

BIT BY BIT

2025/02/14 (금)

Session 8 – 네트워크 Part4

박설진

타임라인

10분 - 오프닝

1시간 30분 - 공유 세션 및 토의 진행

20분 - 마무리

오프닝

강의 듣는데 다들 어떠셨나요?

Ex.

다 들었는지,,

이해는 어느 정도 됐는지,,

시간이 부족하거나 남지는 않았는지 등
숙제여부 / 공부 내용 난이도 / 컨디션

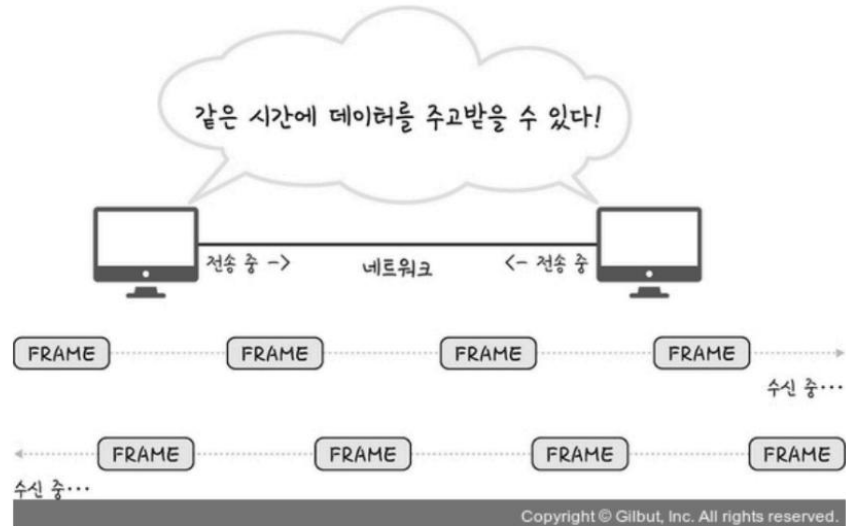
공유 세션

✔ 유선LAN #1. 전이중화 통신, CSMA/CD ★★☆☆	02:36
✔ 유선LAN #2. 케이블 ★☆☆	01:49
✔ 무선LAN #1. 반이중화 통신, CSMA/CA, 와이파이 ★★☆☆	05:23
✔ 무선LAN #2. 주파수와 2.4GHz와 5GHz의 차이 ★☆☆	01:55
✔ DEEP DIVE : 대규모 트래픽으로 인한 서버 과부하 해결방법 #1 ★★★★★	09:16
✔ DEEP DIVE : 대규모 트래픽으로 인한 서버 과부하 해결방법 #2 ★★★★★	08:38
✔ DEEP DIVE : 대규모 트래픽으로 인한 서버 과부하 해결방법 #3 ★★★★★	07:53
✔ Q. REST API란 무엇인가요? ★★★★★	18:08
✔ Q. 브라우저 렌더링과정(Browser Rendering)이란 무엇인가요? ★★☆☆	10:55
✔ Q. www.naver.com을 찾을 때 생기는 과정, 그리고 DNS까지 설명해주세요. ★★★★★	10:30
✔ Q. 이더넷 프레임은 무엇이며 구조가 어떻게 되나요? ★☆☆	02:20
✔ Q. CORS란 무엇인가요? ★★★★★	11:17
✔ Q. 네이글 알고리즘이란 무엇인가요? ★☆☆	02:41
✔ Q. HTTP의 멍등성이 무엇인지 설명해주세요. ★★★★★	05:02
✔ HTTP 멍등성을 고려한 API 구축 실습 ★★★★★	04:43

1. 유선 LAN / 무선 LAN

1. Full duplex | half duplex
2. CSMA/CD | CSMA/CA
3. 무선 LAN – 2.4GHz, 5GHz차이

1. 유선 LAN



▲ 그림 2-24 전이중화 통신

IEEE802.3

IEEE SA STANDARDS ASSOCIATION



P802.3dk

Type of Project: Amendment to IEEE Standard 802.3-2022
Project Request Type: Initiation / Amendment
PAR Request Date: 26 Sep 2022
PAR Approval Date: 03 Dec 2022
PAR Expiration Date: 31 Dec 2026
PAR Status: Active
Root Project: 802.3-2022

1.1 Project Number: P802.3dk
1.2 Type of Document: Standard
1.3 Life Cycle: Full Use

2.1 Project Title: Standard for Ethernet
Amendment: Greater than 50 Gb/s Bidirectional Optical Access PHYs

IEEE802.3이란 이더넷프레임은 어떤 구조를 기반으로 할 것인지, 케이블의 최대 전송량, 어떤 케이블만이 가능하도록 할 것인지 등을 정한 규칙을 말합니다.

1. 동축케이블, 광케이블 등을 기반으로 만들어진 유선 LAN을 이루는 이더넷은 **IEEE802.3 프로토콜** 기반으로 **전이중화 통신**을 사용한다.

1. 유선 LAN – UTP/STP

UTP와 STP의 비교

항목	UTP (Unshielded Twisted Pair)	STP (Shielded Twisted Pair)
차폐	없음	있음 (금속 차폐막)
가격	저렴	비쌈
내구성	전자기 간섭에 민감	전자기 간섭을 차단하여 더 견고
용도	일반적인 데이터 전송, 로컬 네트워크	고속 데이터 전송 및 전자기 간섭이 있는 환경
설치 용이성	설치가 간편	설치가 다소 복잡할 수 있음

트위스트 페어 케이블(Twisted Pair Cable)은 두 개의 구리선이 서로 꼬여있는 구조로, 전자기 간섭(EMI)이나 잡음을 줄이기 위해 사용됩니다.

1. 무선 LAN

CSMA/CD와 CSMA/CA 비교

특성	CSMA/CD (유선 네트워크)	CSMA/CA (무선 네트워크)
충돌 감지 방식	충돌 발생 시 감지하고 재전송	충돌을 미리 방지하려고 전송 예약 사용
사용 환경	유선 네트워크 (이더넷)	무선 네트워크 (Wi-Fi)
효율성	충돌이 적은 환경에서 효율적	충돌을 피하기 위해 지연이 발생할 수 있음
주요 장점	충돌 발생 시 빠르게 대응	무선 환경에서 안정적인 전송 보장

CSMA/CD와 CSMA/CA는 네트워크에서 **충돌 방지**를 위한 접근 방식으로, 주로 이더넷(Ethernet)과 같은 공유된 네트워크에서 사용됩니다. 이 두 가지는 데이터 전송을 위한 규칙을 정의하는 프로토콜로, 각각 **충돌을 감지하고 충돌을 회피**하는 방식에 따라 다릅니다.

1. 무선 LAN - CSMA/CD

작동 원리:

1. **채널 감지 (Carrier Sense):** 장치가 데이터를 전송하기 전에, 네트워크 상에 다른 장치가 데이터를 전송 중인지 확인합니다.
2. **데이터 전송:** 채널이 비어 있으면 데이터를 전송합니다.
3. **충돌 감지 (Collision Detection):** 전송 중에 다른 장치가 데이터를 전송하면 충돌이 발생할 수 있습니다. 만약 충돌이 발생하면, 장치는 이를 감지하고 전송을 중지합니다.
4. **재전송:** 충돌을 감지한 장치는 **랜덤한 시간** 후에 다시 전송을 시도합니다.

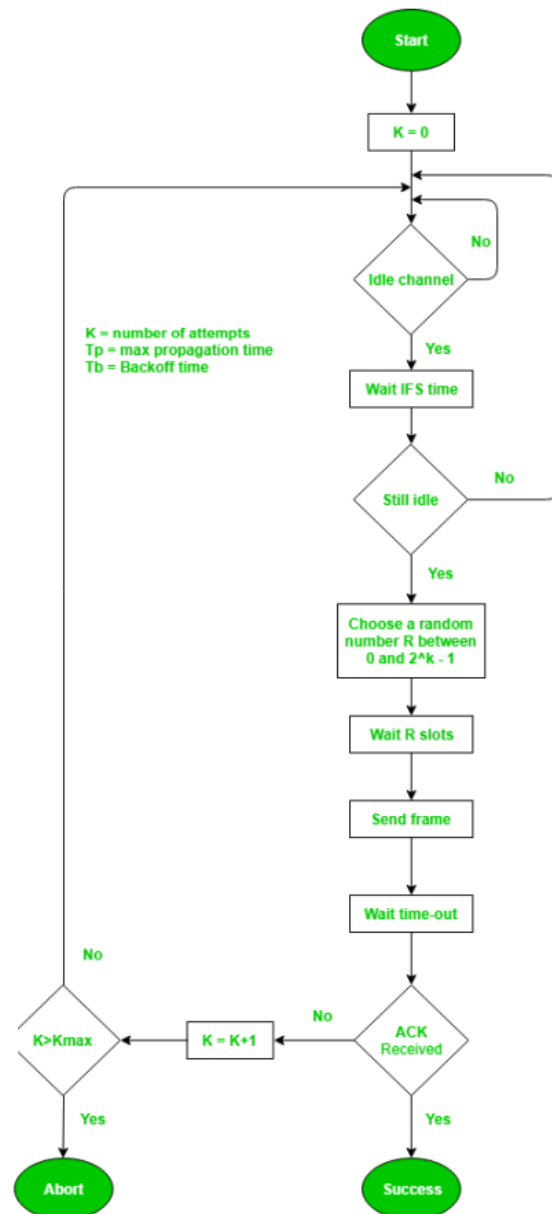
1. 무선 LAN - CSMA/CA

CSMA/CA

CSMA/CA는 반이중화 통신 중 하나로 장치에서 데이터를 보내기 전에 일련의 과정을 기반으로 사전에 가능한 한 충돌을 방지하는 방식을 말합니다.

CSMA/CA로 프레임을 보낼 때 다음과 같은 과정이 일어납니다.

1. 사용중인 채널이 있다면 다른 채널을 감지하다 유후 상태인 채널을 발견합니다.
2. 프레임 간 공간 시간인 IFS(InterFrame Space)시간만큼 기다립니다. IFS는 프레임의 우선순위를 정의할 때도 사용됩니다. IFS가 낮으면 우선순위가 높습니다.
3. 프레임을 보내기전 $0 \sim 2^k - 1$ 사이에서 결정된 랜덤상수를 기반으로 결정된 시간만큼 기다린 뒤 프레임을 보냅니다. 프레임을 보낸 뒤 제대로 송신이 되었고 ACK 세그먼트를 받았다면 마칩니다. 그러나 받지 못했다면 $k = k + 1$ 을 하며 이 과정을 반복합니다. 반복하다 k 가 정해진 K_{max} 보다 더 커진다면 해당 프레임전송은 버립니다. (abort)

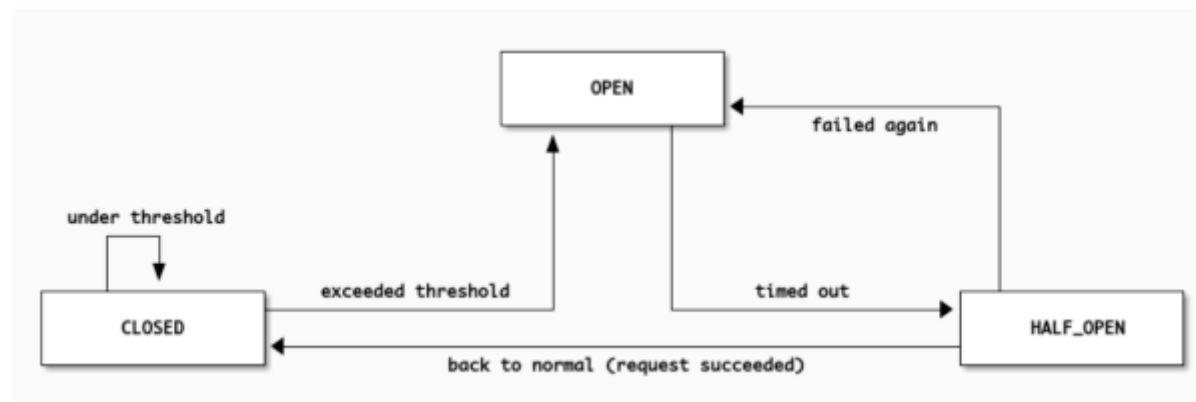
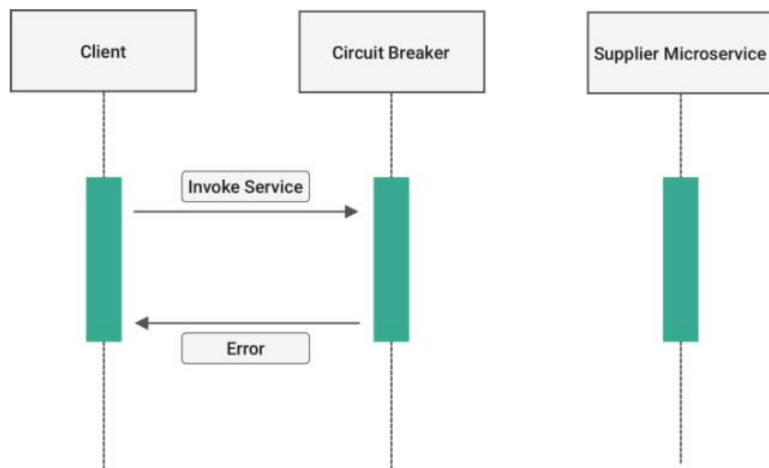
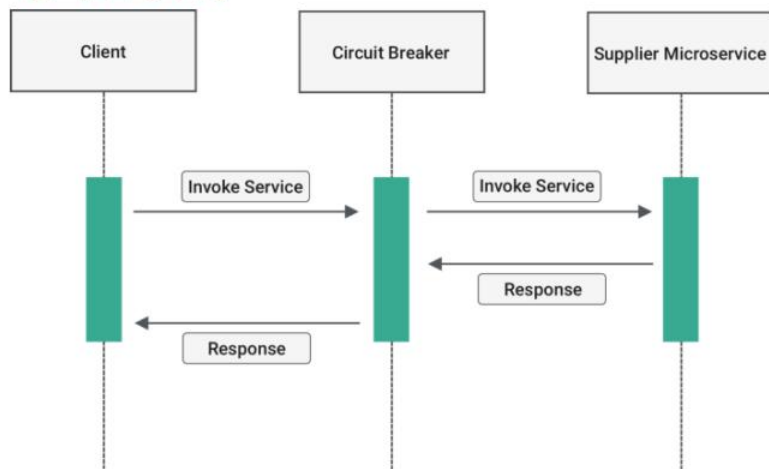


2. 대규모 트래픽으로 인한 서버 과부하 해결 방법

1. 서버과부하?
2. 모니터링 – AWS 오토 스케일링/netdata
3. 로드밸런서
4. 블랙스완 프로토콜
5. 서킷 브레이커 – 넷플릭스 Hystrix
6. 콘텐츠 관리 – 우하한 저하

2. 서버 과부하 해결 방법 - 서킷 브레이커

서킷브레이커 동작과정



서킷브레이커의 장점

연속적인 에러 발생을 막아주며 일부서비스가 종료되더라도 다른 서비스들은 이상없이 동작하게 만들 수 있으며 사용자경험을 높여 줍니다.

3. REST API란 무엇인가요?

- 1. REST API란?**
- 2. Uniform-Interface**
- 3. Stateless**
- 4. Cacheable**
- 5. Client-Server 구조**
- 6. Layered System**

3. REST API : Uniform-Interface

identification of resources

: 특정 자원을 찾기 위한 주소로 자원의 위치를 명확하게 지정한다

ex) `https://test.com/product/1201`

manipulation of resources through representations


: 자원의 표현을 통해 자원을 조작함

ex) `GET /users/123` , `DELETE /users/123`

3. REST API : Uniform-Interface self-descriptive message

- 요청 예시: 클라이언트가 서버에 JSON 데이터를 요청할 때:

http

 코드 복사

```
GET /users/123 HTTP/1.1  
Host: api.example.com  
Accept: application/json
```

클라이언트는 `Accept: application/json` 을 통해 서버에게 JSON 형식의 응답을 기대한다고 명시한다.


3. REST API : Uniform-Interface self-descriptive message

- **응답 예시:** 서버는 요청에 대한 응답을 JSON 형식으로 보내며, `Content-Type: application/json` 을 명시한다.

```
http

HTTP/1.1 200 OK
Content-Type: application/json

{
  "id": 123,
  "name": "John Doe",
  "email": "john@example.com"
}
```

 코드 복사

3. REST API : Uniform-Interface

HATEOAS

(Hypermedia As The Engine Of Application State)

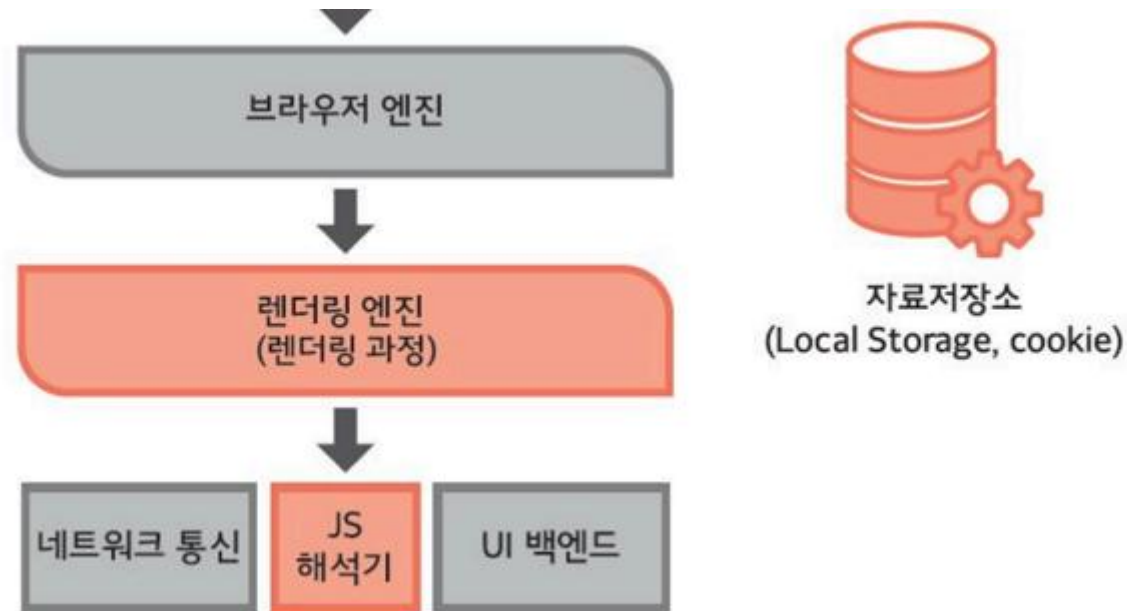
- 클라이언트가 `GET /users/123` 요청을 보내면 서버는 응답으로 자원과 함께 "다음에 할 수 있는 작업"에 대한 링크를 제공할 수 있다.

```
json 코드 복사

{
  "id": 123,
  "name": "John Doe",
  "email": "john@example.com",
  "links": {
    "self": "https://api.example.com/users/123",
    "update": "https://api.example.com/users/123/update",
    "delete": "https://api.example.com/users/123/delete"
  }
}
```

- 여기서 클라이언트는 `update` 링크를 통해 사용자의 정보를 수정할 수 있으며, `delete` 링크를 통해 사용자를 삭제할 수도 있다.

4. 브라우저 렌더링과정



4. 브라우저 렌더링과정

1. DOM트리와 CSSOM트리 구축
2. 렌더 트리와 렌더 레이어 생성
3. 렌더 레이어를 대상으로 Layout 설정
4. 렌더 레이어를 대상으로 칠하기
5. 레이어 합치기 및 표기

4. 브라우저 렌더링과정

1. 렌더트리와 DOM 트리는 1:1대응일까요?
2. Display:none과 visibility:hidden의 차이
3. 그래픽 레이어를 사용하는 이유?

5. www.naver.com

리다이렉트, 캐싱, DNS, IP 라우팅 & ARP, TCP 연결
구축을 거쳐 요청, 응답이 일어나는 TTFB(Time to
First Byte)가 시작되고 이 후 콘텐츠를 다운받게 되고
이 후 브라우저렌더링 과정을 거쳐 네이버라는
화면이 나타나게 됩니다.

6. CORS 란 무엇인가요?

1. 오리진이란?
2. SOP란?
3. CORS 의미
4. Preflight request와 simple request

6. CORS란? – simple request

이름	상태	유형	시작점	크기	시간	목표
127.0.0.1	200	document	기타	792 B	2.04초	
api	CORS 오류	fetch (선택)		0 B	7밀리초	
api	200	preflight	프러플라이트	0 B	1밀리초	
contentscript.js	200	script	inject.js	4.9 KB	24밀리초	

메서드 타입

- GET
- HEAD
- POST

헤더

- Accept
- Accept-Language
- Content-Language
- **Content-Type**
 - - application/x-www-form-urlencoded
 - - multipart/form-data
 - - text/plain
- Range (참고로 firefox 브라우저는 해당 헤더를 허용하지 않습니다.)

6. CORS란? – preflight request

[참고] Access-Control-* 헤더

- Access-Control-Allow-Headers: Content-Type
 - Access-Control-Allow-Origin: <https://kundol.com>
 - Access-Control-Allow-Methods: GET, DELETE, HEAD, OPTIONS
- 이외에도 Access-Control-Max-Age 등이 있습니다.

- 오리진 : Access-Control-Allow-Origin
- 메서드 : Access-Control-Allow-Methods(허용된 방법)
- 헤더 : Access-Control-Allow-Headers(허용된 헤더)
- 인증관련 헤더(true or false) : Access-Control-Allow-Credentials

7. 네이글 알고리즘이란?

1. 네이글 알고리즘이란?
2. 기존 방법과 네이글 알고리즘의 차이
3. 네이글 알고리즘의 장단점

8. HTTP 멱등성

1. HTTP 멱등성(idempotent) 이란?
2. 멱등성을 가지는 메서드
3. 멱등성을 가지지 않는 메서드
4. 멱등성이 중요한 이유

8. HTTP 멱등성

1. HTTP 멱등성(idempotent) 이란?
2. 멱등성을 가지는 메서드 – GET, HEAD, OPTIONS, PUT, DELETE
3. 멱등성을 가지지 않는 메서드 – POST, PATCH
4. 멱등성이 중요한 이유

감사합니다