

고속 임베디드 저장 시스템을 위한 복제전환 기법

Mirror-Switching Scheme for High-Speed Embedded Storage Systems

변시우[†], 장석우^{*}

Siwoo Byun and Seok-Woo Jang

(2011년 2월20일 접수; 2011년 3월18일 심사완료; 2011년 3월20일 게재확정)

Abstract

The flash memory has been remarked as the next generation media of portable and desktop computers' storage devices. Their features include non-volatility, low power consumption, and fast access time for read operations, which are sufficient to present flash memories as major data storage components for desktop and servers.

The purpose of our study is to upgrade a traditional mirroring scheme based on SSD storages due to the relatively slow or freezing characteristics of write operations, as compared to fast read operations. For this work, we propose a new storage management scheme called *Memory Mirror-Switching* based on traditional mirroring scheme.

Our Mirror-Switching scheme improves flash operation performance by switching write-workloads from flash memory to RAM and delaying write operations to avoid freezing. Our test results show that our scheme significantly reduces the write operation delay and storage freezing.

Keywords: Mirror, Switching, SSD, RAM, Flash Memory

1. 서론

오래전부터 하드디스크(HDD: Hard Disk Drive)는 대부분의 정보 시스템에서 사용되는 절대적인 저장 미디어였다. 그러나 최근 임베디드 단말기, 스마트폰, 노트북, 서버 등에의 고급 저장 시스템으로서 플래시 메모리와 Solid State Drive (SSD)가 급부상중이다.[1,2]

하드 디스크는 고용량 저장 능력과 저렴한 저장 비용이 장점이나 기계적인 회전 구조로 인하여, 소음, 전력, 속도, 내충격성 등에서 많은 단점을 내포하고 있다. 반면 플래시 메모리는 이러한 하드 디스크에 비하여 저용량성과 상대적으로 높은 저장 비용이 단점이 되고, 소음, 전력, 속도, 내충격성 등에서 우수한 장점을 가지고 있다. 즉, 가격과 저장 용량을 우선시 한다면 하드 디스크, 휴대성을 겸비한 처리 속도를 우선시 한다면 플래시 메모리가 선택 되게 된다.

플래시 메모리는 비휘발성 저장이 가능하다는 측면에서 하드 디스크와 유사하지만, 내충격성, 휴대성, 접근속도, 무진동성, 무소음성, 부품크기 측면에서 비교할 수 없을 정도로 매우 우수하다. 전력 소모량도 10mA 정도인데, 이는 저전력 메인 메모리의 1/8정도에 불과하다. 또한, 메인 메모리와 비교하면, 메인 메모리의 최대 약점인 휘발저장성을 해결했다는 측면에서 더 우수하므로, 차세대 기억장치라고 불려왔다. 다만, 일반 메인 메모리와는 달리, 쓰기와 소거 연산에 상당히 많은 시간이 더 소모되며, 쓰기 횟수가 최대 1,000,000 번 정도로 제한되는 고유한 특성(약점)이 있다. 일반적으로는 충분한 사용 수명이나, 이를 넘기면 더 이상 쓰기 연산을 수행할 수 없다. 단, 읽기 연산은 계속하여 수행이 가능하다.[3,4] 또한, 플래시는 일반 메인 메모리와 달리 바로 쓰기 연산을 수행할 수 없고, 이전에 미리 블록 단위로 포맷 개념의 소거 연산을 수행한 후 쓰기를 수행할 수 있다. 즉, 플래시 메모리는 PC의 메인 메모리처럼 Update-In-Place (제자리 덮어 쓰기)가 불가능한 Update-Out-Place 구조이다.

[†] 안양대학교 디지털미디어학과
E-mail : swbyun@anyang.ac.kr,
^{*} swjang@anyang.ac.kr (교신저자)
TEL : (031)467-0922

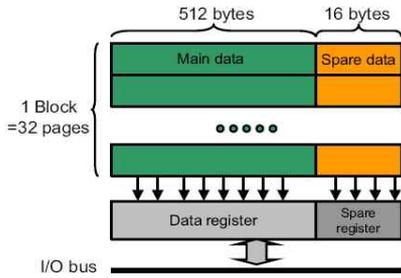


Fig. 1 Samsung Flash Memory

본 연구에서는 이러한 플래시 메모리에 대하여, 저장 장치로서의 약점을 보완하면서 장점을 활용하여, 최종적으로 임베디드 시스템, PC 등에서 입출력 성능을 최대화할 수 있는 효과적인 데이터 저장 기술을 제안한다.

2. 메모리 미러-스위칭 기법의 제안 배경

2.1 플래시 메모리의 I/O 특성 분석 실험

전술한 SSD는 첨단 플래시 메모리를 사용하기 때문에, 매우 빠르다고 알려져 있지만, 실제 사용해 보면 불만족스럽다고 한다. 먼저, 읽기의 경우는 무려 3배 가까운 속도의 향상이 나타난다고 한다. 이렇게 높은 접근 성능의 이유는 SSD는 HDD의 기계적 동작부분이 전혀 없기 때문이다.[2] 즉, 하드 디스크의 물리적/기계적인 접근 시간(약 8~10 ms)인, head의 seek time(헤드 모터 이동 시간)과 디스크 원판의 rotational delay(회전 모터의 섹터 지연 시간)가 전혀 없기 때문이다. 그러나 실제 사용자가 장기간 사용해 보면, 대부분의 SSD의 실제 체감속도는 아주 고가의 SSD를 제외하면, 기대만큼 빠르지는 않다는 것이다. 물론, 읽기 속도는 매우 개선되어 충분한 속도감을 느낄 수 있다. 즉, 이것은 읽기 위주의 부팅이나, 문서 로딩에 국한되는 것이다.

이러한 느린 체감 성능의 가장 큰 이유는, 일반 사용자 환경에서는 읽기만 하는 것이 아니고, 쓰기 연산도 병행되기 때문이다. 따라서 쓰기 비중이 크다면 이전에 장착된 하드 디스크와 비슷하거나 오히려 낮은 성능이 나타난다. 즉, 사용자의 작업은 읽기와 쓰기 연산이 임의로 혼합되어 있으므로, 전체 성능에서는 불만족스럽게 된다. 또한, SSD는 데이터가 별로 없는 제품구입 초기에는 빠르다가도, 이후에 점점 더 데이터가 많이 쌓여갈수록 계속 느려지는 “성능 열화현상”이 나타난다. 이는 SSD에 데이터가 일정용량이상 누적되게 되면, 느린 쓰기 특성과 관리 모듈의 overhead가 급속히 증가하기 때문이며, 이때부터 저장성능이 서서히 떨어진다. 이러한 열화현상은

SSD뿐 아니라 USB용 플래시 메모리에서도 나타난다. 실험결과 전체 SSD 영역에서 60%를 채우면 가비지컬렉션(Garbage Collection) 오버헤드가 10%증가하고, 90%정도를 채우면 오버헤드가 50%가 증가한다고 한다[2].

SSD의 또 다른 문제점은 실제 사용자가 작업하다가, 저장 연산이 집중되는 경우, SSD가 평소 속도보다 상당히 느린 속도(10배)로 반응한다는 문제이다. 물론, 가끔씩 발생하지만, 하드 디스크에 비해서는 상당히 빈번히 발생하고, 홀딩 시간도 하드 디스크의 보다 훨씬 더 길다. 이 현상은 물론 읽기 연산에서는 거의 발생하지 않는다. 하지만, 소프트웨어를 설치하거나, 특히, 큰 콘텐츠를 복사 받는 도중, 심한 업데이트 도중에, 윈도우 운영체제가 심한 경우 10초 썩 정지(흔히 프리징[freezing]이라고 함) 되는데, 이러한 체험 사례는 인터넷의 개인 블로그에서도 심각하게 보고되어 있다.

자체 연구[5]에서도, 이러한 프리징 현상을 확인할 수 있었다. 즉, 이러한 불안정한 응답성은, 아래의 그래프에서 확인 가능하였다. 실험환경은 10KB~500KB 까지 10KB 단위로 50개의 저장 스트림을 발생시켜서 응답시간(Response Time)을 측정한 결과이다. 참고로 일반 시스템 환경에서는 하루 동안 수정이나 읽기가 수행된 파일의 96% 이상이 500KB 미만의 데이터 파일이다. 실험결과, 그래프에서 다수의 응답지연 피크(Peak)가 발생하는데, 스트림-2의 경우처럼 한번 홀딩이 되면 연이어서 계속(4차례) 프리징이 발생한다는 것도 심각하다. 이러한 프리징의 이유로는, 설정된 쓰기 블록의 크기외에도, 쓰기 연산의 집중으로 인하여, SRAM 등의 버퍼가 포화되어, 더 이상 지연시키지 못하고, 버퍼에서 밀려 있던 플래시 메모리의 소거 작업에 따른 웨어레벨링 연산과 느린 CPU, 카피 오버헤드, 구동 소프트웨어 등의 복합적인 요인 때문이다.

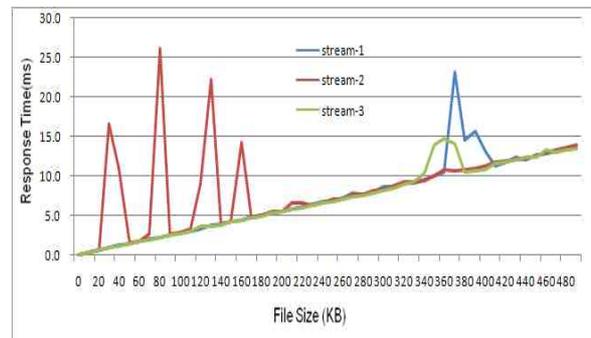


Fig. 2 Test Results (SSD Write Stream)

2.2 미러 스위칭 기법의 제안

전통적으로 하드 디스크 저장 시스템 업계에서는 저장 성능과 응답 성능의 문제해결을 위하여 RAID(Redundant Array of Independent Disks)기술 [6]을 사용하여 왔었다. RAID시스템은 여러 하드 디스크 드라이브의 집합을 하나의 저장 장치처럼 다룰 수 있게 하고, 장애가 발생했을 때 데이터를 잃어버리지 않게 하며, 각각에 대해 독립적으로 동작할 수 있도록 하는 저장 시스템이다. 마찬가지로 SSD도 성능과 안정성을 모두 개선하여야 하므로, 이 RAID 기술의 활용이 필요하다.

이러한 하드 디스크 기반의 RAID 기술을 SSD에 그대로 적용해도, 어느 정도의 성능 개선과 응답 성능 향상을 얻을 수 있다. 그러나 읽기 및 쓰기 성능이 대칭이 되는 하드디스크에 비하여, 10배 정도로 쓰기가 느려서 극심하게 비대칭인 SSD를 하드디스크와 동일한 기법으로 저장 및 검색하면 심각한 I/O 문제가 발생한다.

즉, 플래시 메모리의 장점도 못 살리고, 단점도 극복하지 못하게 되어, 최악의 I/O 성능 저하로 나타날 수 있다. 단일 매체로 구성된 RAID는 아무리 그 구성 방법을 바꾸어도, 그 단일 매체의 자체의 본질적인 약점을 제대로 극복하기는 어렵다. 예를 들어, HDD만으로 혁신적인 RAID기법을 개발하더라도, HDD의 구성만 달라질 뿐, 본질적인 기계적인 약점은 그대로 내재되어 있기 때문이다. SSD의 경우도 마찬가지이다. 반드시 서로의 약점을 보완해줄 성질이 다른 매체 RAID 기술이 필요하다.

본 연구에서는 이러한 관점에서 기존의 RAID-1 기술을 개선하고, 고속 검색용 SSD의 약점을 보완하기 위하여, RAM 메모리를 융합하여, 전체적인 시너지 효과를 낼 수 있는 새로운 미러 스위칭 기법을 제안 한다. 본 논문에서는 미러링된 스토리지 제어에서 읽기 연산은 어느 한쪽의 빠른 드라이브를 통하여 수행된다. SSD의 읽기 속도는 매우 빠르며 별 병목현상 문제가 없으므로 간략히 서술하며, 본 논문의 이슈인 저장 즉 쓰기 연산에 대하여 설명하고자 한다.

제안하는 미러 스위칭 기법을 활용하여 전술한 SSD 프리징 현상을 감소시키며 성능 향상을 할 수 있다. 즉, 일반적으로 SSD의 응답시간이 피크치로 급상승할 때도, RAM의 매우 빠른 매체 특성상 응답시간은 안정적이며 정상적이다. 다만, 장기간 저장이 되지 않는 단점을 고려하여, 배터리 백업을 이용하여 이를 보완하는 것이 중요하다.

일상적인 서버의 데이터에서 하루에 변화되는 데이터의 양은 전체량의 1%로, 300G정도의 일반

디스크 기준으로 보면, 3G 정도이다. 이러한 상호 보완적인 특징을 잘 활용하여, 특성이 상반된 두 매체(SSD와 RAM)를 미러링하면, SSD를 미러링하는 것보다 빠른 응답시간을 얻을 수 있다. 기존의 하드 디스크 기반 RAID-1(미러링) 기술은 안정성과 처리 성능에 일대 혁신을 이룬 상용화된 매우 중요한 기술이다. 즉, 원래 기존 미러링의 목적이 동일한 매체를 동시에 복제하여 안정성을 높이는 것이라면, 미러 스위칭은 좀 더 나아가서, 프리징의 충격까지 흡수시켜서 최종 성능까지 높이는 새로운 RAID 기술이다.

또한, 전술한 대로, SSD를 빠른 입출력을 위하여 기존의 RAID에 그대로 적용하게 되면, 많은 성능과 안정성에 심각한 문제가 발생하게 된다. 그 주요 원인은 read/write 처리 시간이 비슷한 하드 디스크에 비하여, 플래시 메모리 SSD는 read/write 처리 시간의 매우 차이가 나는 극심한 비대칭적 구조를 가지고 있으므로, I/O의 밸런스가 제대로 맞지 않기 때문이다. 이러한 수배에 달하는 심한 읽기/쓰기의 밸런스의 차이는 여러 개의 드라이브를 길게 일렬 연결(RAID-0, 스트라이핑)하여 어느 정도 극복할 수 있다. 즉, 일렬 배열 드라이브에 비례하여 성능은 점차 높아질 수 있다. 본 연구의 시스템 구성도는 아래와 같다. 아래 그림에서 사용자의 Read/Write 요청연산은 Storage Manager를 통과한 후, 미러링된 Safe Mirror-Write 연산과 스트라이핑된 Fast Mirror-Read 연산으로 Redirection 된다. 또한, RAID 기술과 더불어 프리징 장애를 방지하고, I/O성능을 더 가속시키려면, 프리징 징후를 빨리 파악하고 즉각 부하를 밸런싱할 수 있는 Mirror- Operation Manager가 필요하다.

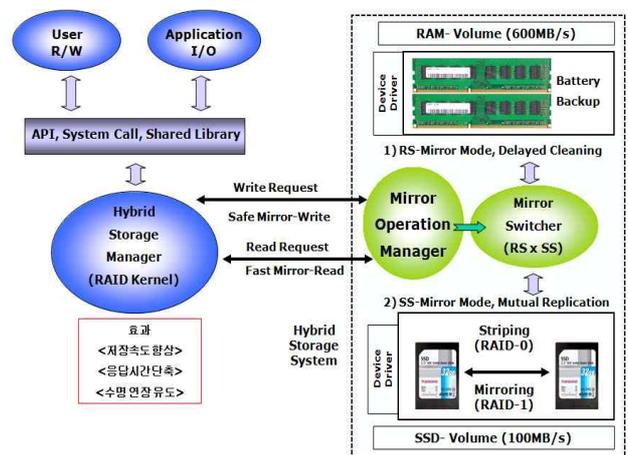


Fig. 3 Memory Mirror-Switching System

즉, SSD에서 소규모 데이터라도 많은 양으로 랜덤 쓰기를 아주 심하게 하면, 프리징 현상이 빈번히 발생되므로, 이러한 과밀 적체를 판단한 후 바로 적체를 부하 조절하여 피해갈 수 있도록 설계하여야 한다. 예를 들어 적체가 임계점을 초과하면서 응답시간이 급속히 느려지기 시작한다면, 오히려 RAM에 우선적으로 저장하고, SSD의 저장은 다른 대기 큐에 전환시키는 지연된 멀티큐 스케줄링이 효율적이다. 이렇게 지연된 SSD 큐잉은 추후에 쓰기 연산이 몰아져서 나오므로, 수명 연장의 효과도 볼 수 있다. 즉, 랜덤 쓰기가 모여져서 순차쓰기로 유도되는데, 플래시 메모리 SSD도 순차쓰기가 빠르며, 특히 작은 크기의 파일일 경우 더 효과적이다. 이는 특히 중대형 데이터베이스 서버나 Multitasking 작업이 많은 사용자 환경에서도 예측가능하고 고르게 안정된 응답시간을 더 높은 수준으로 보장해 줄 수 있다. 제안한 미러-스위칭의 수행 알고리즘은 다음과 같다.

- ① 사용자의 프로그램이 SSD 스토리지에 대한 읽기 또는 쓰기 연산(operation)을 요청한다.
- ② 사용자가 입출력 요청을 접수되면, 요청된 연산은 RAID연산(미러링)으로 전환된다. 이를 관리하는 MOM(Mirror Operation Manager)은 입출력 요청연산을 자신의 연산-큐의 마지막에 삽입한다. 추후에 지금 요청한 연산이 완료되면 결과 값을 최종적으로 사용자 프로그램에 리턴한다.
- ③ 연산-큐에서 한 연산씩 차례로 꺼낸 후 연산의 종류를 분류한다. 읽기 연산이면, (4) MIRROR_READ을 수행하며, 쓰기 연산이면 (5) MIRROR_WRITE를 수행한다.
- ④ **MIRROR_READ:**
 - 1) SSD미러링은 동일한 데이터가 두 장치에 존재하므로, 속도 증강을 위하여, 각각 반으로 나누어서 읽기 명령을 수행한다.
 - 2) 두 SSD장치로 부터 모두 읽기 작업이 성공하면, 판독한 값을 Operation Manager에게 리턴한다.
 - 3) 만일, 한 SSD로부터 읽기 오류가 발생하면, 다른 SSD에 동일하게 읽기 요청을 한 후, 판독한 값을 Operation Manager에게 리턴 한다. 발생된 오류를 시스템에 보고 한다.
- ⑤ **MIRROR_WRITE:**
 - 1) SSD와 RAM을 대상으로 동시에 저장 연산을 수행한다. 여기서, SSD는 두 개를 사용하여, RAID-0(스트라이핑) 모드로 저장한다. 즉, 각

SSD는 저장 데이터를 반으로 나누어서 저장하며, RAM은 전체를 저장하여, 전체적으로는 데이터가 두 카피로 미러링되도록 한다.

- 2) 만일, 해당 장치로부터 쓰기 오류가 발생하면, 이 오류를 시스템에 보고 한다. 심한 경우 드라이브 장애 오류이면, 장애 SSD의 복구 요청을 RAID에 전달하고, 현재의 미러링 모드를 단일 드라이브 모드로 전환하여 계속 운영을 가능하게 한다.
 - 3) 쓰기 연산 작업이 성공하면, 완료 메시지를 MOM에 리턴 한다.
- ⑥ **MIRROR_SWITCH:**
- 1) SSD가 다른 사용자의 연산 처리로 작업이 진행중(busy)이면 유휴(idle)시까지 기다린다. 아니면 아래 단계로 바로 넘어간다.
 - 2) RAM을 통하여, 두 SSD에게 아직 저장하지 않은 나머지 절반의 데이터를 각각 복사한다. (이제 동일한 저장데이터가 RAM 및 두 개의 SSD에 모두 세 영역에 링크가 존재함)
 - 3) RAM의 여유 공간 확보를 위하여, 상호복사(Mutual Replication)된 데이터는 삭제한다.

3. 성능 평가

본 연구에서 제안된 미러 스위칭 기법의 저장 성능을 검증하기 위하여 실제 환경에서 작업 부하를 주고, 그 실험 결과를 분석해 보았다. 본 실험 수행시 앞서 제안한 미러 스위칭 기법(Mr-Switch)과 비교된 저장 관리 기법은 SSD를 1개만 사용하는 저장 기법(SSD-1), 가장 보편적이고 안정적인 SSD RAID 구성인 미러링 기법(SSD-Mr)이다. 또한, 메인 메모리 저장 시스템과의 성능 비교를 위하여, 부수적으로 RAM 디스크 저장 기법(RAM-1)을 추가하여 총 4가지의 저장 시스템을 비교 분석하였다.

3.1 실험 환경과 준비

1) 실험 시스템 구축

실험이 수행된 하드웨어 환경은 중소형 서버 환경에 가깝게 구성하였다. 즉, 보편적인 성능의 DELL 펜티엄4 CPU와 메인 메모리 4G, 운영체제는 윈도우 XP를 사용하였다. 언어는 비주얼 C++를 사용하였으며, 부수적으로 작업 부하 생성 모듈과 물리적인 실험후 통계 자료 수집의 편의상 CSIM[7] 시뮬레이션 라이브러리를 일부 사용하였고, 성능측정 부분은 모두 실제 SATA2로 연결된 2.5인치 SSD와 DDR2 RAM를 통하여 검출

되었다.

2) 저장 데이터의 구성

본 실험에 앞서서, 실험용 샘플이 되는 데이터가 필요한데, 실제 사용자의 윈도우 환경에서 사용 중인 모든 파일을 검색한 후, 사용 빈도가 가장 높은 순으로 나열해 보면, 96%이상이 500KB 미만의 작은 크기의 스몰 파일들이다. 이를 근거로 하여, 10KB부터 500KB 사이의 10KB 단위의 랜덤 파일을 생성하여 미리 저장하여 두었다. 물론, 이러한 크기의 설정은 일반 사용자의 작업 환경에 따라서 조금 다를 수 있으나 실험의 목적에 크게 영향을 미치지 않는다.

3) 워크로드 모델 및 평가 지표

다음으로 실험을 위한 저장 워크로드 생성 및 결과 분석 모델이 필요한데, 이는 CSIM에서 제공되는 폐쇄형 큐잉 모델(Closed Queuing Model)과 통계 라이브러리를 사용하였다. 이 모듈들을 통하여 초당 일정한 수의 쓰기 오퍼레이션을 생성하고, 이와 연관된 처리 시간과 성능을 측정하면 된다. 즉, 본 실험의 주요 성능 평가 지표는 오퍼레이션 처리치(operation throughput)와 응답시간(response time)이다. 오퍼레이션 처리치는 초당 몇 개의 오퍼레이션이 처리되었는지를 의미하고, 응답시간은 오퍼레이션이 발생한 후 수행까지의 지연 시간을 의미한다. 주요 실험 파라미터인 초당 생성된 저장 오퍼레이션의 수는 20개에서부터 20개 단위로 160개까지 8단계로 변화시켜 보았으며, 이는 저장 시스템에 가해지는 작업 부하를 의미한다.

3.2 실험 결과 및 분석

이 실험은 기본적으로 SSD 스토리지를 탑재한 일반적인 사용자 시스템에서 부하된 저장 워크로드, 즉 초당 발생된 저장 연산의 수가 시스템 성능에 어떤 영향을 미치는지를 분석하기 위함이다. Fig.4는 초당 발생된 저장 연산의 수의 증가에 따른 저장 연산의 처리치를 표시한 그래프이다. 발생된 초당 저장 연산의 수가 늘어날수록 점차로 연산의 처리 결과치가 증가함을 알 수 있다. 또한, 전반적인 저장 연산의 처리 성능을 측정된 결과, 제안기법인 Mr-Switch가 SSD-1과 SSD-Mr보다 높게 나타났다. Fig.4에서 보면, 초당 저장 연산의 수가 대략 100~120개를 넘으면서

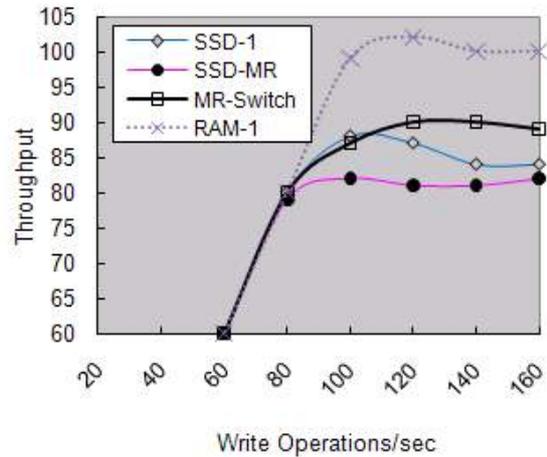


Fig. 4 Operation Throughput

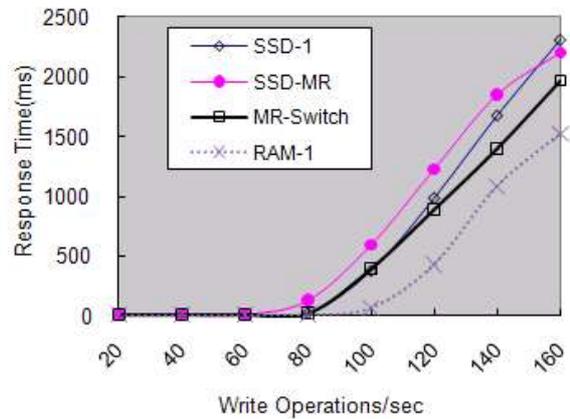


Fig. 5 Average Response Time

각 저장 기법의 성능이 점점 낮아진다. 이는 초당 저장 연산수의 증가에 의한 플래시 메모리 기반 SSD 스토리지에서의 I/O 연산 집중화가 성능에 영향을 크게 미치는 주요 요소임을 의미하며, 이 수치 이상으로 워크로드를 활성화시키는 것이 성능 향상에 도움이 되지 않음을 의미한다. 특히 SSD를 구성하는 플래시 메모리의 쓰기 연산의 느린 속도와 수반 되는 소거 연산이 주요 요인이다. 그래프에서 보면 저장 연산의 부하가 적은 80 이하의 시작 구간에서는 SSD의 느린 쓰기와 소거 연산이 시스템 안에서 수용되어 성능 저하를 일으키지 않는다. 하지만, 시작 구간을 지나면서 생성되는 저장 연산의 수가 증가하므로, 플래시 메모리에 하위 레벨의 쓰기 연산의 수도 증가하게 된다. 따라서 중간 구간 이후에서는 저장 연산들이 SSD 컨트롤러의 작업 큐에 과도하게 쌓이면서, 저장 연산 처리가 지체되며 결과적으로 성능이 서서히 저하되게 된다.

하지만, 동일한 워크로드의 조건에서도 제안한 Mr-Switch 저장 기법이 SSD-1나 SSD-Mr 기법에 비하여 성능이 상대적으로 더 높다. 100이상의 부하 구간에서 평균하면, Mr-Switch가 기존 기법보다 9.2% 정도 처리 성능이 높다. 특히, 최대 저장 연산 처리치는 Mr-Switch가 90개로 SSD-Mr의 81개 보다 11%정도 더 높다.

그 이유는 위의 저장 성능 저하의 주요 원인인 SSD의 쓰기 연산의 부담에 의한 일시적인 홀딩 현상을 미리 스위칭기법은 먼저, RAM에 저장함과 동시에, 2개의 SSD에 RAID-0(striping) 방식으로 분산저장하기 때문이다. SSD에 미러링하는 오버헤드는 과부하시 더 성능을 저하시키는데, 본 기법은 SSD와 초고속의 RAM에 동시 저장함으로써, 미러링의 안정성을 유지하며, SSD에 스트라이핑함으로써 저장 속도를 향상 시킨다. 일단 두 매체에 저장후 결과를 리턴하고, 추후 아이들 타임에 스트라이핑된 데이터를 서로 미러링한 후 RAM의 데이터는 삭제되게 된다.

따라서 본 기법은 안정성을 위한 미러링을 유지하면서도, 스트라이핑으로 속도를 높이며, 쉬는 시간에 지연 저장을 수행함으로써 전체적인 성능 향상과 대기시간을 크게 감소시킬 수 있었다. 또한, SSD의 플래시 메모리 영역에 대하여, 쓰기 연산이 1/2정도 뒤로 지연되어 결과적으로 모아 쓰기를 유도하는 긍정적인 효과도 발생하므로, 플래시 메모리의 수명이 더 연장되며, 소거 연산을 위한 전력소모도 좀 더 줄이는 부수적인 효과도 기대할 수 있다.

또한, Fig.5에서 요청한 저장 연산에 대한 각 스토리지 관리 기법의 응답시간을 비교해보면, RAM-1을 제외하면, Mr-Switch, SSD-1, SSD-Mr 순으로 우수하게 나타났다.

먼저, 단일 SSD를 사용하는 SSD-1 기법에 비하여, SSD를 미러링한 SSD-Mr가 응답시간이 9% 더 늦게 나타났는데, 이는 SSD-Mr는 두 개의 SSD에 모두 즉, 2번의 저장이 완료되어야 하기 때문이다. Mr-Switch는 SSD-1에 비하여, 13% 정도 응답성능이 빠른데, 이는 전술한 대로, Mr-Switch는 RAM 저장이 신속하며, SSD도 순간 스트라이핑으로 저장 연산을 더 신속히 완료시키기 때문이다. 이후 대기연산이 없으면, 스트라이핑된 데이터를 신속히 미러링된 데이터로 상호복제하여 저장연산이 최종적으로 완료되게 된다.

4. 결론 및 향후 연구

기존 하드디스크의 새로운 대체품으로 SSD가 급부상하고 있다. 본 연구에서는 기존의 하드 디스크 기반 저장 시스템에서 성능과 안전성을 높이기 위하여 사용하고 있는 핵심 RAID 기술을 기반으로, SSD의 매체 특성에 맞게 개선된 미러-스위칭이라는 새로운 저장 관리 기법을 제안하였다. 제안한 기법은 SSD에서 발생하는 느린 쓰기 속도와 프리징 현상을 회피하기 위하여, 저장 매체간의 심각한 저장 속도의 차이를 고려하여, 일시적으로 빠른 저장 매체를 우선 저장하고, 느린 저장 매체는 지연 저장을 수행하게 하여 전체적인 성능을 개선하게 된다.

테스트 결과를 통하여 제안 기법이 쓰기 부하와 프리징을 감소시켜서 기존의 SSD-미러링 기반의 저장 관리 기법보다 연산 처리 측면에서 9%, 응답 시간측면에서 13% 더 우수함을 확인하였다.

후기

이 논문은 2010년도 산학협동재단의 학술연구비지원으로 수행된 연구임.

참고문헌

- [1] Samsung, 2010, "Samsung SSD is NOW", <http://www.samsung.com/global/business/semiconductor/products/flash/ssd/home/home.html>
- [2] Solid Data Systems, 2007, "Comparison of Drives Technologies for High-Transaction Databases", White paper
- [3] Samsung, 2010, "What is NAND Flash based SSD?", http://www.samsung.com/global/business/semiconductorproducts/flash/Products_FlashSSD.htm
- [4] YoungHyun Bae, 2007, "Design of High Performance SSD, The KIPS transactions, Vol 25, No. 6, pp. 18-28
- [5] Siwoo Byun, 2010, "A Design of RAID-1 Storage using Hard Disk Drive and Flash Memory Drive " Transactions of the Society of Information Storage Systems, pp. 27-33
- [6] SeongHun Bak, Jongwon Park, 2002, "An Implementation and Performance Evaluation of a RAID System Based on Embedded Linux", The KIPS transactions. Part A, V.9A, No.4, pp.451-458.
- [7] Mesquite, "CSIM2.0", http://www.mesquite.com/documentation/documents/CSIM20_User_Guide-C.pdf, 2008