

유닉스/리눅스 운영체제의 Memory Mapped File 성능 비교·분석

이형봉⁰, 정연철, 권기현

강릉대학교 컴퓨터공학과⁰, 호남대학교 컴퓨터게임학과, 삼척대학교 정보통신공학과
 hblee⁰@kangnung.ac.kr, ycjeong@honam.ac.kr, kweon@samcheok.ac.kr

A Comparative Analysis on the Performance of Memory Mapped File of UNIX/LINUX Operating Systems

Hyung Bong Lee⁰, Yeon Chul Jeong, Ki Hyun Kweon

Dept. of Computer Engineering, Kangnung Univ.⁰, Dept. of Computer Game, Honam Univ.,
 Dept. of Information and Communication Engineering, Samcheok Univ.

요 약

MMF(Memory Mapped File)는 대응된 주소 영역을 접근함으로써 파일에 대한 입·출력 효과를 얻는 기본적인 활용 외에, 커널 영역에 버퍼를 설정하여 입·출력 성능을 향상시키거나 파일의 효율적인 공유를 위한 수단 등 다양한 방법으로 사용된다. 기본 기능 활용의 대표적인 예 중의 하나가 웹 파일 서비스이다. 이 논문에서는 웹 서비스 성능향상을 위한 연구의 일환으로 몇 가지 유닉스/리눅스 시스템을 대상으로 MMF의 기본 기능에 대한 성능을 비교·분석하였다. 그 결과 시스템 별로 상당한 차이가 있다는 사실이 드러났으며, 최악의 경우에는 느려져 성능이 30% 이상 저하되는 경우가 발견되었다.

1. 서 론

MMF의 기본 기능은 그림 1과 같이 파일의 일부 또는 전부를 주소 공간에 대응시킨 후, 해당 주소 부위를 주기의 장치처럼 참조함으로써 파일에 대한 입·출력 효과를 얻게 해주는 데 있다[1,2]. 이러한 MMF 기본 기능의 주목적은 read()/write() 등 시스템 호출에 의한 경우보다 입·출력과 파일 공유에 있어서 편리성을 얻자는 데 있다. 이 때 파일 공유의 편리성은 대응된 주소 공간을 공유 메모리로 설정할 수 있기 때문에 얻어질 수 있다.

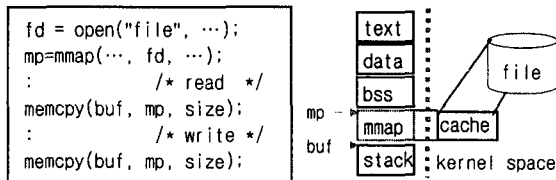


그림 1 Memory Mapped File(MMF)의 기본 기능

그러나, MMF의 기본 기능을 커널 영역의 버퍼로 활용하면 커다란 성능상 이득을 볼 수 있는 경우가 있다. 그 대표적인 예로 웹 파일 서비스를 들 수 있다[3]. 일반적인 유닉스/리눅스 시스템 호출을 사용한 웹 파일 서버의 파일 입·출력 방법은 그림 2와 같다[2]. 이 방법에서는 파일의 내용이 일단 사용자 영역의 버퍼로 복사되었다가, 다시 통신을 위한 커널 영역의 버퍼로 복사되기 때문

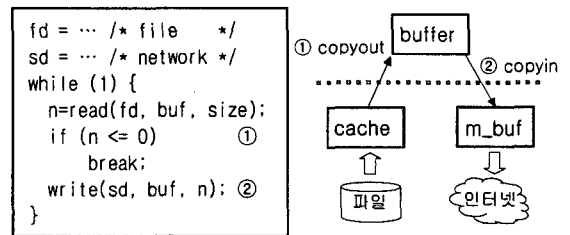


그림 2 일반 웹 파일 서버에서의 데이터 흐름

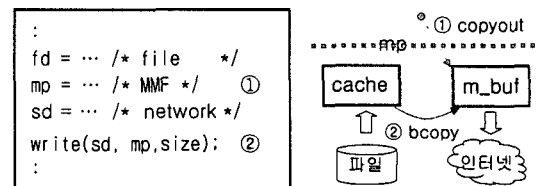


그림 3 MMF 서버에서의 데이터 흐름

에 두 번의 데이터 복사가 필요하다. 하지만 그림 3과 같이 MMF를 이용하면 커널 내에서 곧바로 통신 버퍼로의 복사가 이루어지므로 데이터 이동이 한 번의 복사로 완료된다.

이와 같이 MMF를 이용하여 웹 파일 서버를 구성한다면 어떤 경우라도 최소한의 성능 향상을 기대할 수 있다.

이 연구에서는 그림 3의 MMF 이점에 대한 실질적인 타당성을 검증하기 위해 현재 상용화되어 널리 사용되고 있는 몇 가지 유닉스/리눅스 시스템을 대상으로 MMF 성능을 비교·분석한다. 이를 위하여 2장에서 MMF 성능 측정 환경과 방법을, 3장에서 MMF 성능 측정 결과에 대한 비교·분석을 각각 논하고, 마지막 4장에서 결론으로 맺는다.

2. MMF 성능 측정 환경 및 방법

2.1 성능 측정 시스템 환경

MMF 성능 측정 대상으로 선정된 시스템은 표 1과 같이 유닉스 3종, 리눅스 1종으로 모두 상용화되거나 널리 사용되고 있는 것들이다.

표 1 MMF 성능 측정 대상 시스템

운영 체제	플랫폼	프로세서	메모리
IBM AIX 4.3 (UNIX)	RS/6000 WS140	PowerPC 604e 233MHz	64MB
SUN Solaris 5.7(UNIX)	Enterprize 250	UltraSparc-II 300MHz	128MB
Digital Tru64 UNIX 4.0F	Alpha DS20	Alpha EV67 500MHz	512MB
DebianLINUX 2.4.18	LG-IBM PC 8478	Pentium III 1GHz	756MB

2.2 측정 방법

이 논문에서는 MMF의 성능을 크게 두 가지 관점으로 분류하여 접근하였다. 첫 번째 관점은 그림 3과 같이 데이터 복사의 이점에서 얻는 성능이고, 두 번째 관점은 데이터 복사의 이점을 없기 위한 MMF의 설정 과정의 부담에 따른 성능이다. 데이터 복사에 따른 이점이 다소 있다고 하더라도 설정 비용이 크다면 그 MMF는 사용상 제한을 받을 것이기 때문이다.

위의 두 가지 성능을 측정하기 위하여, [표 1]의 각 시스템에서 아래의 측정 유형을 고안하였다.

■ 데이터 복사 경로 유형

서버로부터 클라이언트로의 데이터 전달 경로를 그림 4와 같이 파이프와 소켓 두 가지를 선정하고, 그림 2와 그림 3의 방법 즉, 사용자 버퍼를 거쳐서 이동하는 'normal' 방식과 MMF에서 곧바로 이동하는 'mmf' 방식을 각각 적용하였다. 여기서 파이프와 소켓 두 가지 이동 경로를 선택한 이유는 장치의 특성에 따른 MMF 성능 추이를 알아보기 위함이다(client는 로컬에 설치됨).

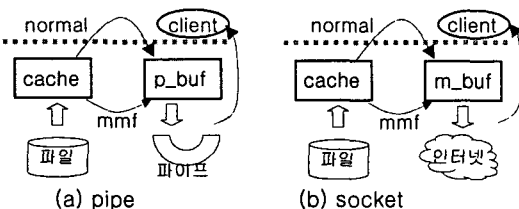


그림 4 데이터 복사 성능 측정을 위한 경로 유형

■ 데이터 접근 유형

데이터 접근 유형은 파일 개방과 MMF 설정을 파일 요구가 있을 때마다 새롭게 하는 'n_open' 방법과, 최초에 한 번만 하고 이후부터는 설정된 상태를 그대로 유지한 채 입·출력만 수행하는 'l_open' 방법으로 구분하였다.

■ 데이터(파일) 크기

데이터는 1K~30M의 다양한 크기의 파일을 사용하였으나 참조는 1,3,5M만 사용했고, 시스템 소요시간 측정 단위가 대부분 1/100 초이기 때문에 위의 각 파일에 대하여 1000번 내려받는 성능을 1회 측정 값으로 간주하였다.

표 2에 성능 측정 유형을 요약하였고, 그림 5에 표 2의 다양한 유형의 성능을 측정할 수 있는 프로그램을 보였다.

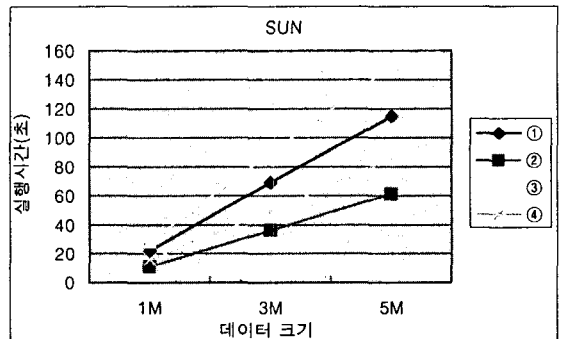
표 2 MMF 성능 측정 유형

파일접근방법	데이터 이동 경로	데이터 크기
n_open	pipe	normal...① 1K 700K
		mmf...② 10K 1M
	socket	normal...③ 30K 3M
		mmf...④ 50K 5M
l_open	pipe	normal...⑤ 70K 10M
		mmf...⑥ 100K 20M
	socket	normal...⑦ 300K 30M
		mmf...⑧ 500K

3. 성능 측정 결과 및 비교·분석

3.1 MMF에 의한 데이터 복사 성능 이득

그림 5에 표 1의 각 시스템에 대하여 표 2의 측정 유형 ①~④ 즉, 데이터 복사 성능 측정 결과를 보였다. 이 결과로부터 파이프나 소켓 등 장치에 관계 없이 모든 시스템에서 MMF를 사용했을 때 데이터 복사 성능이 우수함을 알 수 있다. 다만, normal 대비 MMF의 평균 성능 이득은 SUN이 47%, IBM이 24%, Digital이 32%, Debian이 20% 정도로 시스템에 따라 차이가 크다.



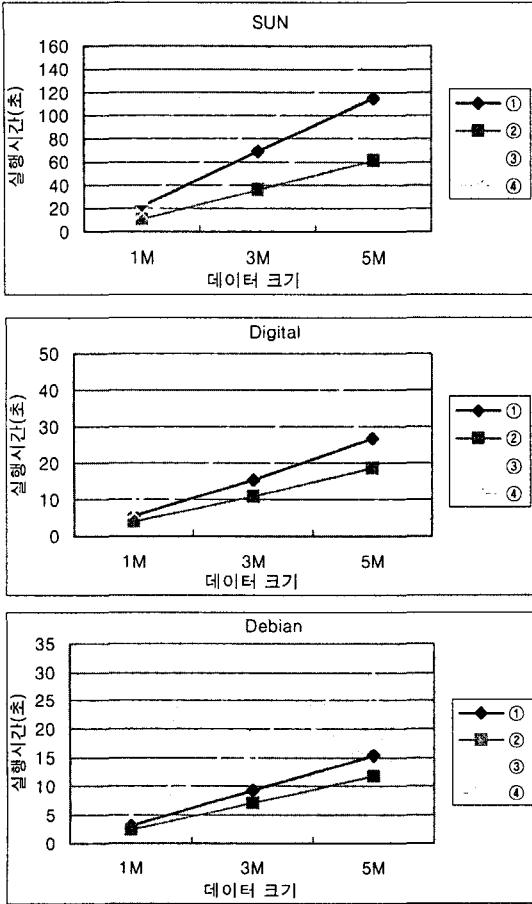


그림 5 MMF에 의한 데이터 복사 성능 측정 결과

3.2 MMF 설정에 따른 부담 성능

그림 6에 MMF 초기 설정 부담 성능을 측정하기 위해 표 2의 ⑤~⑧ 유형에 대한 성능 측정 결과를 보였다. 이 그림으로부터 SUN은 데이터 크기에 관계 없이 설정 부담을 감당할 수준이지만, Debian과 Digital은 부담이 크고(~30%), IBM은 데이터 크기에 따른 임계 특성을 가진다는 사실을 알 수 있다.

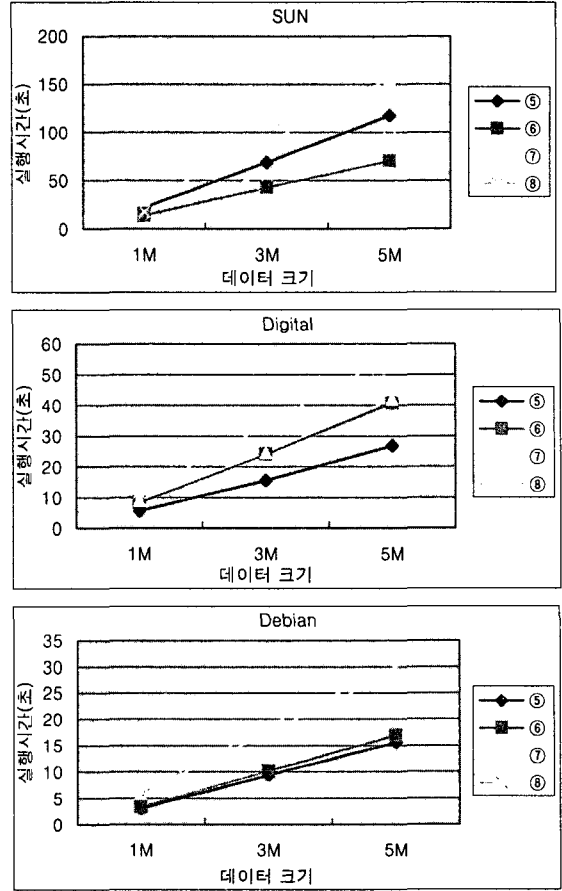
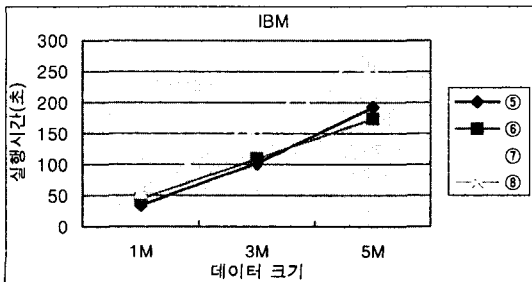


그림 6 MMF 설정 부담에 따른 성능 측정

4. 결론

이 논문에서, 유닉스/리눅스의 MMF 성능은 동일 계열의 운영체제에서 지원되는 기능이란 하나 그 운영체제가 탑재된 시스템에 따라 성능 특성이 상이함을 보였다. 즉, MMF가 다양한 분야에 활용될 수 있다는 이론적 배경도 중요하지만, 실제로 얼마만큼의 성능 이득이 얻어지는가는 더욱 중요하다. 이 논문에서 고안한 성능 측정 방안은 실제적이 MMF 활용에 기여할 수 있을 것이다.

참고문헌

- [1] Digital, "mmap()", Digital UNIX Reference Page Section 2: System Calls, pp.243-249, March 1996
- [2] 이형봉, 최형진, 차홍준, "Memory Mapped File을 이용한 웹 서비스 성능향상 연구", 한국정보과학회 2003 추계 학술발표대회, 제30권, 제2-3호, pp. 154-156, 2003년 10월.
- [3] Erich Nahum, Tsipora Barzilai, Dilip Kandlur, "Performance Issues in WWW Servers", IEEE Transactions on Networking, 10(2):2-11, Feb 2002