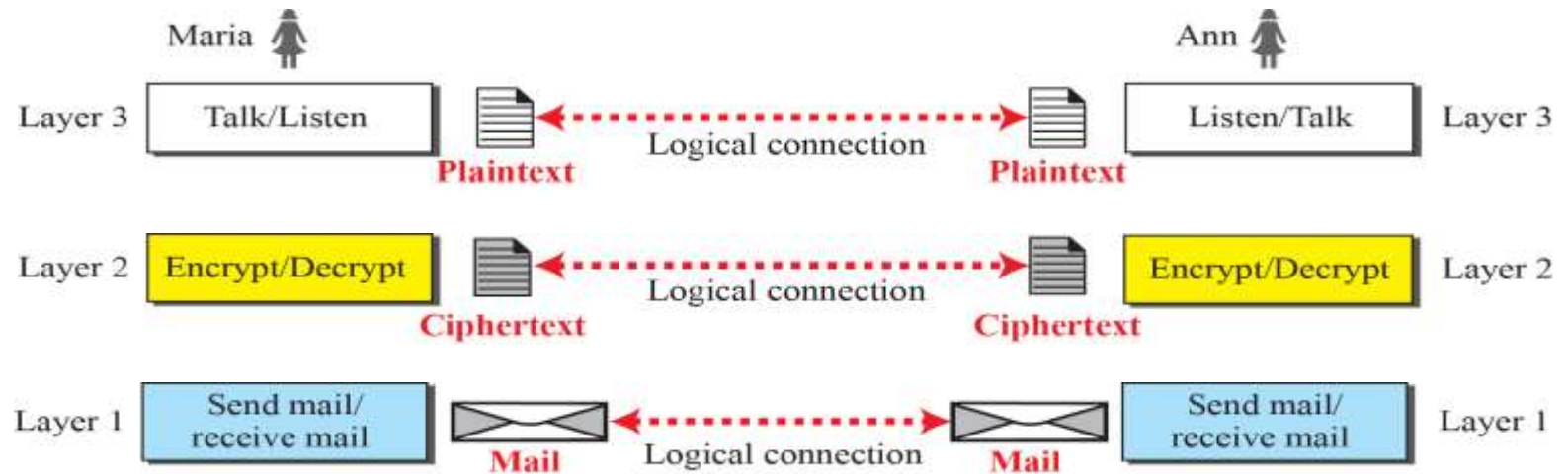
- ✓ 계층-대-계층 통신을 갖는다는 의미; 한 장치의 x번째 계층은 다른 장치의 x번째 계층과 통신



- ✓ 장치들 사이에서 프로토콜이라는 협의된 규칙과 규약에 의해 제어

## ■ 프로토콜과 인터페이스

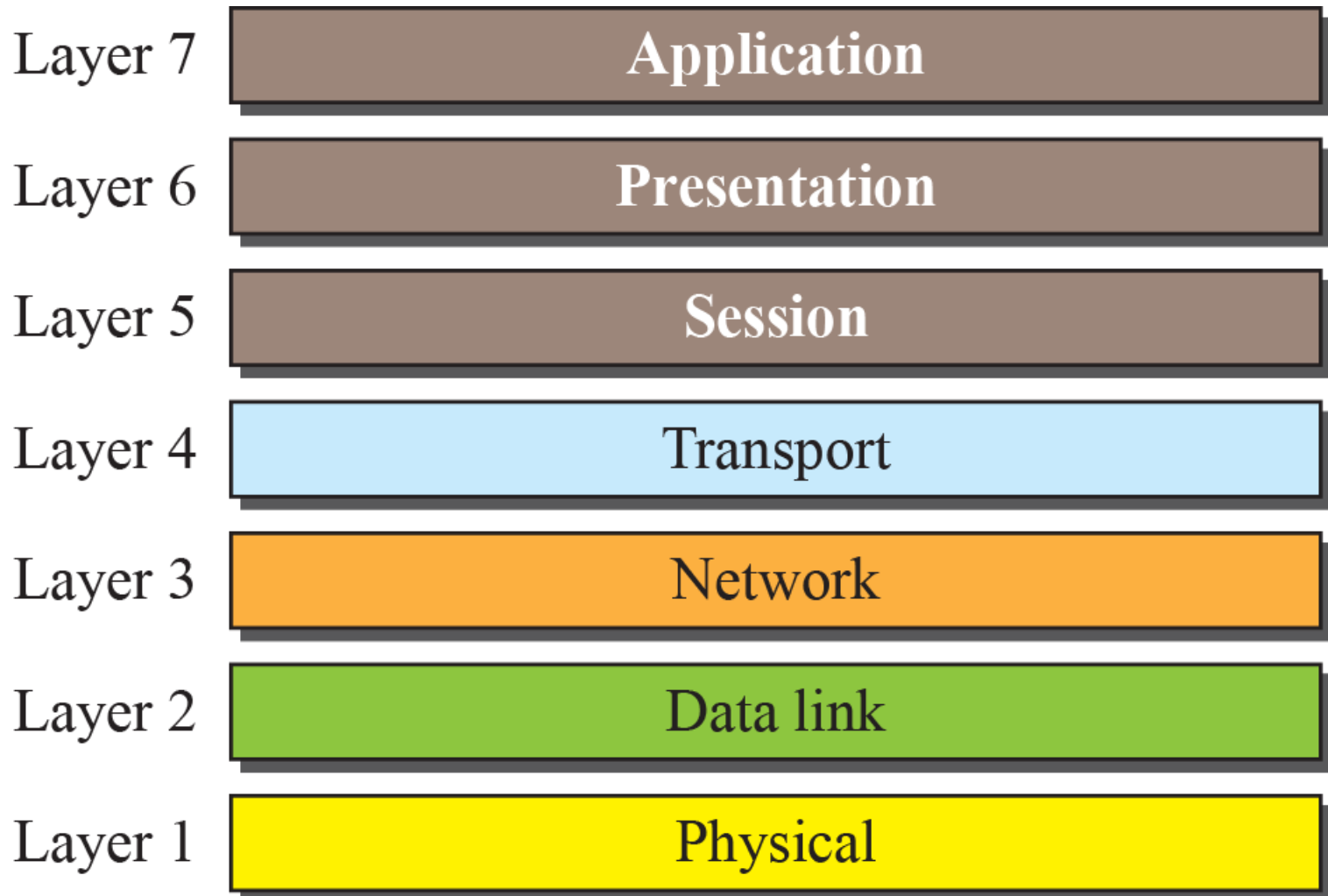
- ✓ 프로토콜: 서로 다른 호스트에 위치한 동일계층간의 통신 규칙
- ✓ 인터페이스(접점): 같은 호스트에 위치한 상하위계층 간의 규칙; 한 계층이 인접한 계층에게 제공해야 하는 정보와 서비스
- ✓ 서비스: 하위 계층이 상위 계층에 제공하는 인터페이스



## § 3. OSI 모델

- OSI (Open System Interconnection) 개방시스템 상호연결
  - ✓ 국제표준기구(ISO)에 의해 설계
    - International Organization for Standard
    - 국제 표준을 제정하는 다국적 기관
    - 1970년 후반에 처음 소개
    - TCP/IP가 자리잡은 뒤 OSI가 완성
  - ✓ 서로 다른 시스템 간의 통신을 허용하는 네트워크 시스템을 설계하기 위한 목적
- OSI Model은 TCP/IP이후에 규격화 됨 → OSI 모델의 실패
  - ✓ **이유1:** OSI는 TCP/IP가 자리잡고 난 이후 완성됨. 다른 모델로의 변경이 어려워 졌음.
  - ✓ **이유2:** OSI 모델의 일부는 완전히 정의되지 않았음.
  - ✓ **이유3:** TCP/IP측에서 볼 때, 구현된 OSI가 높은 수준의 성능을 보여주지 못하여, OSI로 변환시킬 매력을 못 느낌.

# OSI 모델의 계층구조

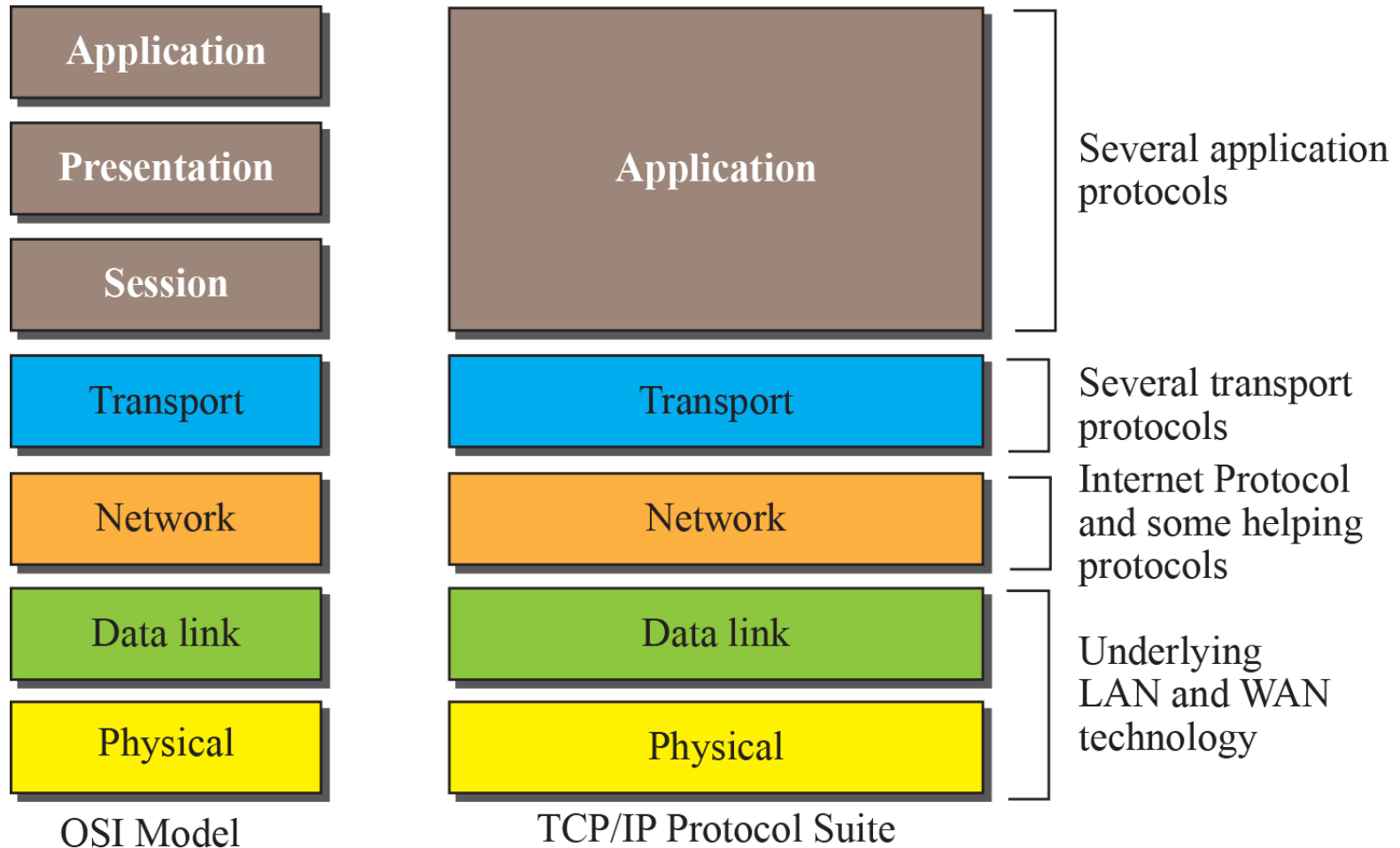


# OSI 모델의 계층별 기능 요약

계층	명칭	기능
7	응용 계층	응용에 대한 서비스 제공, 프로세스간의 정보교환(ftp, e-mail, telnet)
6	표현 계층	문자코드, 데이터압축, 암호화 등의 데이터 변환 ↓
5	세션 계층	세션의 설정,관리,종료 및 Dialog Control, Synchronization ■ 목적지주소
4	전송 계층	데이터 전달 보장. 분할과 조립, 다중화, 분리, 흐름제어, 에러제어 ■ 패킷 ■ 패킷번호
3	네트워크 계층	경로설정 및 논리주소 결정 ■ 에러제어
2	데이터 링크 계층	네트워크 기기간의 데이터 전송 및 물리주소 결정, 흐름제어, 에러제어, 정보프레임화 ■ 플래그
1	물리계층	물리적 연결과 전기신호를 변환 및 제어 (부호화, 변조, 복조, 신호전파)

# OSI vs. TCP/IP

- TCP/IP 프로토콜에는 세션층과 표현층이 없음

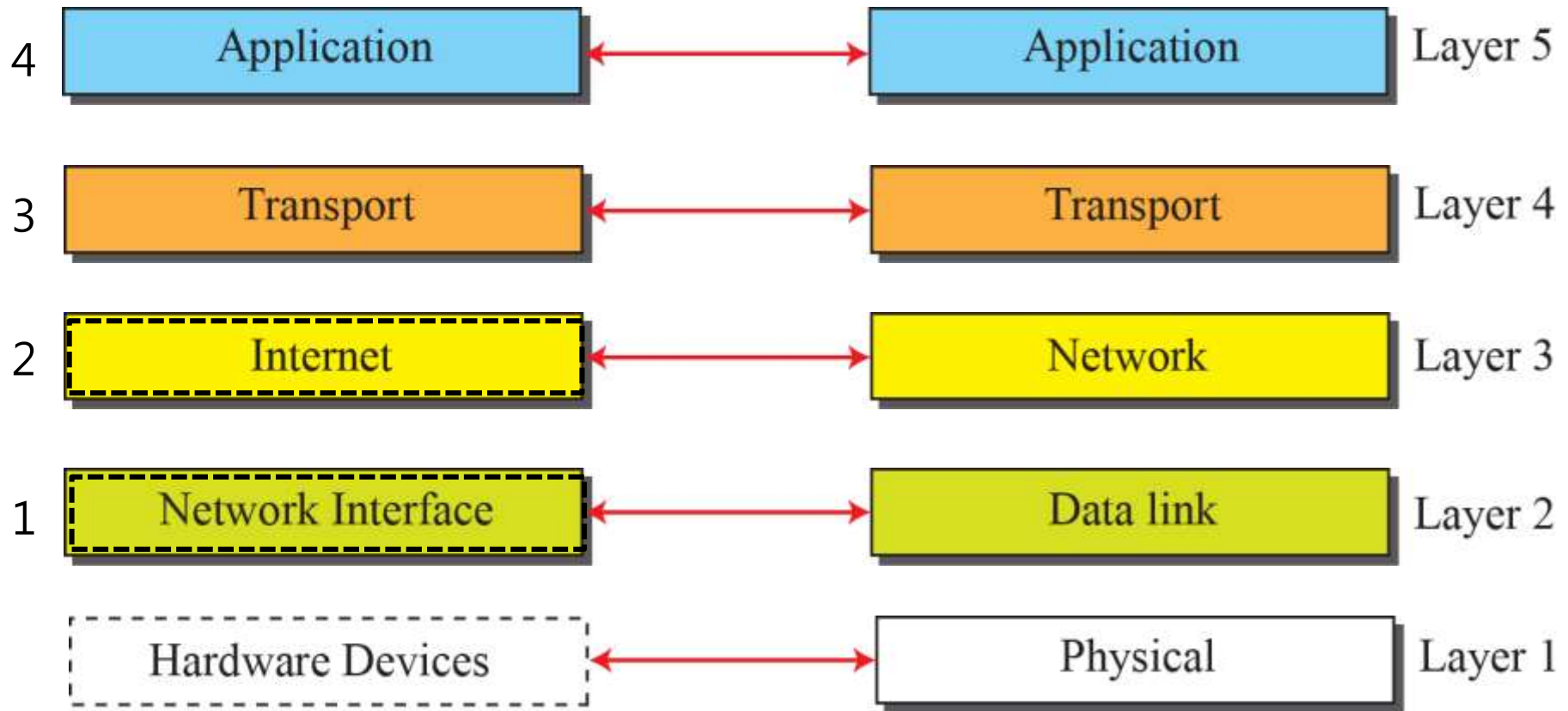


## § 4. TCP/IP 모델

---

- 서로 다른 다수의 망을 통하여 통신하기 위한 프로토콜의 집합
- 현재 인터넷에서 사용하는 프로토콜 그룹
  - ✓ TCP/IP (Transmission Control Protocol / Internetworking Protocol)
  - ✓ 여러 계층에서 조직된 프로토콜들의 집합
  - ✓ 상호 작용하는 모듈로 이루어진 계층적 프로토콜

# TCP/IP 프로토콜 계층



< Original layers >

→ Hardware에 설치된  
4개의 software 계층

< Layers used at present >

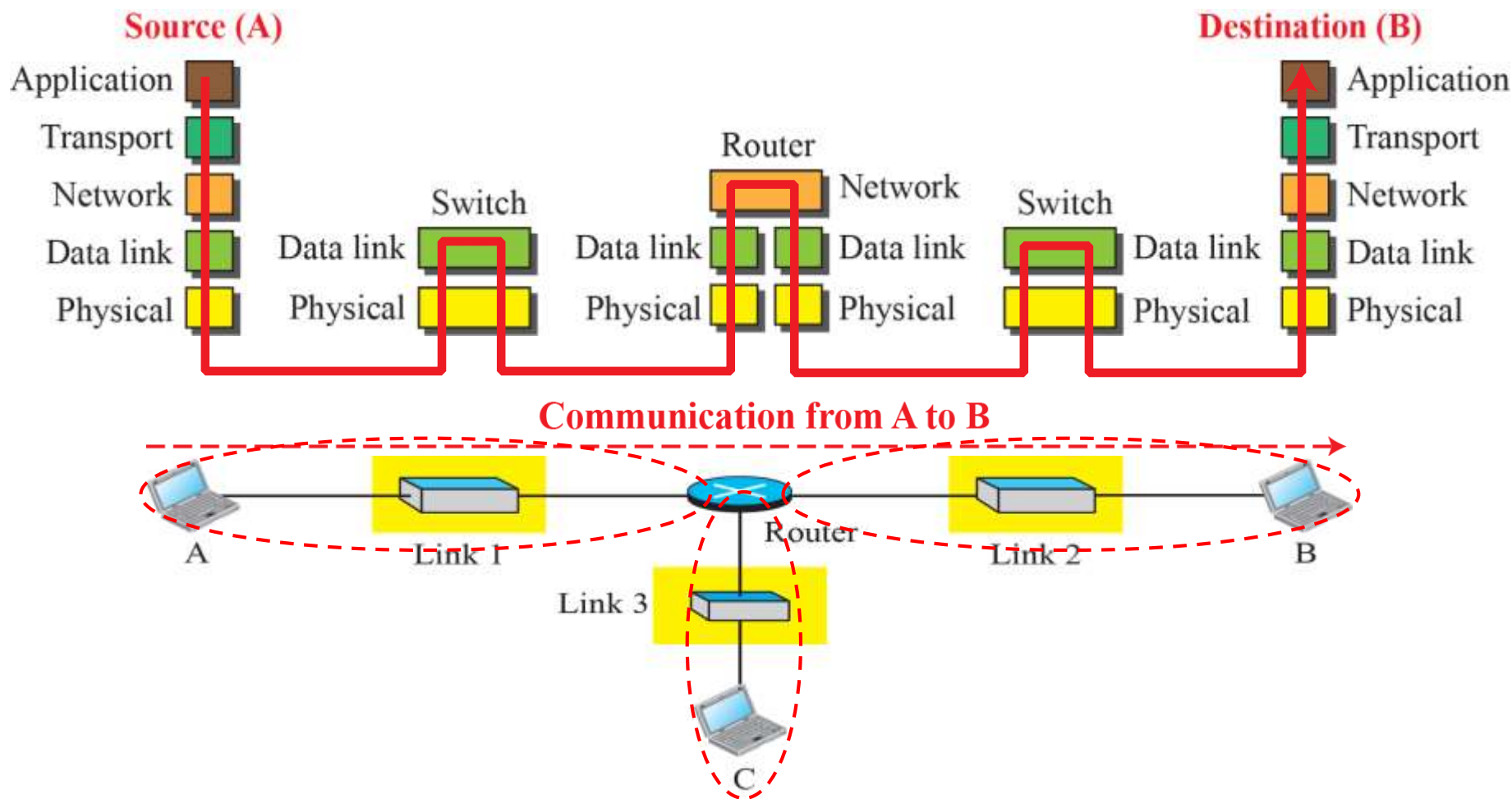
5개의 계층



# 계층적 구조에서의 통신 예

## ■ 3개의 LAN으로 이루어진 인터넷을 가정

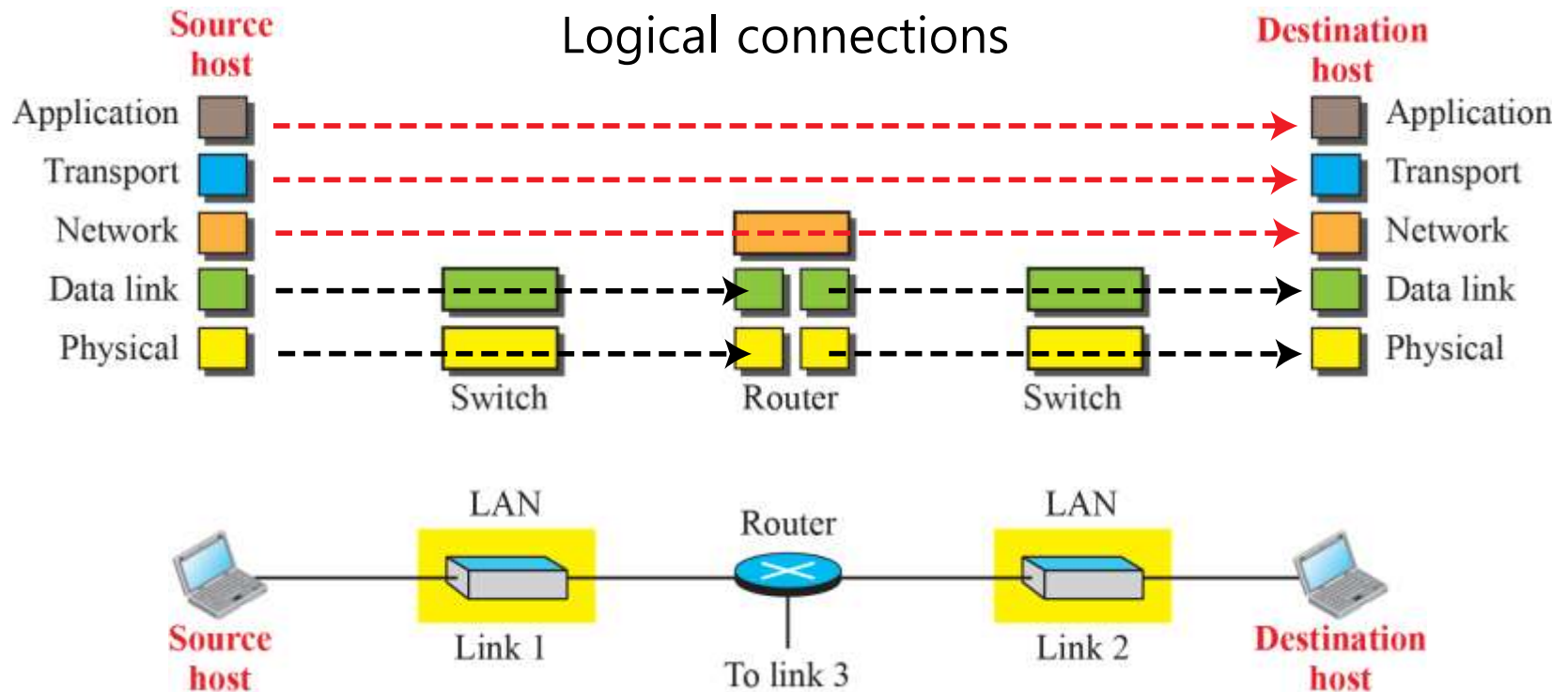
✓ 각 링크는 하나의 라우터에 연결됨



# 각 계층 간 논리적 연결

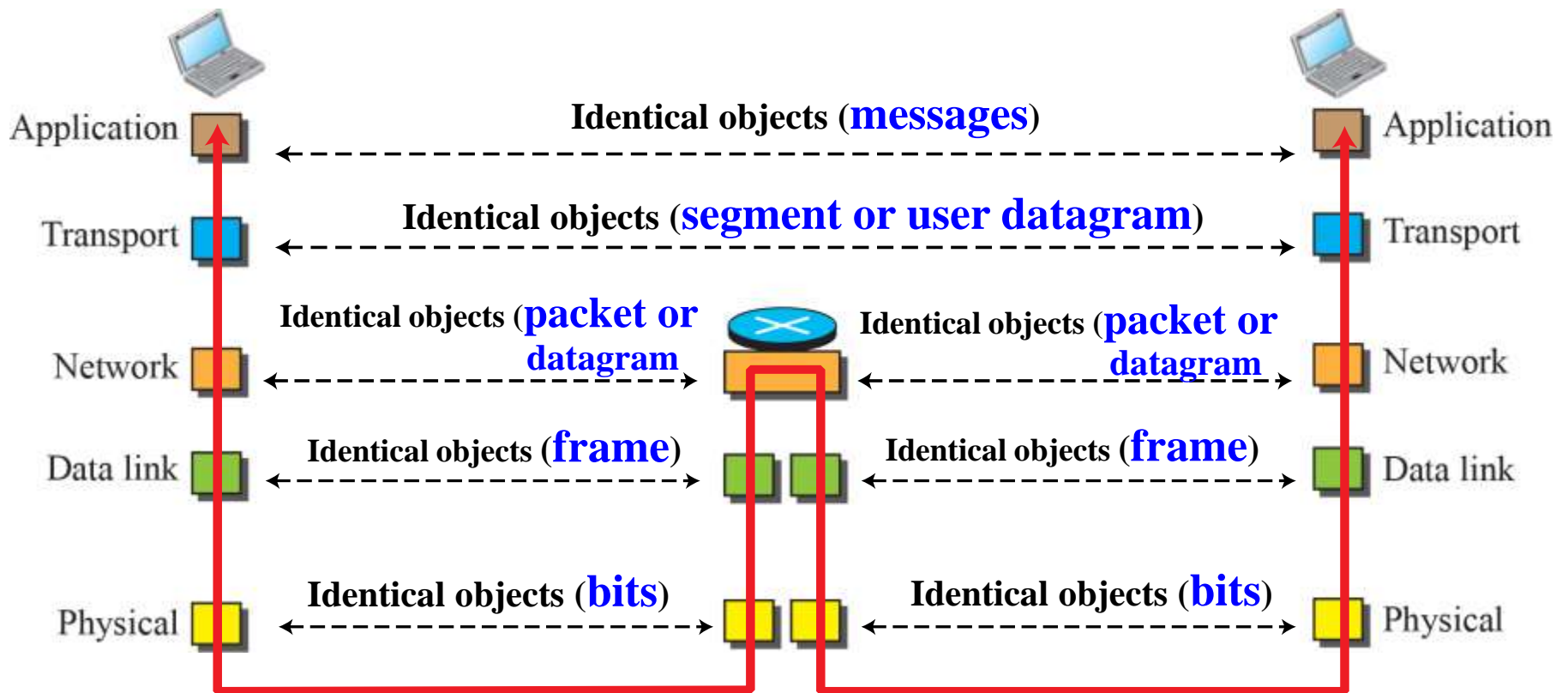
## ■ 각 계층의 의무

- ✓ 종단 대 종단(end-to-end) vs. 홉 대 홉(hop-to-hop)



# 각 계층 내의 동일한 객체들

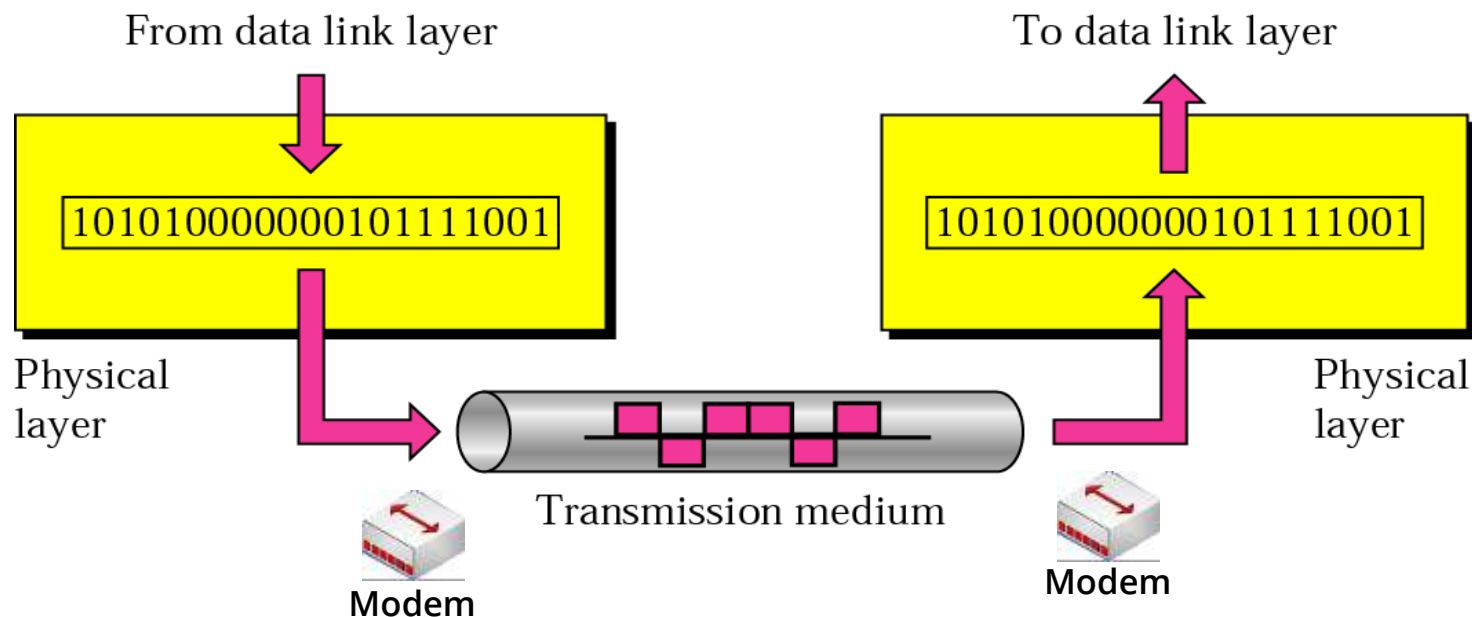
- 각 장치에 있는 각 계층은 동일한 객체를 가짐



# 계층1: 물리층(Physical Layer)

## ■ 장치간의 물리적인 접속을 제어하기 위한 기능을 제공하는 계층

- ✓ 물리적 매체를 통해 비트 스트림을 신호로 전달하는 책임
- ✓ 물리적인 장치와 인터페이스가 전송을 위해 필요한 기능과 처리 절차를 규정

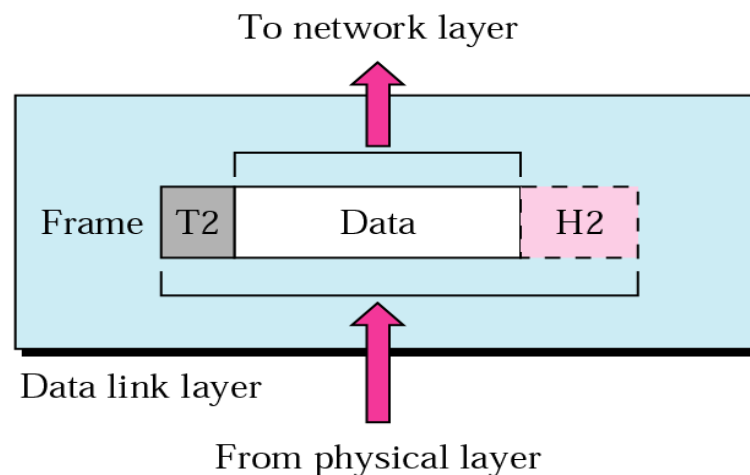
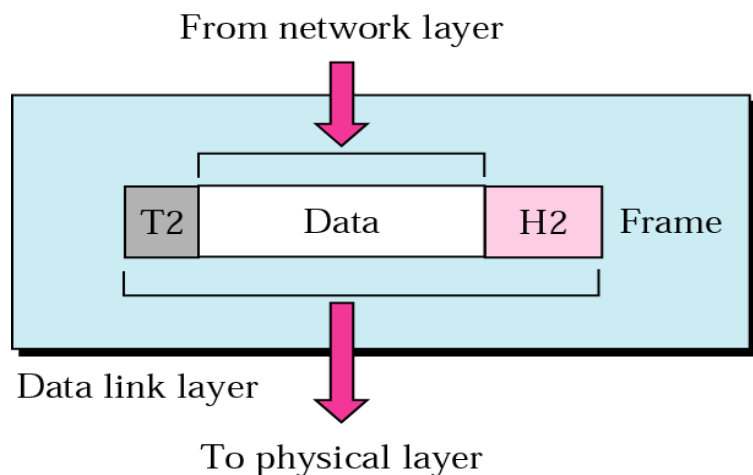


- 데이터 부호화 방식, 신호형식, 데이터 충돌 감지 등을 정의
- 기계적, 전기적, 기능적 및 절차적 특성을 고려
  - ✓ 연결 유형: 매체에 대한 장비의 연결 방법
  - ✓ 물리적인 접속형태
  - ✓ 전송속도(신호가 유지되는 bit의 주기) 규정
- 인터페이스와 매체의 물리적인 특성 규정
  - ✓ 전송매체의 유형에 대해 규정
  - ✓ 장치와 전송매체 간의 인터페이스 특성 규정
- 대표적인 프로토콜

V.24, RS-232C, RS-449, CDMA PHY, Coax, Fiber, Microwave, Satellite

## 계층2: 데이터 링크층(Data Link Layer)

- 물리계층에서 사용되는 통신매체를 통해 데이터 frame의 전송 에러 검출 및 에러 제어를 관리하고 규정하는 계층
- 한 node에서 다른 node로 프레임들을 전송하는 책임
  - ✓ 같은 링크에 있는 두 시스템 간의 패킷 전달을 제공
  - ✓ 다양한 링크층 프로토콜에 따라 다양한 서비스 제공
- 2개 부계층, Media Access Control (MAC)과 Logical Link Control (LLC)으로 나뉜다.



# 데이터 링크층의 역할

## ■ 프레임 구성

- ✓ 네트워크층으로부터 받은 비트 스트림을 프레임 단위로 캡슐화

## ■ 물리주소 (physical address) 지정

- ✓ 프레임의 송/수신자의 물리주소 정보를 헤더에 추가
- ✓ 수신자가 외부 네트워크에 연결된 경우, 수신자 주소는 해당 네트워크에 연결된 접속장치의 주소가 됨

## ■ 흐름제어(flow control)

- ✓ 송수신자 사이의 전송률을 제어

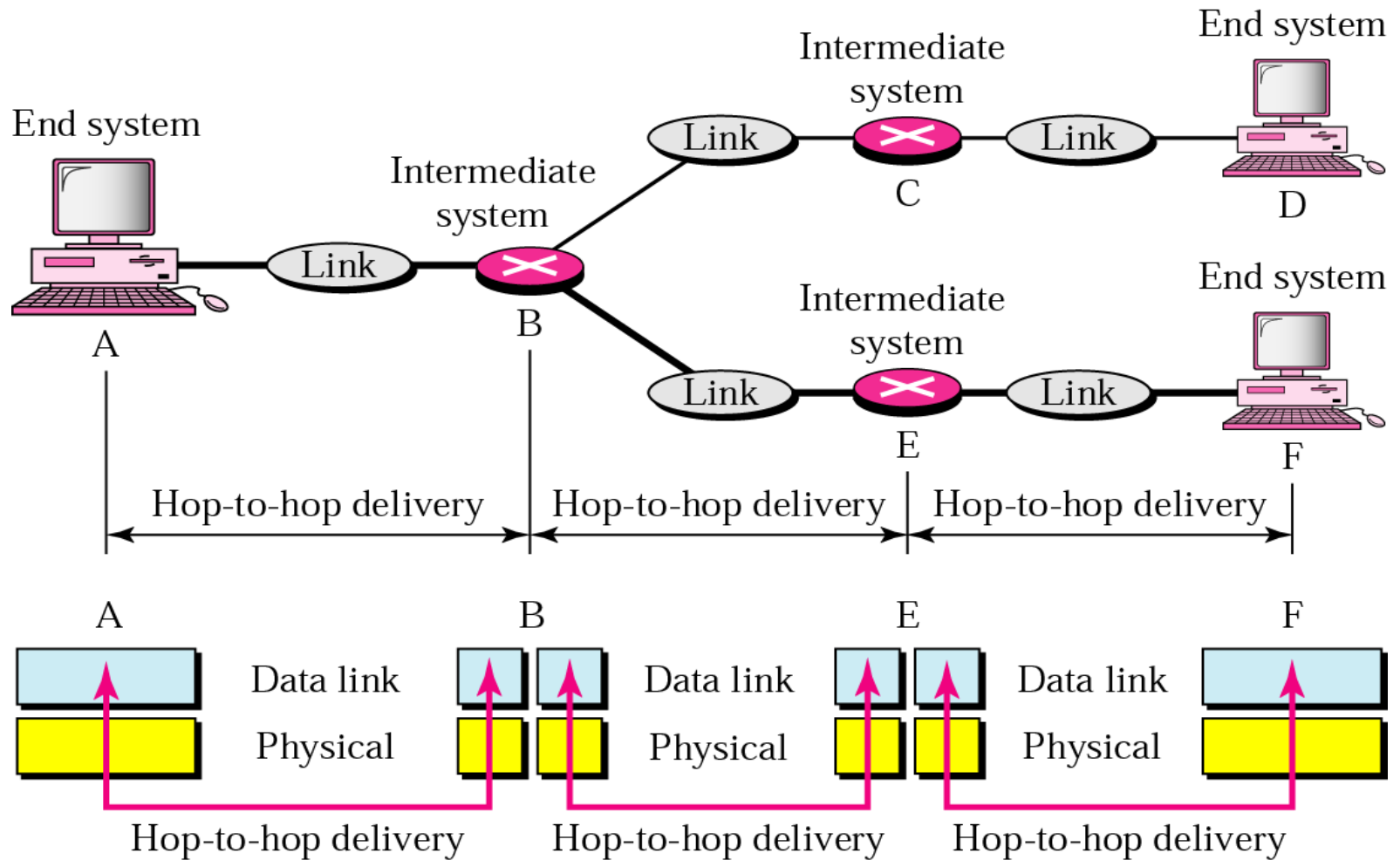
## ■ 오류제어(trailer 이용)

- ✓ 전송시 오류나 분실한 프레임을 찾아 재전송
- ✓ 프레임 중복을 막기 위한 메커니즘도 사용

## ■ 접근제어(access control)

- ✓ 같은 링크에 다수 장치가 연결되어 있을때, 주어진 순간에 링크를 사용하는 장치 결정

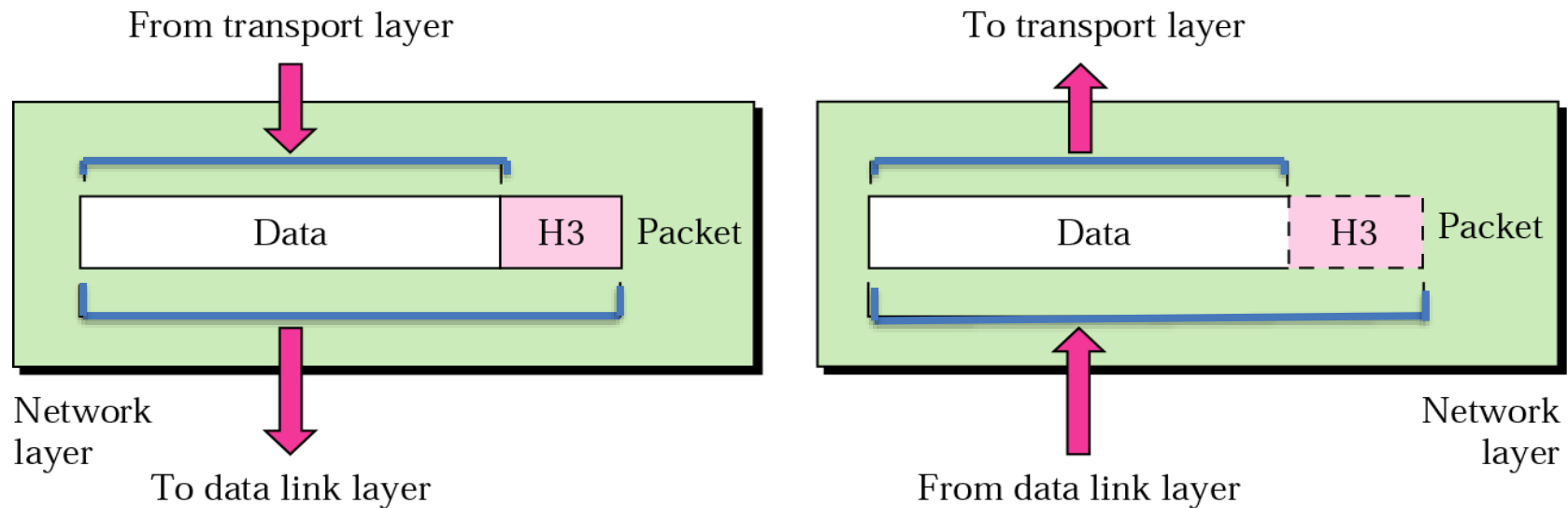
# Node-to-Node Delivery





# 계층3: 네트워크층(Network Layer)

- 사용자 데이터에 대한 경로선택, 데이터 교환 및 중계 기능
- 각 패킷을 발신지로부터 여러 네트워크를 통해 최종 목적지로 전달하는 책임

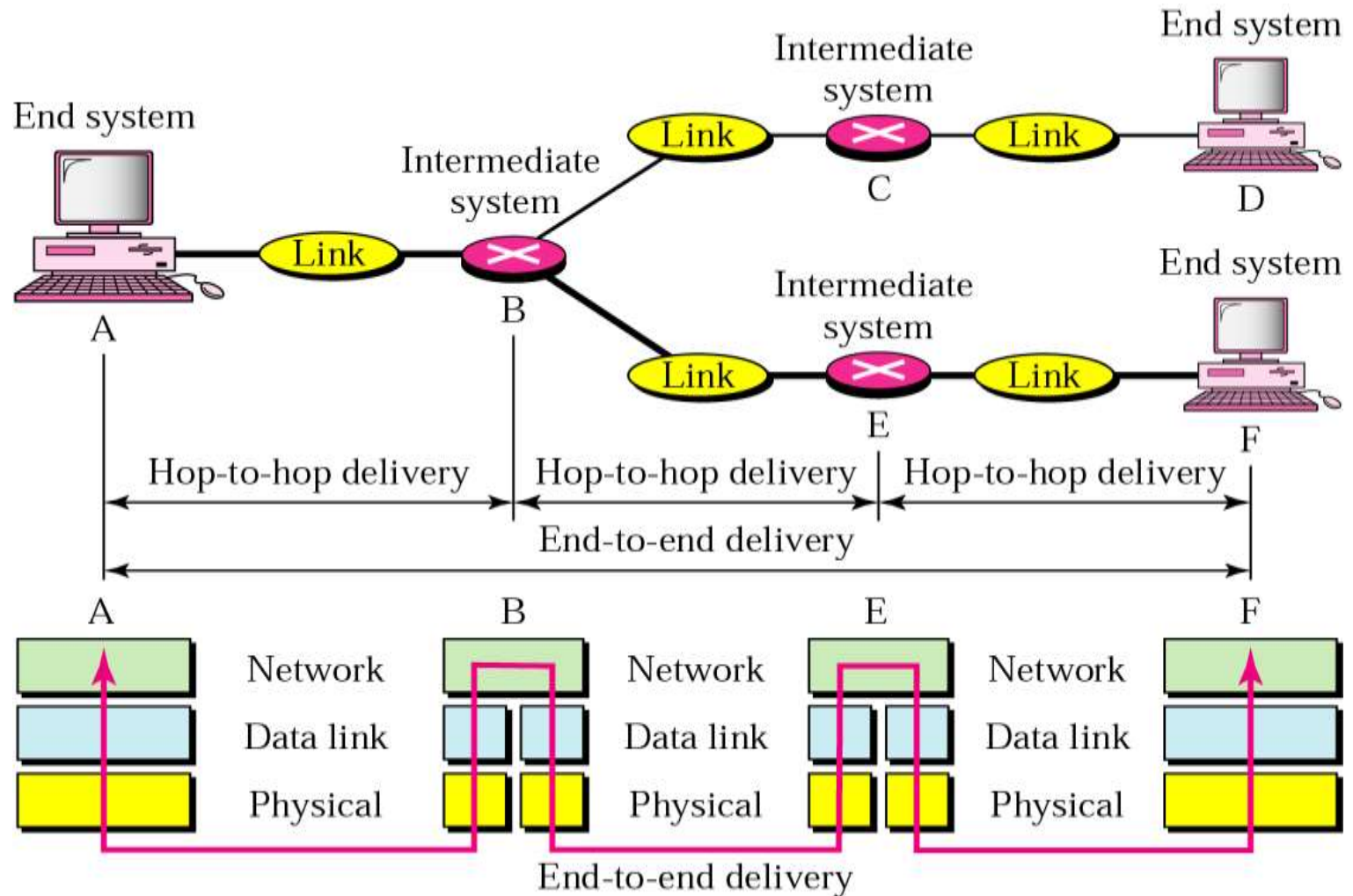


# 네트워크층의 기능

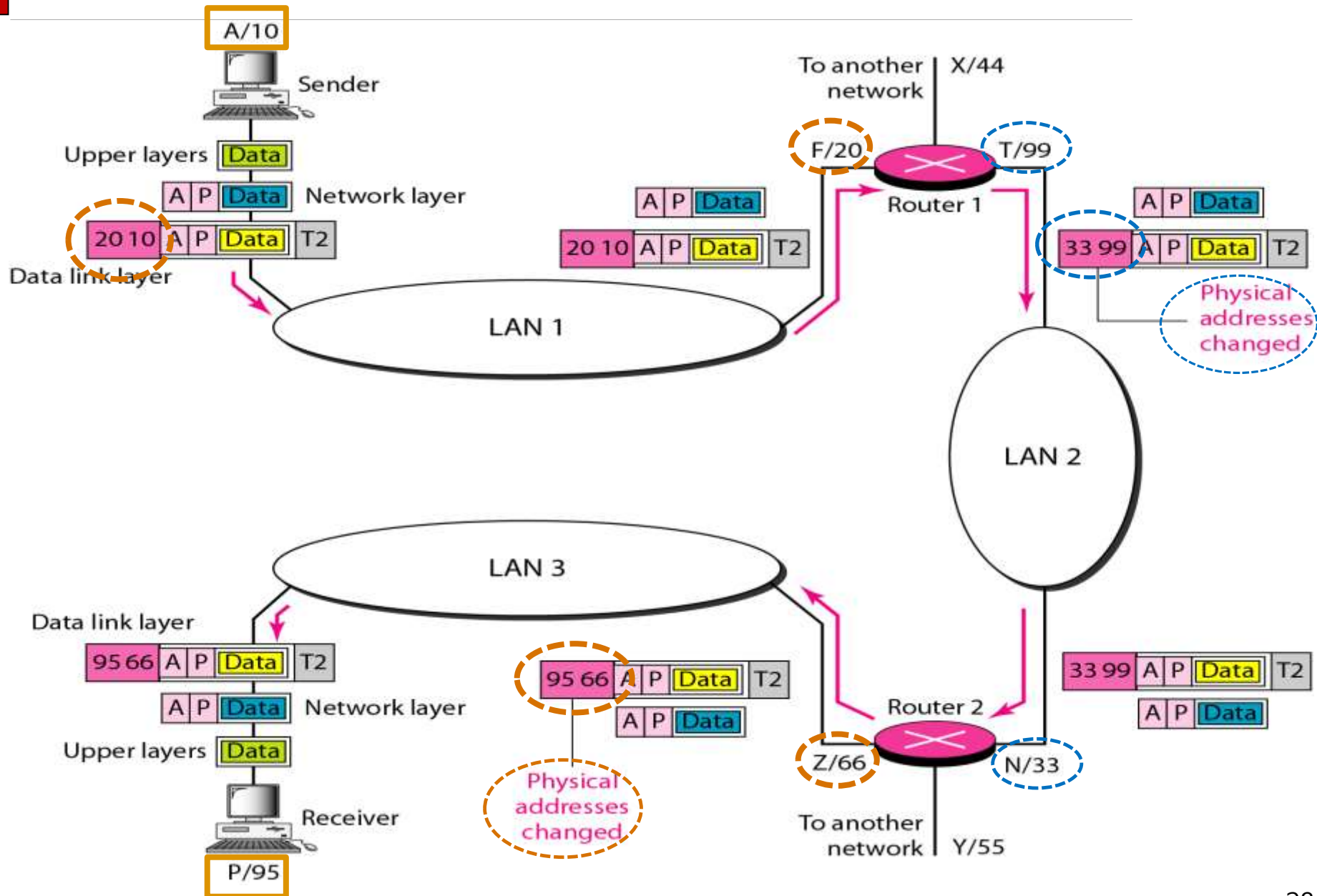
---

- 데이터그램이라는 패킷의 형식을 정의하는 인터넷 프로토콜 (IP, Internet Protocol)을 포함
- 라우팅 메커니즘 제공
  - ✓ 패킷이 최종 목적지에 전달되도록 경로 지정하거나 교환 기능
  - ✓ 라우터(Router): 독립적인 네트워크를 연결하여 inter-network 구성할 때 사용하는 장치
- 일-대-일(unicast), 다-대-다(multicast) 라우팅 프로토콜
- 논리주소(Logical address) 지정
  - ✓ 헤더에 발신지와 목적지의 논리주소 추가
- 대표 프로토콜 : IPv4, IPv6
- IP 전달과 라우팅 작업을 도와주는 보조 프로토콜 포함
  - ✓ ICMP, IGMP, DHCP, ARP 등

# Source-to-Destination Delivery

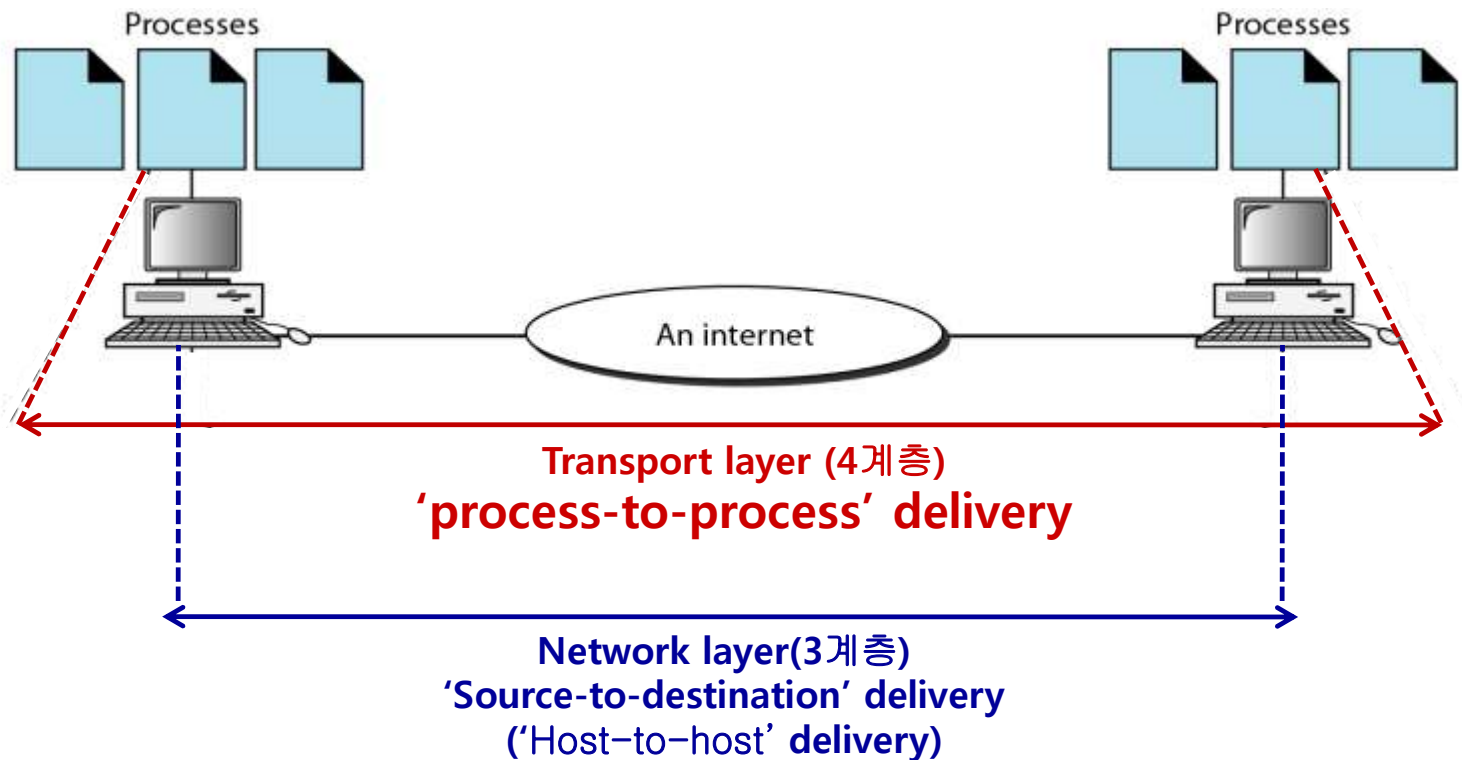


# 네트워크층-데이터 링크층의 예

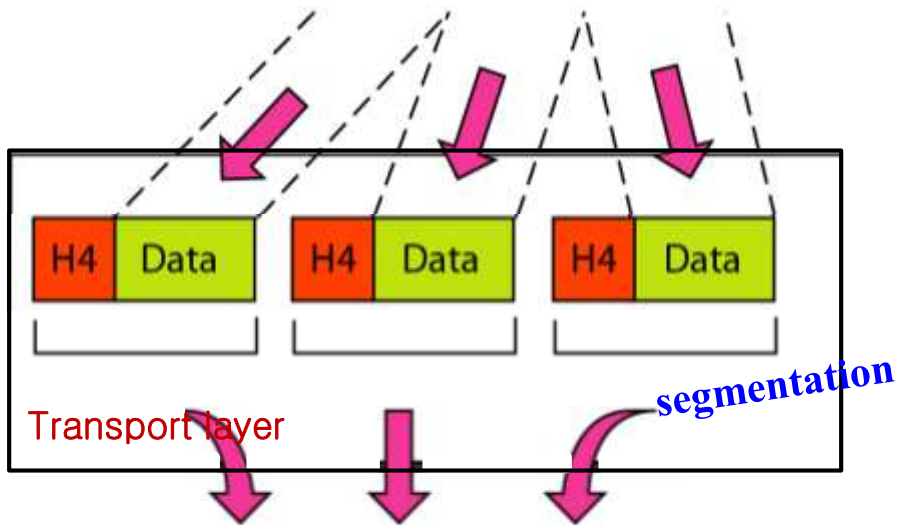


## 계층4: 전송층(Transport Layer)

- 종단-대-종단(process-to-process)에 대해 하나의 process로부터 다른 process로 메시지를 전달하는 책임
  - ✓ port addressing : 프로세스 구분을 위한 주소 개념 필요 (예:포트 번호)



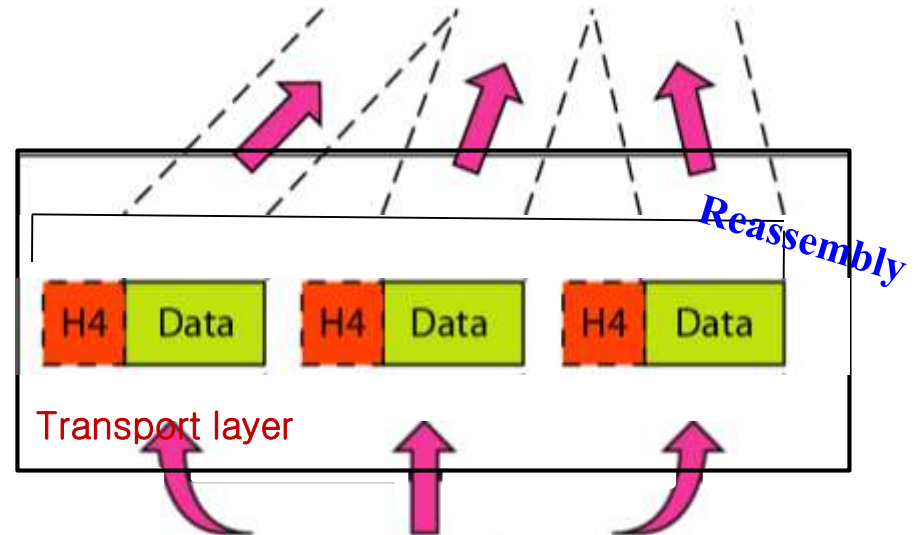
Session layer 로 부터



Network layer 로 ..

<송신호스트>

Session layer으로 ..



Network layer로 부터

<수신호스트>

# 전송층의 역할

- 서비스 지정 주소(포트 주소) 지정
  - ✓ 전체 메시지를 해당 컴퓨터의 정확한 프로세스에 전달
- 분할과 재조립
  - ✓ 메시지를 전송 가능한 세그먼트(데이터그램) 단위로 캡슐화
  - ✓ 각 세그먼트는 순서번호를 가짐
    - 목적지에서 재조립 혹은 손실 패킷 검출시 사용
- 연결제어
  - ✓ 비연결 / 연결지향
- 흐름제어
  - ✓ 발신지의 데이터 송신율과 목적지의 수신율을 맞춤
- 오류제어
  - ✓ 오류 없이 전달되고 훼손된 세그먼트의 재전송 보장
- 혼잡제어
  - ✓ 네트워크 혼잡으로 인한 세그먼트 손실을 줄임

# 전송층의 주요 프로토콜

---

- 여러 프로토콜이 존재하며 각 구체적 작업을 위해 설계됨
  - ✓ TCP(Transmission Control Protocol)
    - 연결형 프로토콜
    - 흐름제어/오류제어/혼잡 제어를 제공
  - ✓ UDP(User Datagram Protocol)
    - 비연결형 프로토콜
    - 흐름제어/오류제어/혼잡 제어를 제공하지 않음
  - ✓ SCTP(Stream Control Transport Protocol)
    - 멀티미디어를 위한 응용 위해 설계



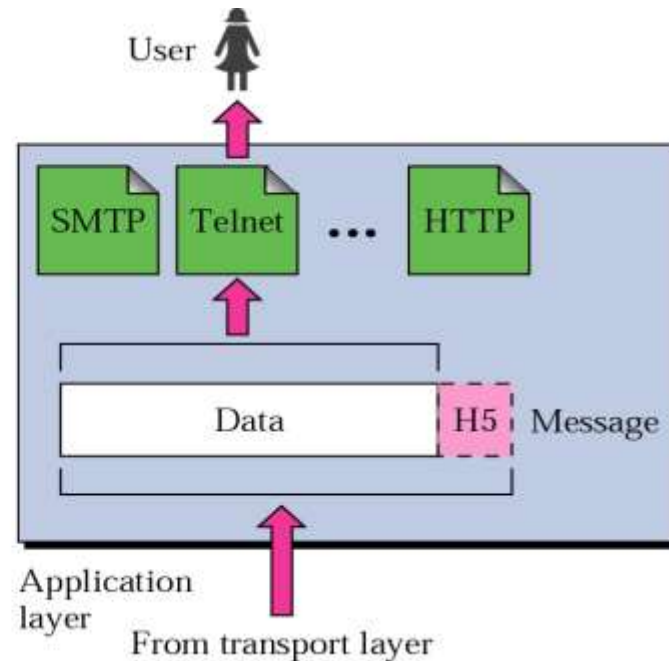
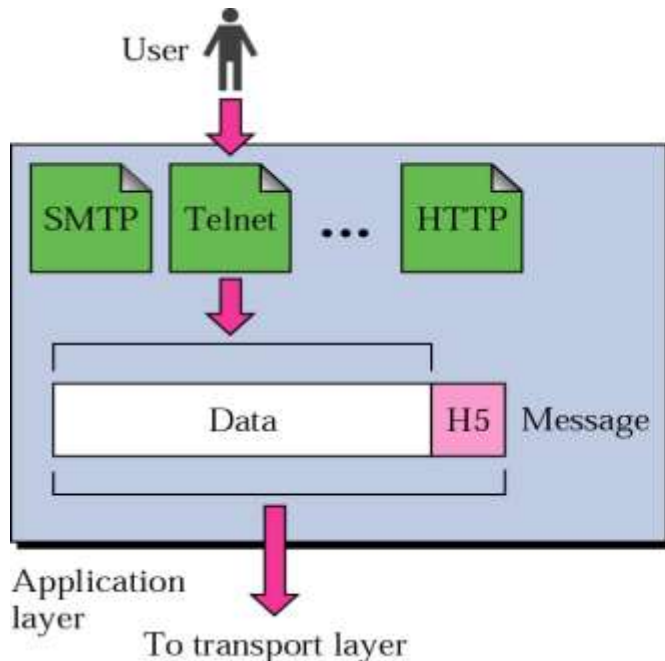
# 계층5: 응용층(Application Layer)

## ■ 사용자에게 서비스를 제공하는 책임

- ✓ 종단-대-종단에 대해 응용 프로세스들의 정보교환이 되는 계층

## ■ 대표 프로토콜

- ✓ SMTP(이메일 전송), FTP(파일전송), Telnet(원격접속), HTTP(인터넷 하이퍼터미널 액세스)



# Specific Address

---

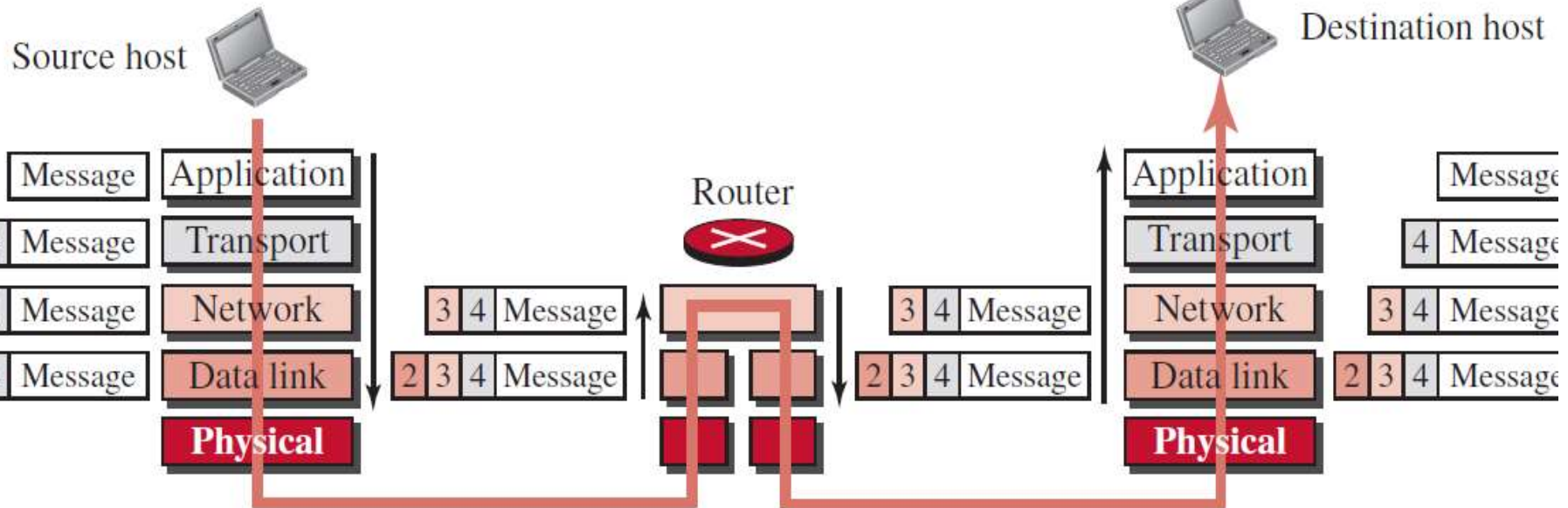
- 특정 응용 프로그램이 사용하는 주소
  - ✓ 사용자의 편리한 사용 위해
  - ✓ 예) 전자우편주소, URL 등
- 송신 컴퓨터에 의해 논리주소와 포트주소로 변환됨

# 캡슐화와 역캡슐화

## Legend

- 4 Header at transport layer
- 3 Header at network layer
- 2 Header at data-link layer

↓ Encapsulate  
↑ Decapsulate



\* 송신호스트: 상위계층에서  
하위계층으로 갈수록 헤더 추가  
➔ **Encapsulation**

\* 수신호스트: 하위계층에서  
상위계층으로 갈수록 헤더 제거  
➔ **Decapsulation**

## ■ 발신지 호스트에서 캡슐화

1. 응용층에서 교환되는 메시지는 전송층에 전달
2. 전송층에 반드시 전달해야 하는 페이로드로 받아 헤더 정보를 추가하여 세그먼트 또는 데이터그램으로 만들어 네트워크층에 전달
3. 네트워크층은 페이로드로 받아 헤더를 추가하여 데이터그램이라는 네트워크층 패킷으로 만들어 링크층에 전달
4. 링크층은 페이로드로 받아 헤더추가해 프레임 만들어 물리층에 전달

## ■ 라우터에서 캡슐화와 역캡슐화

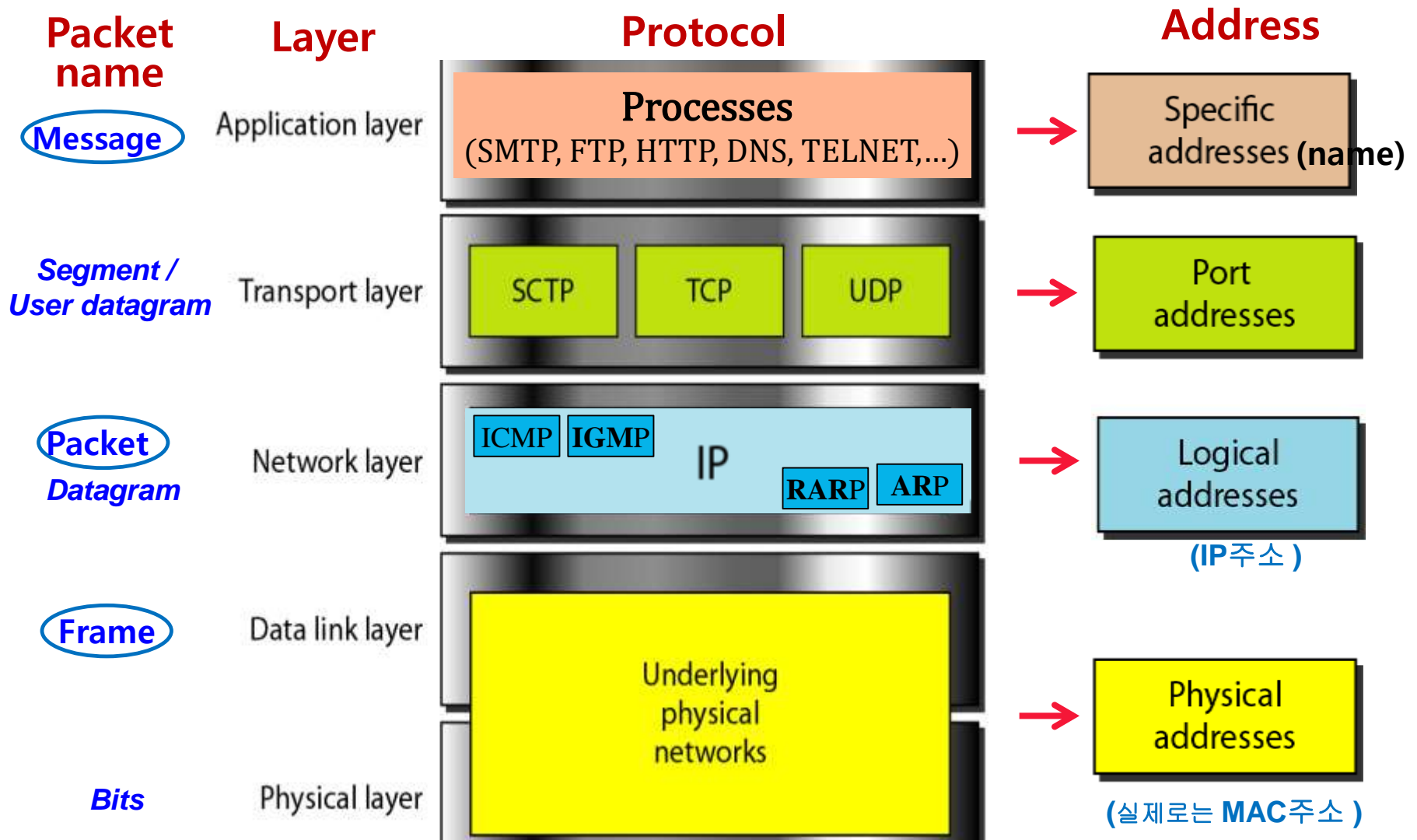
1. 1계층에 비트들이 전송되면 프레임으로부터 데이터그램을 역캡슐화
2. 3계층은 헤더를 조사하여 전송할 다음홉을 찾기 위해 포워딩 테이블 조사, 다음링크의 2계층으로 보냄
3. 다음링크의 2계층은 프레임을 캡슐화하고 1계층에 보냄

## ■ 목적지에서 캡슐화

1. 각계층은 응용층까지 역캡슐화하고 페이로드를 상위층에 전달
2. 오류 검사 필요

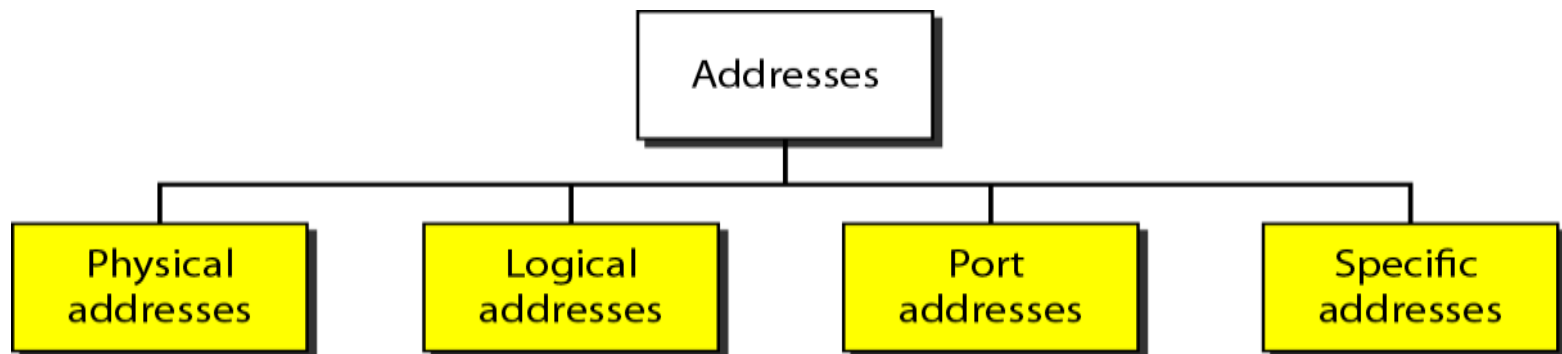
# TCP/IP 프로토콜 및 주소 지정

- 각 계층의 프로토콜, 메시지 이름, 발신지와 목적지간 통신을 위해 사용되는 4쌍의 주소



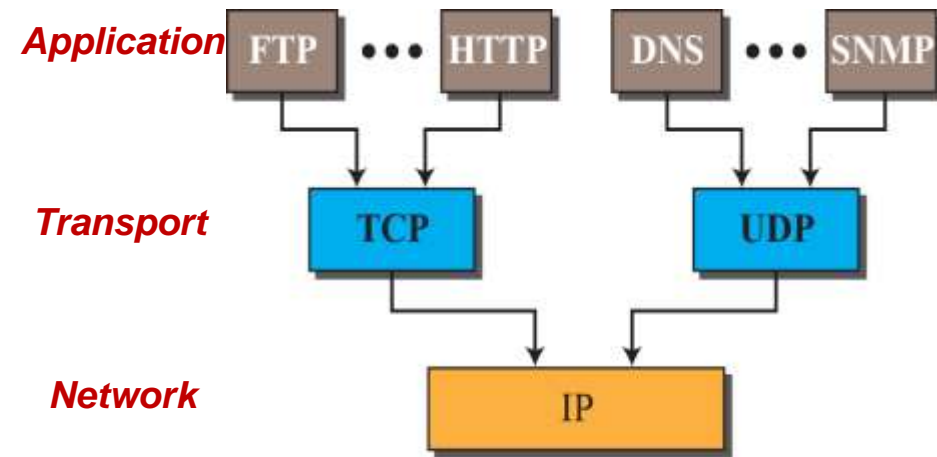
## ■ TCP/IP를 사용하는 인터넷 : 4종류의 주소 사용

- ✓ 1, 2계층: 물리주소- 실제로는 **MAC 주소**
  - 좁은 지역에서만 의미를 가짐. (널리 전달되지 않음)
- ✓ 3계층: 논리적 주소- IP 주소등 광역으로 사용하기 위한 주소
  - 물리주소와 논리주소 변환이 필요 : ARP(Address Resolution Protocol)
- ✓ 4계층: Port 주소- 프로세서 구분 (TCP, UDP, TELNET 등)
- ✓ 5계층: 특정프로세스가 사용하는 별칭주소(name)
  - 사용자 친화형 주소(전자우편주소, URL 등)

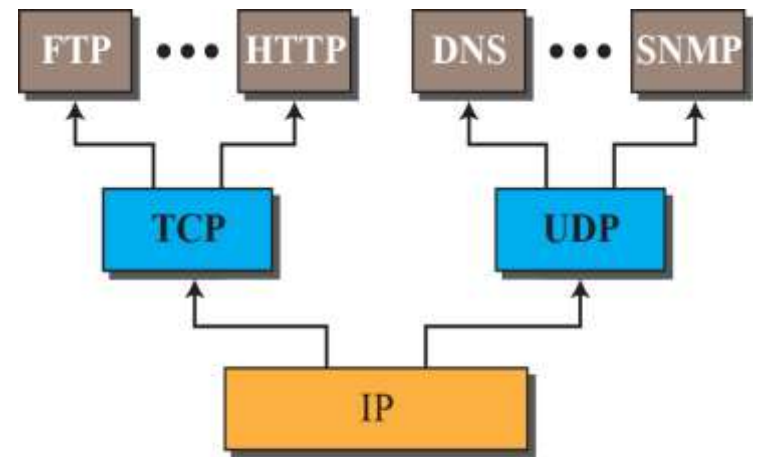


# 다중화(multiplexing)와 역다중화(demultiplexing)

- 일부 계층에서 여러 개의 프로토콜을 사용하기 때문에 발신지에서의 다중화, 목적지에서의 역다중화 필요



a. Multiplexing at source



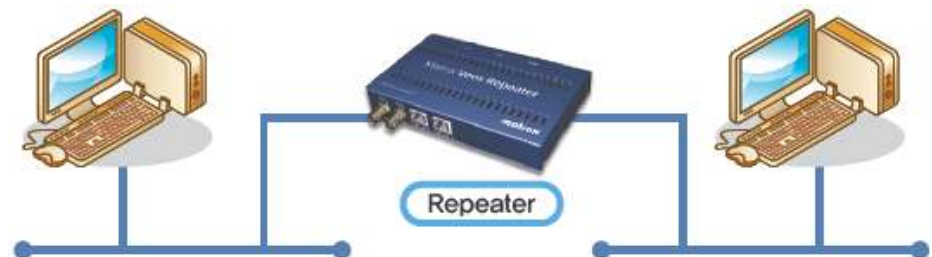
b. Demultiplexing at destination

# [참고] 네트워크 장치들 :

리피터, 허브, 브리지, 스위치, 라우터, 게이트웨이

## ✓ 리피터(repeater)

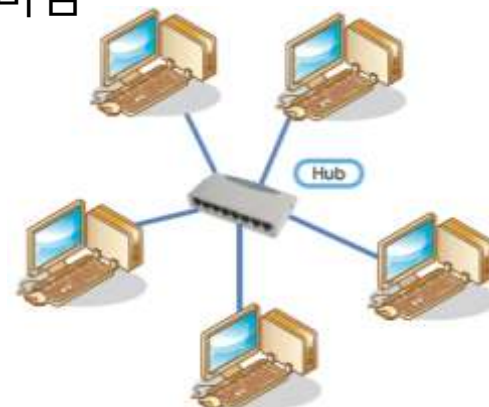
- 물리 계층에서만 동작하는 장치
- 주된 기능: 신호의 증폭
- 증폭기 vs. 리피터
  - 증폭기
    - 신호와 잡음을 구별할 수 없기 때문에 입력되는 모든 것을 증폭
  - 리피터
    - 신호 재생기(신호를 증폭시키는 것이 아니라 새로 만드는 것)





## ✓ 허브(hub)

- 물리 계층에서 동작
- 허브는 하나의 네트워크 안에 여러 장치들을 연결함
- 종류
  - 더미 허브(dummy hub)
    - 한 선의 입력 신호를 다른 선들에게 그대로 전달
  - 스위칭 허브(switching hub)
    - 더미 허브의 기능에 목적지 주소에 해당하는 선으로 입력 신호를 전송할 수 있는 기능을 추가
    - 데이터 전송 효율이 높으나 가격 또한 비쌘



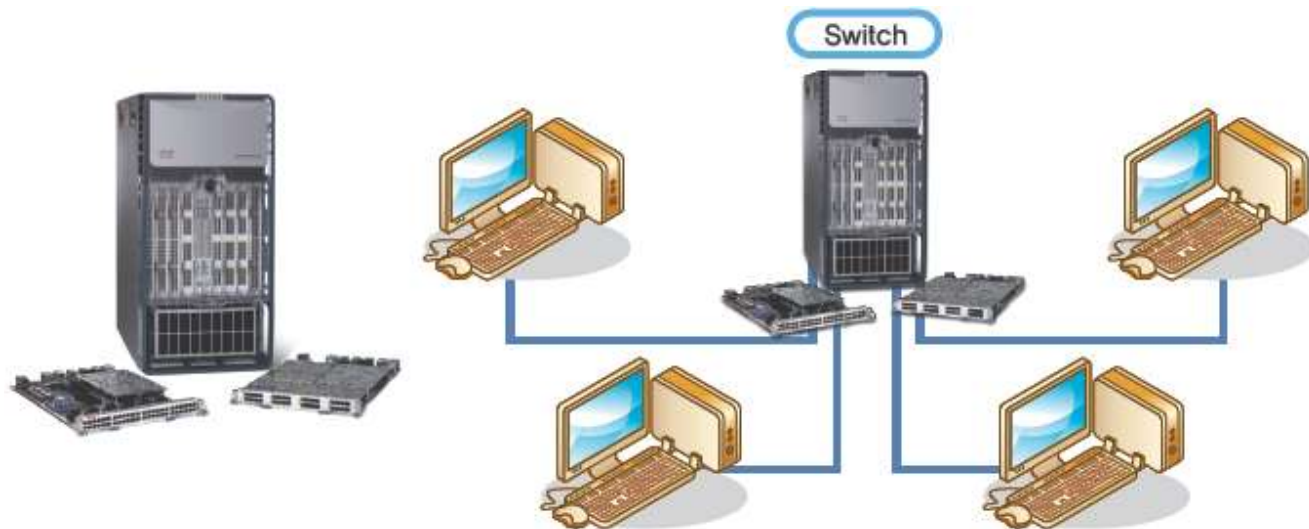
## ✓ 브리지(bridge)

- 물리 계층 및 데이터 링크 계층의 연결 기능을 제공
- 데이터를 전송할 때 주소를 확인하고 판단하여 접속
- 네트워크에 연결된 컴퓨터들의 통신 프로토콜을 바꾸지 않고 네트워크를 확장할 수 있도록 함
- 데이터 전송을 위하여 오류 및 흐름 제어를 가짐
- 장점
  - 설치가 쉽고 관리하기가 편함
  - 값이 저렴함
- 단점
  - 선택을 다양하게 할 수 없음
  - 지연을 유발할 수 있음
  - 장애 발생 시 대처가 어려움



## ✓ 스위치(switch)

- 데이터 링크 계층에서 동작하는 통신 기기
- 네트워크를 작게 분할하여 정체가 되는 구역을 줄여주는 장치



## [참고] 데이터링크 계층에서 사용하는 장비

### ① 랜카드

- LAN어댑터나 NIC(network interface card) 등으로 불림
- 랜카드 자체는 물리계층에서 작동하고, 드라이버를 포함하면 데이터 링크에서 작동



### ② 브리지(bridge)

- 두개 혹은 그 이상의 네트워크 세그먼트를 연결하거나 패킷을 전송할때 사용
- 소프트웨어 기반으로 작동



### ③ 스위치(switch)

- 멀티포트 브리지라고 부름
- 하드웨어 기반으로 작동(브리지와 기능은 같음)
- ASIC(주문형 반도체)를 기반으로 만들어 졌으며 VLAN(virtual LAN)기능이 있음



## ✓ 라우터(router)

- 동일한 프로토콜을 사용하는 네트워크를 연결
- 네트워크 계층에서 사용
- 장점
  - 라우터는 전체 네트워크 성능을 개선
  - 유지보수가 쉬움
  - 대규모 통신망을 쉽게 구성
- 단점
  - 초기에 환경설정이 어려움
  - 일부 프로토콜만을 지원
  - 가격이 비쌈



## ✓ 게이트웨이(gateway)

- 프로토콜이 전혀 다른 네트워크 간을 연결
- 세션 계층, 표현 계층, 응용 계층 간을 연결
- 메시지 형식 변환, 주소 형식 변환, 프로토콜 변환 등의 기능 수행
- 장점 : 네트워크 사이의 유연성 제공
- 단점 : 복잡하고 비쌈



# 2장 - 끝 -