

Web Server의 원활한 데이터 입출력을 위한 Buffer Management에 관한 연구

홍창호* 박경배 이승철
중앙대 전기공학과

A Study on the Buffer Management for Web Server

Hong, Chang-ho Park, Kyung-bae Lee, Seung-chul
Dept. of Electrical Eng, Chung-Ang University

Abstract - 근래 Web Server의 안정된 운용의 중요성이 급증하는 Network Traffic과 더불어 점차 증대되고 있다. 기본적인 Network 구성망은 Ethernet으로 표준적인 프로토콜은 TCP/IP로 기반을 잡고 있다. 날로 증가하는 Network Traffic을 효과적으로 처리하기 위해서는 Network의 bandwidth 및 서버가 처리할 수 있는 데이터 양을 분석하여 병목현상을 줄이고 데이터의 흐름을 원활히 하기 위한 데이터 buffer들의 설치와 적절한 운영 방법에 관한 연구가 필요하다. 본 논문에서는 Web Server로부터 image 데이터를 전송 받아 이를 원활히 display하기 위한 Application Layer에서의 buffer 크기 및 제어방법에 관한 연구를 수행하고 특히 연속적인 이미지의 전송과 재생에 관한 buffer management에 관하여 고찰한다.

1. 서 론

근래에 들어 케이블, ADSL과 같은 고속 인터넷망의 발전으로 인터넷에서의 Web의 사용횟수와 데이터의 양이 급증하게 되었다. 이러한 고속 인터넷망의 확장으로 인하여 Web서버는 대용량의 화상과 음성과 같은 멀티미디어 파일도 Internet을 통하여 client들에게 실시간으로 전달하게 되었다. 멀티미디어 파일의 경우 통상 그 사이즈가 매우 크고 또한 일반적으로 동시에 다수에게 각기 다른 멀티미디어 파일을 네트워크를 통하여 전송하는 경우가 많다. Web서버로부터 제공된 파일은 OSI 7계층을 단계별로 통과한 후 네트워크를 통하여 client에게 전달되고 client에서는 그 역변환을 수행하게 된다. 이러한 과정 중 여러 단계의 buffer를 거치게 되며 따라서 원활한 데이터의 전송을 위해서 buffer management를 적절히 수행하여야 한다.

본 논문에서는 특히 다수의 client에게 서비스하는 Web Server로부터 동영상 이미지를 전송받아 display하는 client가 전송된 동영상을 끊김 없이 원활히 보기 위해 필요한 application level에서의 buffer management에 대하여 고찰하기로 한다.

2. 본 론

2.1 계층구조에 따른 Buffer Management

OSI 7계층은 각각 물리층, 데이터링크층(LLC, MAC) 네트워크층, 트랜스포트층, 세션층, 프리젠테이션층, 애플리케이션층으로 구성되어 있다. 이를 TCP/IP 개념층(4계층 모델)으로 구성하면 네트워크 인터페이스층, 인터넷층, 트랜스 포트층, 애플리케이션층으로 크게 분류 할 수 있다. 애플리케이션층은 FTP, SMTP, DNS, TEL-NET, HTTP등과 같은 서비스로 이루어져 실제 사용자가 이용할 수 있도록 제공되고 각각의 서비스에 따라 buffer management가 달라지게 되며 이러한 서비스를 제공하는 서버와 서비스를 이용하는 클라이언트에서도 각기 다른 버퍼 관리가 필요하다.

2.1.1 Link Layer Buffer

이 Layer에서의 buffer management는 Ethernet packet의 구성과 송수신을 위해서 사용된다. Ethernet card내의 출력 버퍼의 크기는 통상 한 개의 Ethernet Packet의 크기인 1514byte 보다 조금 큰 1536 byte 정도이고 입력 버퍼는 각 카드들에 따라 차이가 있으나 64Kbyte 정도의 크기로 카드내에 메모리가 할당되어 있다.[1] 이 계층에서의 버퍼는 그림 1에서 보인바와 같은 링형 버퍼로 버퍼내의 데이터의 입력값과 출력값의 위치를 표시하는 current page 레지스터와 경계 포인터를 내장하고 있어 데이터의 overlap을 막는다. 이러한 형태로 구성된 buffer는 카드내의 전용 컨트롤러에 의해 제어 된다.

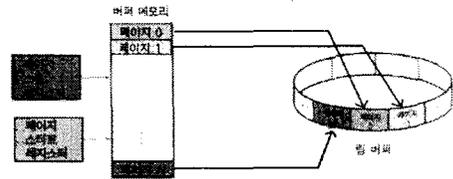


그림 1. Ethernet 카드에서의 링버퍼

2.1.2 Network & Transport Layer Buffer

네트워크 및 트랜스포트층에 해당하는 TCP/IP 층은 여러 방법에 의해서 buffer management가 이루어지는데 Berkeley networking code를 기초로 설계된 MBufs(Memory Buffers)가 그 한 예이다.[2][3] 그림 2는 전송시의 MBufs 구조로서 buffer 하나의 크기는 128byte이고 이중 헤더는 20byte로써 네개의 4byte fields와 두 개의 2바이트의 field를 가지고 있다. Mbufs는 m_next와 m_nextpkt를 통하여 다른 Mbufs들과 link되어 있다. m_len은 현재의 MBufs에서의 데이터의 길이를 m_type은 socket의 이름과 같은 특별한 정보를 나타낸다. 20byte의 header 다음에는 16byte의 destination IP address 와 port number를 가지게 된다. 크기가 큰 데이터의 취급을 위하여 M_flag를 이용하여 2048byte의 Cluster(external buffer)를 사용할 수 있다. 이러한 Mbufs는 데이터를 상위레벨의 프로토콜에서 하위레벨의 프로토콜로 캡슐화할 때 혹은 그 반대 과정에서 사용하게 된다. 그림 2는 MBufs 구조의 한 예이다.

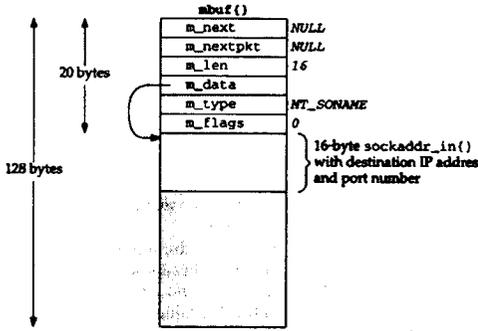


그림 2. MBufs (Memory Buffer)

2.1.3 Application Layer Buffer

애플리케이션층의 버퍼는 각 애플리케이션의 용도에 따라 그 크기 및 제어방법이 상이하다. 본 논문에서는 여러 애플리케이션 중 특히 Web Server에서 연속적인 image 데이터를 원활하게 display하기 위한 main memory 상에서의 buffer management에 관하여 고찰하기로 한다.

그림 3은 Web서버에서 client 컴퓨터로 이미지를 전송하는 단계를 frame 단위로 나타낸 그림이다. client의 요구가 Web서버에 전달되면, 저장되어 있는 이미지 파일을 서버의 main memory buffer에서 순차적으로 재 조합한 후 인터넷을 통하여 client의 main memory로 전달된다. 네트워크를 통하여 전달되는 이미지 프레임의 수가 이미지의 재생 속도 보다 빠를 경우에는 client의 하드디스크와 같은 저장장치에 저장한다. 본 논문에서는 네트워크를 통하여 전송되는 이미지 파일의 속도가 재생 속도보다 느릴 경우의 한해서 고찰한다.

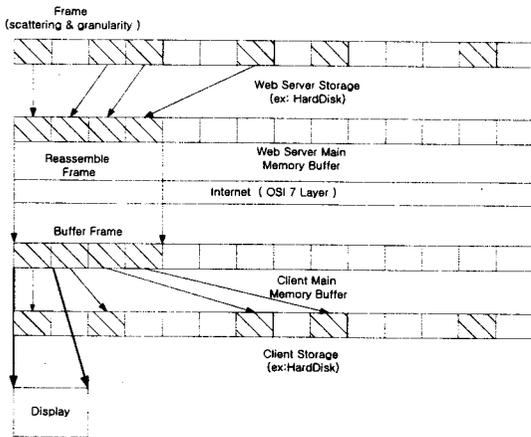


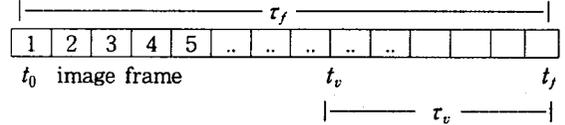
그림 3. Web서버에서 client로 전송되는 이미지의 저장 및 재생

2.2. Application Layer에서의 Main Memory Image Buffer Management

Application Layer에서의 메인 메모리에 설치되는 image buffer의 관리에 대한 분석을 위해 첫째, web server는 복수의 client를 serve하며 둘째 packet들의 경로(Route) 변경으로 인한 변수로 인하여 image packet의 도착시간은 random한 값을 가지며 따라서 packet들의 interarrival time은 일반적으로 지수 분포(exponentially

distributed)를 갖는 것으로 나타나 있다.

그림 4는 client의 Application Layer에서의 image buffer의 예를 나타낸 것으로 서버로부터의 image의 전송속도가 display 속도보다 느릴 경우 동영상 image를 일정시간 끊김 없이 보기 위한 buffer management에 관하여 고찰하기로 한다.



- t_0 : 이미지 전송 시작시간
- t_v : 이미지를 보기 시작하는 시간
- t_f : 이미지 재생이 완료되는 시간
- $\tau_v(\text{sec})$: 이미지 재생시간

그림 4 client의 Application Layer에서의 Image Frame Buffer

시간 t_v 로부터 시작하여 τ_v 시간동안 이미지를 끊김 없이 볼 수 있을 확률은 τ_f 동안 N_{df} 프레임을 Main Memory image buffer에 모두 채울 확률과 같다. packet들의 도착을 poisson arrival로 가정하면 frame의 도착도 poisson arrival로 되어 t 초 동안 k 개의 frame이 도착할 확률은

$$P[N(t) = k] = \frac{(\lambda' t)^k e^{-\lambda' t}}{k!}$$

이 되고 따라서 시간 τ_f 동안 N_{df} 이하의 frame이 도착할 확률 p' 은

$$p' = e^{-\lambda' \tau_f} \left(1 + \frac{(\lambda' \tau_f)^1}{1!} + \frac{(\lambda' \tau_f)^2}{2!} + \dots + \frac{(\lambda' \tau_f)^{N_{df}}}{N_{df}!} \right)$$

$$= e^{-\lambda' \tau_f} \sum_{k=0}^{N_{df}} \frac{(\lambda' \tau_f)^k}{k!}$$

이 되어 τ_v 동안 동영상 이미지를 끊김 없이 볼 수 있을 확률은

$$p_o = 1 - p'$$

가 된다.

또한 p_o 의 확률로 끊김 없이 이미지를 보기 위해서는 t_0 후 $(\tau_f - \tau_v)$ 시간 후부터 display를 시작해야 된다.



그림 5. client로 전송된 이미지의 재생 과 Buffering

2.2.1 적용 데이터의 분석

그림 5는 1분 56초 길이의 132,295,680byte의 압축하지 않은 176 x 144의 resolution과 24bit color, 총 1740 프레임은 가지는 연속적인 이미지(avi) 화일을 15 frame/s로 재생하는 이미지 버퍼 및 이미지 재생 진행 상황을 나타내고 그림6은 이미지를 capture하여 전송하기 위한 system setup을 보인다. 이 이미지 파일을 client에서 디스플레이를 통하여 데이터 전송과 동시에 재생을 진행하기 위해서는 1140KB/sec 이상의 인터넷을 통한 데이터 전송률이 필요하다. 본 논문에서는 서버에 접속자가 많을 경우를 simulation 하기 위해 압축하지 않은 대용량의 avi파일을 사용하였다. image buffer를 채우는데 약 2분45초의 시간이 소요되었다. 이때 frame의 arrival rate λ' 은 10.54(arrivals/sec)가 되고 따라서 이미지의 재생 buffer를 채우기 시작한 후 약 49초 후 부터 시작하도록 할 경우 image를 끊김 없이 연속적으로 1분 56초 동안 display할 수 있는 확률은 $p_b = 1 - p'$

$$= 1 - e^{-10.54 \times 165} \sum_{k=0}^{1740} \frac{(10.54 \times 165)^k}{k!}$$

이 된다. 상기 식에서 p' 은 image buffer의 크기가 충분히 커서 0에 수렴하였고. 따라서 p_b 는 1에 근접하여 그림 5에서 실제 buffering과 재생이 buffer의 끝부분에서 매회 거의 일치하는 것을 관측할 수 있었다.

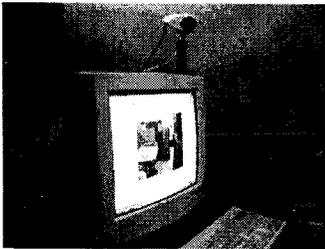


그림 6. 이미지의 capture를 위한 카메라 장치

3. 결 론

본 논문에서는 Web 서버와 client간의 원활한 멀티미디어 데이터 입출력을 위한 네트워크의 계층별 buffer management에 대해 고찰하였고 client에서 web 서버를 통하여 전달되는 동영상 이미지를 끊김 없이 보기 위해 필요한 application level에서의 buffer management의 예를 보였다.

향후 보다 다양한 네트워크 traffic 환경에서 buffer size와 이미지의 resolution 변경에 따른 display image의 품질 변화에 관한 연구를 수행할 예정이다.

본연구는 네트인텔리젠스(주)의 지원에 의하여 수행되었음.

[참 고 문 헌]

[1] Realtek.inc "RTL8019AS" Ethernet card develop reference

[2] GaryR. Wright W.Richard Stevens "TCP/IP Illustrated, Volume 2(i) The Implementation"
 [3] GaryR. Wright W.Richard Stevens "TCP/IP Illustrated, Volume 2(ii) The Implementation"
 [4] 서원일, 박용운, 정기동, "연속 미디어 서버를 위한 이중 모드 버퍼 캐쉬 관리 기법", 한국 정보처리학회 논문지 제6권 제 12호 (99.12)
 [5] 양택훈, "연속미디어 제한 블록할당에서 전송율과 소비율에 기반한 버퍼크기에 관한 연구", 중앙대학교 컴퓨터 공학과 대학원 제84회 졸업논문
 [6] D.J. Gemmell, H.M. Vin, D. D. Kandlur P.V. Rangan, "Multimedia Storage Server: A Tutorial," IEEE computer Magazine, May 1995, pp40-49