



# R E P O R T

제목: 2장 연습문제

과목: 컴퓨터 네트워크

날짜(년/월/일): 2021/10/05

소속 학과: 컴퓨터 전자시스템공학과

학번: 201904458

이름: 이준용

본 보고서의 내용 중 다른 문서(자료)를 인용한 것이 있습니까?

예(✓) 아니오( )

위에서 '예'로 답한 경우, 인용한 다른 문서는 무엇인지 아래에  
명시해 주세요. (여러 개의 경우 주요 자료 2개 까지)

- 저자 1: [w3.org/Protocols/rfc/2616-sec10.html](https://www.w3.org/Protocols/rfc/2616-sec10.html)
- 제목 1: [part of Hypertext Transfer Protocol – HTTP/1.1](#)
- 저자 2: [RFC 2616 Fielding, et al.](#)
- 제목 2: [10. Status Code Definitions](#)

P1. True or false?

a. A user requests a Web page that consists of some text and three images. For this page,

the client will send one request message and receive four response messages. *False*

클라이언트는 유저로부터 한 번의 request message와 한 번의 response message를 받게 된다면 some text가 전달된 것입니다. 그리고 3개의 이미지가 남아있으므로 이미지 한번 전달하는데 위의 작업을 반복하는 것이므로 3번더 반복합니다.  
그러므로 4개의 request messages와 4개의 response messages를 받게 됩니다.

b. Two distinct Web pages (for example, `www.mit.edu/research.html` and `www.mit.edu/students.html`) can be sent over the same persistent connection. *True*

Persistent HTTP는 연결을 연린 상태로 둡니다.

c. With nonpersistent connections between browser and origin server, it is possible for a

single TCP segment to carry two distinct HTTP request messages. *False*

각 TCP segment는 하나의 요청만 전달할 수 있습니다.

d. The Date: header in the HTTP response message indicates when the object in the response was last modified. *False*

날짜는 request generation time(요청 생성 시간)입니다.

e. HTTP response messages never have an empty message body. *False*

204 No Content

서버는 요청을 수행했지만 엔티티 본문을 반환할 필요가 없으며 업데이트된 메타정보를 반환할 수 있습니다. 응답은 엔티티 헤더의 형태로 새롭거나 업데이트 된 메타정보를 포함할 수 있습니다. 존재하는 경우 요청된 변형과 연관되어야 합니다.

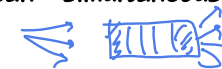
204 응답은 메시지 본문을 포함하지 않아야 하므로 항상 헤더 필드 다음의 첫 번째 빈 줄로 종료됩니다.

P20. Suppose you can access the caches in the local DNS servers of your department. Can you propose a way to roughly determine the Web servers (outside your department) that are most popular among the users in your department? Explain

로컬 DNS 서버에서 시스템 다운을 피하기 위해, 스냅샷을 기반으로 한 백업을 수행합니다. 그러므로 DNS캐시를 주기적으로 찍을 수 있습니다.  
DNS 캐시에 가장 자주 나타나는 서버는 웹 서버입니다. 아주 많은 사용자가 웹 서버에 관심이 있으면 해당 서버에 대한 DNS 요청이 사용자의 의해 더 자주 전송 되기 때문에 웹 서버는 DNS캐시에 더 자주 나타납니다.

P23. Consider distributing a file of  $F$  bits to  $N$  peers using a client-server architecture. Assume a

fluid model where the server can simultaneously transmit to multiple peers, transmitting to each peer at different rates, as long as the combined rate does not exceed  $u$ .

server  패킷이 앞에 들어온 것과 뒤에 들어온 것과 시간은 다를 수 있다. 하지만 이것을 무시하고 동시에 똑같은 wait 시간으로 보낼 때는 전체적 모델

a. Suppose that Specify a distribution scheme that has a distribution time of  $NF/u$ .

$$NF/u_s$$

$\downarrow_{min} \geq u_s/N$  을 가정하고 있다.

이 속도는 각 클라이언트의 다운로드 속도 보다 늦습니다.

그러므로 각 클라이언트는  $u_s/N$  비율로 수신할 수 있습니다.

$u_s/N$  비율로 수신하므로 각 클라이언트가 전체 파일을 수신하는데 걸리는 시간은

$$F/(u_s/N) = NF/u_s$$

모든 클라이언트가  $NF/u_s$ 로 파일 받으므로 전체 배포 시간은  $NF/u_s$  입니다

$$\text{조건이 바뀔} \quad d_{\min} \leq u_s/N$$

b. Suppose that Specify a distribution scheme that has a distribution time of  $F/d_{\min}$

각 클라이언트는  $d_{\min}$  의 속도로 수신하므로 각 클라이언트가 전체 파일을 수신하는 시간은  $F/d_{\min}$  입니다.

모든 클라이언트가 파일을 수신하므로 전체 배포 시간은  $F/d_{\min}$  입니다.

c. Conclude that the minimum distribution time is in general given by  $\max\{NF/u_s, F/d_{\min}\}$

$$D_{C-S} = NF/u_s$$

$$D_{C-S} \geq \max \{ NF/u_s, F/d_{\min} \}$$

$u_s/N \leq d_{\min}$  이므로  $\left\{ \frac{NF}{u_s}, \frac{F}{\frac{u_s}{N}} \right\} \Rightarrow D_{C-S} \geq NF/u_s$  식을 얻을 수 있다.  
그러나 (a) 에서  $D_{C-S} \leq NF/u_s$  를 갖는다.

$$D_{C-S} = NF/u_s \quad (u_s/N \leq d_{\min}) \quad \dots \textcircled{1}$$

$$D_{C-S} = F/d_{\min} \quad (u_s/N \geq d_{\min}) \quad \dots \textcircled{2}$$

$$\therefore D_{C-S} = \frac{NF}{u_s}$$

P24. Consider distributing a file of  $F$  bits to  $N$  peers using a P2P architecture. Assume a fluid model. For simplicity assume that  $d_{\min}$  is very large, so that peer download bandwidth is never a bottleneck

압록드 스피드와 상인

a. Suppose that  $u_s \leq (u_s + u_1 + \dots + u_N)/N$ . Specify a distribution scheme that has a distribution time of  $F/u$ .

$$u = u_1 + u_2 + \dots + u_N \text{ 라고 가정하면 } u_s \leq (u_s + u)/N$$

파일을  $N$  부분으로 나눈  $i$  번째 부분은  $(u_i/u)F$  크기를 갖습니다.

서버는  $i$  번째 부분을  $r_i = (u_i/u)u_s$  속도로 Peer  $i$  에 전송합니다.

$$r_1 + r_2 + r_3 + \dots + r_N = u_s \text{ 이므로}$$

Aggregate server rate는 서버의 링크 속도를 초과하지 않아야 합니다.

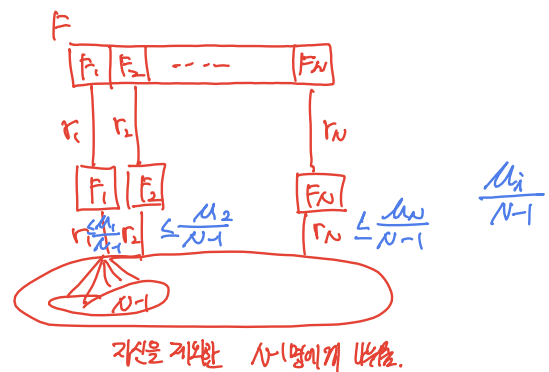
각 Peer  $i$  가 수신한 비트를  $r_i$ 의 비율로  $N-1$  Peer 각각에 전달해야 합니다.

Peer  $i$ 의 총 전달 속도는  $(N-1)r_i$  입니다.

$$(N-1)r_i = (N-1)(u_i \cdot u_s)/u \leq u_i$$

Peer  $i$ 는 총 속도로 비트를 수신해야 합니다.

결국 각 Peer는  $F/u_s$ 의 속도로 파일을 받습니다.



$$u_s \geq \frac{u}{N-1}$$

$$r_{ex} = (u_s - \frac{u}{N-1})/N$$

$$r_i + \sum_{j \neq i} r_j + r_{ex}$$

b. Suppose that  $u_s \geq (u_s + u_1 + \dots + u_N)/N$ . Specify a distribution scheme that has a distribution time of  $NF/(u_s + u)$

서버로부터 전송 비율은  $u_s \geq \frac{(u_s + u)}{N}$

이 배포 방식에서 서버는  $i$ 번째 Peer ( $i = 1, \dots, N$ )로 비트를  $r_i$  속도로 보냅니다.

각 Peer  $i$ 는 비율  $r_i$ 에 도달하는 비트를 다른  $N-1$ 개 Peer 각각에 전달합니다.

또한 서버는  $(N+1)$ 번째 부분에서  $r_{N+1}$ 의 속도로  $N$ 개의 Peer 각각에 비트를 보냅니다.

Peer는  $(N+1)$ 번째 부분 비트는 전달하지 않으므로 계산상에서 양변에  $(N+1)$ 번째 부분을

더해준다면 식은 변함이 없습니다.

$$r_1 + r_2 + \dots + r_N + N r_{N+1} = \frac{u}{(N-1)} + u_s - \frac{u}{(N-1)}$$

$$= u_s$$

서버의 총 전송 속도는 링크 속도를 초과하지 않습니다.

Peer  $i$ 의 총 전송 속도는  $(N-1)r_i = u_i$

따라서 각 Peer의 전송 속도는 링크 속도를 초과하지 않습니다.

$$\frac{u}{(N-1)} + \frac{\left(u_s - \frac{u}{(N-1)}\right)}{N} = \frac{u_s + u}{N}$$

따라서 각 Peer는  $NF/(u_s + u)$ 의 파일을 받습니다.

c. Conclude that the minimum distribution time is in general given by  $\max\{F/u_s, NF/(u_s+u_1+\dots+u_N)\}$

$$D_{P2P} \geq \max \left\{ \frac{F}{u_i}, \frac{NF}{(u_s+u)} \right\}$$

(a)번과 (b)번의 답을 이용하면 (c)번 문제에서 알하고자 하는 것은

$$D_{P2P} = \max \left\{ \frac{F}{u_i}, \frac{NF}{(u_s+u_1+u_2+\dots+u_N)} \right\}$$