

가상머신에서 TRIM Command를 지원하는 SSD를 사용하는 방법

김해성^o, 원유집
한양대학교

matia513@hanynag.ac.kr, yjwon@hanyang.ac.kr

Approach for using SSD with TRIM Command on Virtual Machine

Haesung Kim, Youjip Won
Department of Electronics and Computer Engineering
Hanyang University

요 약

덮어쓰기를 할 수 없는 NAND Flash Memory의 단점을 보완하고자 고안된 TRIM Command를 가상 머신의 스토리지에서 지원하는 방법을 제안한다. 이를 통해 우리는 일체의 하드웨어 장비 없이 TRIM Command를 지원하는 SSD를 사용할 수 있다. 또한 시스템의 모든 구성은 QEMU 가상머신 기반에서 소프트웨어로 구현되기 때문에 성능 분석에 용이하게 사용할 수 있다.

1. Introduction

NAND Flash Memory기반의 SSD는 빠른 IO와 저전력 등과 같은 장점으로 인해 차세대 저장 장치로 급부상하고 있다. 하지만 NAND Flash Memory는 덮어쓰기를 할 수 없다는 제한사항을 가진다. 이러한 단점을 보완하기 위해 Technical Committee T13에서는 TRIM Command를 제안하였다. 이는 스토리지의 사용하지 않는 영역에 대한 정보를 운영체제가 호스트인터페이스에 알려줌으로써 저장 공간을 효율적으로 사용할 수 있는 기회를 제공한다. 하지만 이러한 TRIM Command는 IO 명령이 아니기 때문에 Trace 틀을 사용하더라도 정보를 수집하기가 어렵다. 뿐만 아니라 운영체제와 디스크에서 모두 TRIM Command를 지원해야만 이를 활용할 수 있다.

본 논문에서는 QEMU 기반의 가상머신에서 TRIM Command를 지원하는 SSD를 사용하는 방법을 제안한다. 우리는 이를 통해 일체의 하드웨어 없이 SSD를 사용하는 시스템 환경을 구축할 수 있다. 이러한 방법은 하드웨어를 포함한 모든 시스템이 소프트웨어로 구현되기 때문에 사용자가 쉽게 접근할 수 있을 뿐만 아니라 시스템을 구성하는데 드는 비용을 절약할 수 있다[1].

본 논문의 2장에서는 Background에 대해서 설명하고, 3장에서는 가상머신에서 TRIM Command 지원을 위해 필요한 호스트 인터페이스 설정에 대한 소개를 한다. 그리고 4장에서 실험결과를 보이고, 마지막 5장에서 결론을 내린다.

2. Background

2.1 QEMU

QEMU[2]는 2005년 Fabrice Bellard가 발표한 오픈소

스 머신 에뮬레이터이다. CPU Emulator, Emulated Device(display, mouse, keyboard, hard disk), 그리고 Emulated device와 해당 호스트 device를 연결하는 Generic device(block, character, network devices)와 그 외 Machine description, Debergger, User Interface 와 같은 서브시스템으로 구성된다. QEMU는 동적변환기를 사용하여 특정 머신을 위해 만들어진 운영체제나 프로그램을 다른 머신에서 수행할 수 있도록 한다. 예를 들어 ARM에서 실행 되도록 작성된 프로그램을 x86 머신에서 실행시킬 수 있다. QEMU에서 사용하는 동적 변환기는 타겟 CPU의 명령어를 micro operation이라고 부르는 작은 C코드의 조각으로 바꾸고 실행 중에 여러 개의 micro operation을 연결하여 호스트 머신의 기능을 수행할 수 있도록 한다. 이러한 기법을 통해 QEMU는 매우 뛰어난 성능의 머신 에뮬레이터를 실현하지만 실제 하드웨어의 성능에는 많이 못 미치는 상황이다.

2.2 TRIM Command

T13에서는 덮어쓰기가 불가능한 SSD의 단점을 보완할 수 있는 TRIM이라는 Data Set Management Command[3]를 제안하였다. HDD기반 컴퓨터 시스템의 운영체제에서는 파일을 삭제하면 해당파일과의 연결을 끊어 파일이 삭제된 것처럼 보이게 하지만 실제 파일은 삭제되지 않고 그대로 남아 있게 된다. 덮어쓰기가 가능한 HDD는 상관이 없지만 SSD는 이것을 처리하기 위해서도 많은 오버헤드가 발생한다. TRIM command는 이렇게 삭제된 파일들이 차지하는 저장매체의 불필요한 섹터 영역들에 대한 정보를 알려 SSD에게 알려줌으로써 SSD가 IO 요청을 처리할 때 이를 활용하여 IO 효율을 얻을 수 있는 기회를 제공한다. 즉, TRIM 자체가 SSD의 성능을 좋아지게 하는 것은 아니지만 TRIM 정보를 어떻게 사용하느냐에 따라 SSD의 성능에 대한 기여도는

달라질 수 있다.[4]

3. TRIM Command 지원을 위한 호스트 인터페이스 설정

본 논문에서 TRIM Command를 지원하는 가상의 SSD를 사용하기 위해 QEMU가 에뮬레이션하는 IDE 장치를 활용한다. 하지만 기존의 QEMU에서 사용하는 IDE는 HDD만을 사용할 수 있도록 구현되어 있다. 따라서 QEMU 기반의 가상 머신에서 SSD를 사용하기 위해서는 QEMU가 에뮬레이션하는 IDE의 내부 구현을 변경해서 사용해야 한다.

운영체제는 디스크를 인식하는 과정에서 디스크의 동작과 관련된 정보를 얻기 위해 IDE로 Identify 요청을 보낸다. 요청을 받은 IDE의 호스트 인터페이스는 여기에 응답을 하게 된다. 본 연구에서는 IDE장치를 SSD로 인식하도록 하기 위해 ATA-8의 표준에 따라 Identify 응답을 변경하였다. 운영체제는 디스크 장치를 확인할 때 디스크의 RPM을 확인한다. 이때 반환된 값이 "0001h"일 경우 운영체제는 이를 Non-rotating media, 즉 SSD로 인식하게 되며 "0401h - FFFeh" 일 경우 일반적인 RPM 수치를 나타낸다. 예를 들어 5400RPM의 하드디스크의 경우 "1518h" 값을 반환한다.

RPM값을 수정하는 것 이외에 SSD가 TRIM Command를 지원하도록 하기 위해 이와 관련된 호스트인터페이스 설정을 수정한다. 이 설정 값들을 표 1에 나타내었다.

	Word	Bit	Value
Device Configuration Overlay (DCO) Data	21	10	1
Identify Device Data	69	14	1
	169	0	1
	206	0	1
	217	0	1

표 1. ATA/ATAPI Command Set-2(ACS-2)

4. Experiment

TRIM Command는 시작 섹터 주소와 16bit(최대 65536)의 길이로써 Trimmed 영역에 대한 정보를 알린다. Windows7에서 TRIM Command를 추적해본 결과 NTFS는 운영체제가 부팅될 때 현재 사용하지 않고 있는 모든 디스크의 영역에 대해 TRIM Command를 통해 디스크에 알린다. 그리고 부팅된 이후에도 운영체제에서 아무작업도 하고 있지 않을 때 불규칙적으로 TRIM Command를 전송하는 것을 확인하였다.

그리고 가상머신상에서 TRIM Command를 지원하는 SSD를 사용하여 IO Trace를 수집해 보았다. 그림 1은 Photoshop CS4를 설치하면서 발생한 Write 요청에 대한 분포도이다.

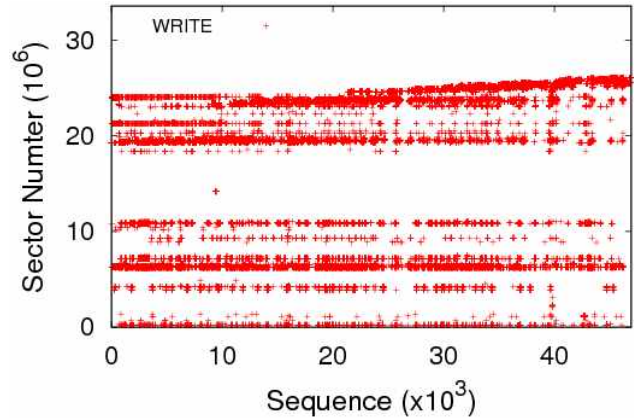


그림 1. IO distribution with TRIM Command

5. Conclusion

TRIM 정보는 IO 요청이 아니며 NTFS의 소스는 공개되지 않기 때문에 IO trace 도구를 사용하더라도 소프트웨어 레벨에서는 이를 추적하기가 어렵다. 따라서 우리는 QEMU 기반의 가상머신을 활용하여 TRIM Command를 지원하는 SSD를 사용할 수 있는 방법을 제안하였다. 이를 통해 우리는 일체의 하드웨어 장비 없이도 TRIM Command 정보를 수집할 수 있다. 그리고 실험을 통해 TRIM을 사용하였을 경우 IO의 패턴이 바뀌는 것을 확인하였으며 이는 SSD의 동작에도 영향을 미친다. 따라서 우리는 본 논문에서 제안한 방법은 TRIM Command의 사용으로 인한 SSD의 성능 변화를 분석하는데 활용할 수 있다.

6. Acknowledgement

본 연구는 지식경제부 및 한국산업기술평가 관리원의 산업원천기술개발사업(정보통신)의 일환으로 수행하였습니다. (No. 10035202, 대용량 MLC SSD 핵심기술 개발)

Reference

- [1] 김해성, 김주현, 이성진, 원유집. 가상 SSD 시뮬레이터. 한국정보과학회 제37회 추계학술발표회. 2010년 11월
- [2] Fabrice Bellard. Qemu, a fast and portable dynamic translator. USENIX 2005 Annual Technical Conference, April 2005
- [3] F. Shu and N. Obr. Data set management commands proposal for ata8-acs2. In Technical Committee T13, 2007.
- [4] Joohyun Kim, Haesung Kim, Seongjin Lee, and Youjip Won. Ftl design for trim command. In Proceedings of 15th International Workshop on Software Support for Portable Storage(IWSSPS2010), October 2010.