제19회 한국정보처리학회 춘계학술발표대회 논문집 제10권 제1호 (2003.5)

비 휘발성 캐시를 이용한 RAID 5 제어기의 개념 설계

허정호*, 장태무**
동국대학교 컴퓨터 공학과
e-mail : hjh99*, jtm{dgu.ac.kr}

Conceptual Design of a RAID 5 Controller with Non-volatile Cache

Jung-Ho Huh*, Tae-Mu Chang**
Dept. of Computer Science, Dong-Guk University

요 약

RAID 시스템에서 비 휘발성 쓰기 캐시를 이용한 디스크 제어기는 시스템 성능 향상의 중요한 요소 중 하나이다. 2단계 캐시는 1단계 캐시에 비해 우수한 성능을 보이고 시간적, 공간적 지역성에도 효율적이며 호스트 측에 비 휘발성 기억소자로 구성된 L1 캐시를 두어 디스크 캐시의 신뢰도를 높일 수 있다. 호스트에서 읽기/쓰기 적중된 데이터가 L1 캐시에서 수행되는 동안 L2 캐시에서는 디스크로 destage하는 동작을 비동기적으로 병렬 처리하고 데이터와 패리티를 함께 캐시에 적재하여 RAID 5의 "소규모 쓰기 문제"를 완화시키고자 한다. 제안된 캐시 시스템은 2단계로 구성되어 대용량 디스크 캐시에서 디스크 입출력 시간을 향상시키고 효율적으로 일관성을 유지할 수 있는 디스크 제어기 상에 위치하는 RAID 5 디스크 캐시 모델을 제시하여 수행속도를 개선시키고자 한다.

1.서론

RAID(Redundant Arrays of Inexpensive Disks)는 일반적으로 자기 디스크 드라이브의 그룹이 제어기에 의해 논리적으로 단일 입출력 장치와 같이 사용되도록 구성되어 신뢰성, 가용성, 성능 등의 면에서 대표적인 대용량 저장 수단으로 많은 응용 분야에서 활용되고 있으며 시장 규모도 급신장 하고 있다. 프로세서의 처리속도는 매년 40-100%씩 증가되는데 비해 기계적인 부품을 가진 자기 디스크의 성능 향상은 7%정도에 불과하므로 중앙처리장치 및 주 기억장치와 디스크의 입출력 속도 차이를 극복하기 위한 방법으로 RAID는 널리 사용된다[1,2]. 또한 디스크 캐시는 디스크 접근을 줄이는 방안으로 널리 쓰이며 DRAM(Dynamic Random Access Memory)의 가격은 매 10년 만에 100 분의 1로 떨어질 정도로 저렴해지는 추세이므로[3] 디스크 시스템에서 대용량의 캐시를 사용하는 경향이 높다. 본 논문에서는 대용량화된 디스크 캐시에서 적절한 구현 방안이 되고 성능도 개선할 수 있는 2단계 디스크 캐시를 이용한 RAID 5 제어기의 모델을 제시하고자 한다. 2단계 디스크 캐시는 각 단계별 캐시 간에 데이터 일관성 유지를 위한 쓰기정책과 포함관계 특성에 따라 동작의 차이를 갖게 되는데 본 논문에서는 두 캐시의 내용이 어느 정도 포함관계를 갖는 모델(NVM 모델 1)과 포함관계를 갖지 않는 모델(NVM 모델 2)로 구분하였다.

본 논문에서는 2단계 디스크 캐시의 예로 리디움 밧데리(Lithium Battery)를 이용한 비 휘발성(Non-Volatile) 기억소자를 1단계 캐시로 사용하고 휘발성(Volatile) 기억소자를 2단계 캐시로 사용하는 모델의 예를 들고 운영방법을 제시하여 비 휘발성 기억소자를 사용하는 통상적인 디스크 캐시 제어기에 비해 캐시 적중률을 높여 수행속도를 개선시키는 결과를 얻고자 한다.

2. RAID에서 2단계 캐시의 필요성

RAID에서의 캐시는 응용 프로그램에서 메모리 접근이 임의로 이루어지지 않고 지역성의 원리가[4] 크다. 응용환경과 구성 모델의 분석에 따라 RAID의 캐시에 전형적인 메모리 캐시를 구별하여 적용해야 하는 점은 다음과 같다.

첫째, 메모리 캐시에 비해 RAID의 캐시는 구성의 제약이 비교적 덜하여 용량 증가가 비교적 용이하다.

둘째, RAID 시스템의 공간적 지역성을 높일 수 있다. 세그먼트(segment)나 페이지(page)에 기초한 알고리즘을 사용하는 전형적인 메모리는 논리적으로 연속적인 파일이 메모리에서는 물리적으로 연속되지 않는다. 메모리 캐시는 주로 시간적 지역성에만 집중되지만 RAID의 캐시는 디스크 배열로부터 호스트(host)로 데이터를 저장할 수 있는 실제적인 버퍼이며 운영체제에서 파일은 트랙이나 버퍼 단위로 저장된다. 또한 대부분의 응용프로그램이 벤치마킹 결과 50% 이상의 인접 데이터를 사용하므로 시간적 지역성뿐만 아니라 L2 캐시를 이용하여 공간적 지역성 또한 높일 수 있다[4].

셋째, RAID의 캐시는 RAID 제어기에 내장된 프로세서에 의해 운영되므로 알고리즘이 호스트의 프로세서에 영향을 주지 않는다[5].

위와 같은 이유로, 메모리 캐시에 RAID 디스크 캐시를 동일하게 적용시키기가 어려우므로 RAID 디스크에 해당되는 캐시 설계가 필요하다. 즉, L1 캐시는 시간적 지역성을, L2 캐시는 공간적 지역성을 높여 주므로 RAID에서 2단계로 캐시 설계를 하여 이를 읽기, 쓰기의 버퍼처럼 사용하여 저속으로 실행되는 대용량 디스크 배열의 접근시간의 감소를 가져와 수행 속도를 개선한다.

3. RAID 5에서의 2단계 디스크캐시 모델

본 논문에서는 NVRAM(Non-Volatile RAM)을 디스크 캐시에 이용하여 L1 캐시에서 읽기, 쓰기에 이용하고 L2 캐시는 휘발성 기억소자로 구성된 2가지 디스크 캐시 모델을 제안한다. 각 단계 캐시의 용량은 NVRAM을 사용하는 통상적인 시스템에서 NVRAM 및 휘발성 기억소자의 캐시의 용량과 유사하게 구성하며 L1 캐시의 읽기, 쓰기와 L2 캐시와 디스크 사이의 동작은 병렬성이 있다. L1 캐시에 선택적으로 캐싱된 작은 블록은 생명 주기가 증가되어 시간적 지역성이 높아져 캐시 부재가 발생할 때마다 L2 캐시에 있는 다수의 이웃한 소규모 블록들을 선인출(prefetching)하여 공간적 지역성을 높인다. 본 논문에서 제안된 2가지 모델의 주요 특성은 다음과 같다.

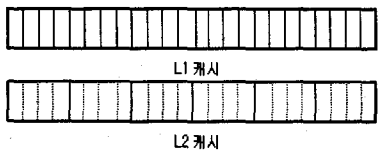
첫째, L1 캐시는 셋트 연관 사상 알고리즘을사용하고 L2 캐시는 완전 연관 사상 알고리즘을 사용한다.

둘째, 교체 알고리즘은 L1 캐시는 시간적 지역성을 고려한 LRU 정책을 사용하고 L2 캐시는 디스크와 내용이 일치하는 것에 한하여 LFU 정책을 사용한다.

셋째, L1 캐시는 작은 단위인 섹터나 그보다 더 작은 단위로 하고 L2 캐시는 비교적 큰 단위인 트랙 단위로 한다.

넷째, 쓰기 정책은 write back 방법을 사용하여 L2 캐시를 디스크 동작에 전담시킨다.

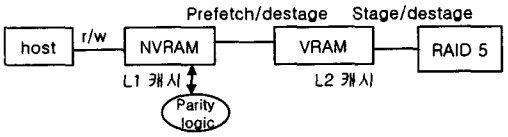
2단계 캐시의 개념적인 형태는 (그림 1)과 같으며 L2 캐시는 작은 블록들이 특정한 큰 블록에 속해 있는 형태로 그 블록이 교체될 때까지 존재한다[5]. 또한 작은 블록이 포함된 큰 블록이 교체될 때 뿐 아니라 전에 접근되었던 여러 개의 작은 블록들이 L1 캐시로 선택적으로 이동한다. L1 캐시를 작은 블록 사이즈로 하면 시간적 지역성을 가진 데이터가 주어진 캐시 공간에서 L1 캐시의 블록 개수를 더 증가시킨다.



(그림 1) 2단계 캐시의 구조

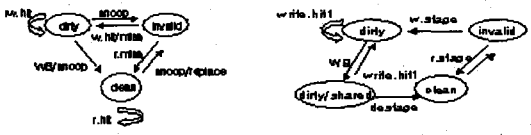
3.1 NVM 모델 1

본 논문에서 제시하는 RAID 5에서의 2단계 디스크 캐시 NVM(Non-Volatile Memory) 모델 1의 구조는 (그림 2)와 같다.



(그림 2) NVM 모델 1

이 모델은 읽기, 쓰기는 L1 캐시에서 하며 L1 캐시의 내용을 L2 캐시가 일부 포함한다. L1 캐시에서 쓰기가 발생할 때마다 L2 캐시로 자료 전송은 하지 않고 수정 발생 사실만 알린다. 이는 L2 캐시에서 발생되는 디스크와의 stage/destage동작의 독립성을 높이고자 하는 방법이다. 두 캐시는 모두 write back으로 운영되고 L2 캐시를 디스크 동작에 전담시키고 2개 이상의 프로세서가 디스크를 공유하여 읽기/쓰기 동작을 모두 하는 경우에 사용할 수 있다. (그림 3)은 NVM 모델 1의 각 단계별 캐시의 상태도이며 두 캐시가 공통적으로 dirty상태일 때는 L1 캐시는 L2 캐시와 내용이 다르고 L2 캐시는 디스크와 내용이 다르다는 것을 나타내며 동작과정은 다음과 같다.



(가)L1 캐시 (나)L2 캐시

(그림 3)NVM 모델 1의 L1, L2 캐시의 상태도

(1)읽기 적중:L1 캐시에서 읽는다

(2)읽기 부재:L2 캐시를 점검하여 적중하였을 때 clean상태면 디스크와 일관성이 있는 상태이고 dirty 상태이면 L1 캐시에서 쓰기 동작을 하였다는 것이므로 (그림 3)의 (나)에서 라인교체 등의 이유로 L1 캐시에서 L2

캐시로 write back이 발생하면 L2 캐시에서는 일관성 유지를 위해 dirty/shared상태로 바뀐다. L1 캐시에서는 write back이나 snoop동작에 의해 clean상태가 되고 L2는 destage에 의하여 clean상태로 된다. L2캐시가 clean이나 dirty/shared상태면 L2 캐시에서 L1 캐시로 전송하면 되고 dirty상태면 L1 캐시의 수정된 데이터를 L2 캐시로 write back을 먼저 해주어야 한다. 이때 모든 L1 캐시를 조회하여 L1 캐시와 마찬가지로 L2 캐시에도 해당 라인이 존재하지 않으면 stage동작이 일어난다.

(3)쓰기 적중:L1 캐시에서 수정이 이루어져 L2 캐시에서 수정된 사실이 통보되면 L2 캐시는 다른 L1 캐시에게 무효화 신호를 보내고 L2 캐시는 dirty 상태로 바뀐다.

(4)쓰기 부재:L1 캐시에서 수정이 이루어지며 L2 캐시에 이 라인이 존재하지 않으면 stage동작을 하고 존재하면 dirty상태로 바뀌면서 다른 L1 캐시에게 무효화 신호를 보낸다. NVM 모델 1의 동작을 (그림 4)와 같이 정리할 수 있

다.

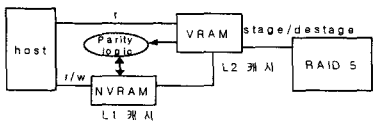
```
READ(L1):
 if (HIT) {Transfer from C1 Cache;
 Update LRU status;}
 else {Send Request to L2 Cache;}
READ(L2):
 if(HIT) Transfer to L1 Cache(Sequential Prefetching);
 else Read from Disk and Restart READ;
Write(L1):
if(HIT) Update L1 Cache marking DESTAGING;
 else allocate a Line frame marking EXCLUSIVE;
DESTAGE(envoked asynchronously):
 Transfer from L1 Cache to L2 Cache marking DIRTY;
 Transfer DIRTY lines to Disk;
STAGE (envoked asynchronously):
 Transfer EXCLUSIVE lines from Disk;
 Envoke a DESTAGE operation;
```

(그림 4)NVM 모델 1의 개략적인 동작

패리티 회로(parity logic)는 L1 캐시에 쓰기 시에 패리티 갱신이 이루어지는 회로이며[6] 일반적인 RAID 5 쓰기 방법인 읽기-수정-쓰기(Read-Modify-Write)방식으로 독자적인 패리티 엔진에서 동작한다.

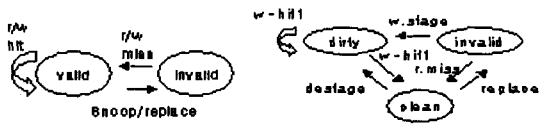
3.2 NVM 모델 2

RAID 5에서의 2단계 디스크 캐시 NVM 모델 2는 (그림 5)와 같이 L1 캐시는 읽기, 쓰기,L2 캐시는 읽기에 이용된다.



(그림 6) NVM 모델 2

이 모델은 L2 캐시가 L1의 내용을 포함하지 않으며 전체 캐시의 용량이 L1 캐시들의 합만큼 커질 수 있다. 이는 호스트 측에서 L1, L2 캐시에 모두 접근할 수 있고 읽기 부재 시에 동일한 내용이 두 캐시에 중복되어 존재하지 않는다. stage/destage 동작을 L1 캐시의 동작과 분리하기 위하여 L1 캐시와 디스크는 연결하지 않고 호스트 측에 위치한 L1 캐시가 디스크 측에 위치한 L2 캐시에 비해 비교적 빠르므로 L1 캐시에서 읽기 적중하는 것이 좋다. 이 모델은 write through 수정 정책을 사용하면 (그림 6)의 상태도와 같고 write back 수정 정책을 사용하면 NVM 모델 1의 상태도와 같다.



(가)L1 캐시 (나)L2 캐시

(그림 6)NVM 모델 2의 L1, L2 캐시의 상태도

L1 캐시는 valid와 invalid상태로 운영되어 valid상태에서는 L2의 라인 내용과 일치하고 snoop는 다른 L1캐시에서의 쓰기로 인한 무효화 동작이다. L2 캐시는 dirty, invalid, clean의 상태로 운영되어 dirty는 수정된 상태, clean은 디스크와 내용이 일치하는 상태, stage동작은 쓰기 부재로 인한 w.stage와 읽기 부재로 인한 r.stage로 구별된다. replace는 clean상태와 L1 캐시에서 포함하고 있지 않을 때 일어나며 w.hit1은 L1 캐시에서 쓰기 적중이 일어났을 때이며 동작 원리는 다음과 같다.

(1)읽기 적중: L1 캐시에서 읽는다
(2)읽기 부재: L2 캐시에 접근하여 적중 시 읽고 부재 시에는 디스크에서 stage한다. 라인 교체 발생시 수정되지 않는 라인을 우선한다.
(3)쓰기 적중: L1 캐시에 쓴다. 쓰기 동작에 공간적 지역성이 높은 경우 여러 라인을 모아서 L2 캐시에 쓰고 L2 캐시에서 디스크에 destage할 때도 마찬가지이다. L2 캐시에도 부재 시 디스크에서 stage한다.
(4)쓰기 부재: L1 캐시에 쓴다. L2 캐시에 부재 시 디스크에서 stage하고 L1, L2 캐시 모두 라인이 교체될 수 있으며 이후의 동작은 쓰기 적중 시와 같고 (그림 7)과 같이 동작을 정리할 수 있다.

```
READ:
 if (HIT on L1 Cache)Transfer from L1 Cache;
 else if(HIT on L2 Cache)Transfer L2 Cache;
     else {READ from Disk;
           Restart L2 READ;}
WRITE:
 if(HIT) Update L1 Cache marking DESTAGING;
 else allocate a Line frame marking EXECULSIVE;
DESTAGE(envoked asynchoronously):
 Transfer from L1 Cache to L2 Cache, marking DIRTY;
 Transfer DIRTY lines to Disk;
STAGE(envoked asynchronously):
 Transfer EXCLUSIVE lines from Disk;
Envoke a DESTAGE operation;
```

(그림 7)NVM 모델 2의 개략적인 동작

패리티 회로(parity logic)는 NVM 모델 1에서와 같이 읽기

-수정-쓰기(Read-Modify-Write)방식으로 독자적인 패리티 엔진에서 이루어진다.

4. RAID 5에서의 2단계 디스크캐시 모델의 장점

본 논문에서 제안한 RAID 5에서의 2단계 디스크 캐시 모델인 NVM 모델 1과 NVM 모델 2의 장점을 요약하면 다음과 같다.

첫째, 디스크 캐시 전체를 고가의 반도체 기억장치 부품으로 구성하지 않더라도 성능이 높고 대용량인 저가의 디스크 캐시를 제공할 수 있다.

둘째, stage/destage 등의 디스크 동작과 각 단계의 읽기 적중, 쓰기 등의 캐시 동작들이 병렬처리가 가능하다.

셋째, 각 단계의 캐시들이 디스크 캐시 전체의 용량으로 사용 가능하다.

넷째, 각 단계별 캐시 라인의 크기와 교체 알고리즘, 사상 방식을 다르게 적용하여 캐시 적중률을 높인다.

다섯째, RAID 5 디스크를 이용하여 패리티 정보를 분산 저장하여 디스크 쓰기 동작을 병렬로 수행하고 신뢰도를 높이며, 캐시에 패리티를 함께 적재하여 '작은 쓰기 문제'(Small Write Problem)를 완화한다.

5.결 론

본 논문에서는 시간적, 공간적 지역성의 원리를 이용하여 2단계 캐시로 캐시 적중률을 높이고 L1 캐시에 비 휘발성 기억소자를 이용하여 전원이 차단되어 시스템이 오류가 나더라도 디스크 캐시의 신뢰도를 높인다. 또한 쓰기 캐시에 데이터와 패리티를 함께 적재하여 쓰기 시에 패리티의 추가적인 디스크 접근 없이 RAID 5의 '작은 쓰기 문제'를 완화시키고자 하였다.

빠른 접근 시간과 다양한 응용 프로그램에 적합한 시간적, 공간적 지역성을 이용한 캐시 적중률을 높이기 위해서는 단일 캐시 구조만으로는 상반된 특성을 가진 두 가지 지역성을 효율적으로 운영하는데 한계가 있으므로 각각의 지역성을 효과적으로 반영할 수 있는 2단계 디스크 캐시를 이용한 RAID 5 제어기를 구현하기 위한 모델을 제시하여 통상적인 2단계 디스크와 비교하였을 때 성능향상을 기대할 수 있다.

앞으로의 작업으로는 본 논문에서 제시한 각 모델들의 L1, L2 캐시 간에 일관성을 유지하기 위한 오버헤드 문제와 NVM 모델 2가 캐시의 용량이 커짐에 따라 성능 또한 나아진다고 단정할 수는 없으므로[7] 이들을 수학적인 이론으로 구체화시켜 분석적 모형을 만들고 여러 가지 다양한 작업 부하를 사용한 시뮬레이션을 통해 비교 연구해 볼 수 있다.

참고 문헌

[1]S. Chen and D. Towsley, "A Performance Evaluation of RAID Architectures," IEEE Transactions on Computers, Vol.4, No.10, pp.1116-1129, 1996.

[2] A. K. Sahai, " Performance Aspects of RAID Architectures," IEEE International Conference on Performance Computing and Communications,pp.321-327, 1997.

[3] J. Gray and R. Shenoy, "Rules of Thumb in Data Engineering," Technical Report MS-TR-99-100, MicroSoft Research, 2000.

[4] Jung-Hoon Lee, Jang-Soo Lee and Shin-Duk Kim, "A new cacne architecture based on temporal and spaial locality," Journal of Systems Architecture, pp. 1451-1467, 2000.

[5] Chen Yun, Yang Genke, Wu Zhiming, "The Application of Two-Level Cache in RAID System," Proceedings of the 4th World Congress on Intelligent Control and Automation, June. 2002.

[6] Anujan Varma and Quinn Jacobson, "Destage Algorithms for Disk Arrays with Nonvolatile Caches," IEEE Transactions on Computers, Feb. 1998.

[7] R. Karedla, J. S. Love, B. G. Wherry, "Caching Strategies to Imporve Disk System Performance," IEEE Computer, pp. 38-46, Mar. 1994.