(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2015년11월26일
(11) 등록번호 10-1572401
(24) 등록일자 2015년11월20일

(51) 국제특허분류(Int. Cl.)
G06F 12/08 (2006.01)
(21) 출원번호 10-2014-7013814
(22) 출원일자(국제) 2012년12월10일
심사청구일자 2014년07월23일
(85) 번역문제출일자 2014년05월22일
(65) 공개번호 10-2014-0106516
(43) 공개일자 2014년09월03일
(86) 국제출원번호 PCT/IB2012/057140
(87) 국제공개번호 WO 2013/108097
국제공개일자 2013년07월25일
(30) 우선권주장
13/352,230 2012년01월17일 미국(US)
(56) 선행기술조사문헌
US20020199070 A1
US20030070042 A1
US20040068612 A1
US20090210620 A1

(73) 특허권자
인터내셔널 비지네스 머신즈 코포레이션
미국 10504 뉴욕주 아몬크 뉴오차드 로드
(72) 발명자
굽타, 로케쉬, 모한
미국 애리조나 85744-0002, 투싼, 사우스 리타 로드 9000, 엠디:9032-1 103, 아이비엠 코포레이션
칼로스, 매튜, 조셉
미국 애리조나 85744-0002, 투싼, 사우스 리타 로드 9000, 엠디:9032-1 102, 아이비엠 코포레이션
(뒷면에 계속)
(74) 대리인
허정훈

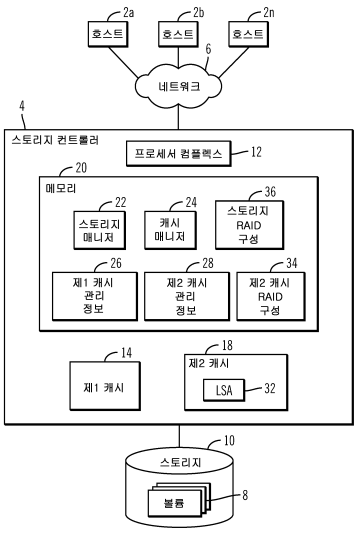
전체 청구항 수 : 총 9 항 심사관 : 김대성

(54) 발명의 명칭 **제1 캐시로부터의 트랙들의 제1 스트라이드를 제2 캐시 내 제2 스트라이드에 라이트하기 위해 파퓰레이트하는 방법**

(57) 요 약

제1 캐시, 제2 캐시, 및 스토리지 시스템을 포함하는 캐시 시스템에서 데이터를 관리하기 위한 컴퓨터 프로그램 제품, 시스템, 및 방법이 제공된다. 제1 캐시로부터 강등(demote)시킬, 스토리지 시스템에 저장된 트랙들에 관한 결정이 이뤄진다. 강등시키기로 결정된 트랙들을 포함하는 제1 스트라이드가 형성된다. 제1 스트라이드 내 트랙들을 포함시키기 위한 제2 캐시 내 제2 스트라이드에 관한 결정이 이뤄진다. 제1 스트라이드로부터의 트랙들은 제2 캐시 내 제2 스트라이드에 추가된다. 제2 캐시로부터 강등시킬, 제2 캐시 내 스트라이드들에서의 트랙들에 관한 결정이 이뤄진다. 제2 캐시로부터 강등시키기로 결정된 트랙들은 강등된다.

대 표 도 - 도1



(72) 발명자

벤헤이스, 마이클, 토마스

미국 애리조나 85744-0002, 투싼, 사우스 리타 로드 9000, 엠디:9032-1 102, 아이비엠 코포레이션

닐슨, 칼, 앨런

미국 애리조나 85744-0002, 투싼, 사우스 리타 로드 9000, 엠디:9032-1 103, 아이비엠 코포레이션

애쉬, 케빈, 존

미국 애리조나 85744-0002, 투싼, 사우스 리타 로드 9000, 엠디:9032-1 103, 아이비엠 코포레이션

명 세 서

청구범위

청구항 1

제1 캐시, 제2 캐시, 및 스토리지 시스템을 포함하는 캐시 시스템에서 데이터를 관리하는 동작들을 수행하기 위해 그 내부에 구현되며 실행하는 컴퓨터 판독가능 프로그램 코드를 갖는 컴퓨터 판독가능 스토리지 매체로서, 상기 동작들은:

상기 제1 캐시가 풀(full)인 것에 응답하여, 수정 리스트 내에 나타난 수정 비순차 트랙들(modified non-sequential tracks)을 상기 제1 캐시로부터 상기 스토리지로 디스테이지하는 단계(destaging);

상기 제1 캐시로부터 상기 스토리지로 디스테이지된 상기 수정 비순차 트랙들을 상기 제1 캐시 내 비수정 비순차 트랙들(unmodified non-sequential tracks)로서 나타내는 단계(indicating);

상기 제1 캐시 내 비수정 비순차 트랙들을 제1 비수정 리스트(a first unmodified list)에 나타내는 단계 - 상기 제1 비수정 리스트는 오직 상기 제1 캐시 내 비수정 비순차 트랙들만 나타냄 -;

상기 제1 캐시로부터 강등(demote)시킬 상기 스토리지 시스템에 저장된 상기 제1 캐시 내 비수정 트랙들의 제1 비수정 리스트에 나타난 트랙들을 결정하는 단계 - 상기 제1 캐시 및 제2 캐시 내의 각 스트라이드는 데이터의 트랙들로 파퓰레이트되고, 상기 트랙들은 상기 스토리지 시스템 내에 유지됨 -;

상기 제2 캐시의 구성에 기초하여 강등시키기로 상기 제1 비수정 리스트로부터 결정된 트랙들을 포함하는 제1 스트라이드(stride)를 형성하는 단계;

상기 제1 스트라이드 내 트랙들을 포함시키기 위한 상기 제2 캐시 내 제2 스트라이드를 결정하는 단계;

상기 제1 스트라이드로부터의 트랙들을 상기 제2 캐시 내 상기 제2 스트라이드로 추가하는 단계;

상기 제2 캐시 내 제2 스트라이드로 추가된 트랙들을 상기 제2 캐시 내 비수정 트랙들의 제2 비수정 리스트에 나타내는 단계;

상기 제2 캐시로부터 강등시킬 상기 제2 캐시 내 스트라이드들에서 상기 제2 비수정 리스트로부터의 트랙들을 결정하는 단계; 및

상기 제2 캐시에 대한 상기 제2 비수정 리스트로부터 강등시키기로 결정된 트랙들을 강등시키는 단계를 포함하는, 컴퓨터 판독가능 스토리지 매체.

청구항 2

삭제

청구항 3

삭제

청구항 4

삭제

청구항 5

삭제

청구항 6

삭제

청구항 7

삭제

청구항 8

삭제

청구항 9

삭제

청구항 10

삭제

청구항 11

삭제

청구항 12

삭제

청구항 13

스토리지 시스템과 통신하는 시스템으로서, 상기 시스템은,

프로세서;

상기 프로세서에 접근가능한 제1 캐시;

상기 프로세서에 접근가능한 제2 캐시;

컴퓨터 판독가능 스토리지 매체 - 상기 컴퓨터 판독가능 스토리지 매체는, 그 내부에 구현되고 동작들을 수행하기 위해 상기 프로세서에 의해 실행되는 컴퓨터 판독가능 프로그램 코드를 가짐 - 를 포함하며,

상기 동작들은,

상기 제1 캐시가 풀(full)인 것에 응답하여, 수정 리스트 내에 나타난 수정 비순차 트랙들(modified non-sequential tracks)을 상기 제1 캐시로부터 상기 스토리지로 디스테이지하는 단계(destaging);

상기 제1 캐시로부터 상기 스토리지로 디스테이지된 상기 수정 비순차 트랙들을 상기 제1 캐시 내 비수정 비순차 트랙들(unmodified non-sequential tracks)로서 나타내는 단계(indicating);

상기 제1 캐시 내 비수정 비순차 트랙들을 제1 비수정 리스트(a first unmodified list)에 나타내는 단계 - 상기 제1 비수정 리스트는 오직 상기 제1 캐시 내 비수정 비순차 트랙들만 나타냄 -;

상기 제1 캐시로부터 강등(demote)시킬 상기 스토리지 시스템에 저장된 상기 제1 캐시 내 비수정 트랙들의 제1 비수정 리스트에 나타난 트랙들을 결정하는 단계 - 상기 제1 캐시 및 제2 캐시 내의 각 스트라이드는 데이터의 트랙들로 파퓰레이트되고, 상기 트랙들은 상기 스토리지 시스템 내에 유지됨 -;

상기 제2 캐시의 구성에 기초하여 강등시키기로 상기 제1 비수정 리스트로부터 결정된 트랙들을 포함하는 제1 스트라이드(stride)를 형성하는 단계;

상기 제1 스트라이드 내 트랙들을 포함시키기 위한 상기 제2 캐시 내 제2 스트라이드를 결정하는 단계;

상기 제1 스트라이드로부터의 트랙들을 상기 제2 캐시 내 제2 스트라이드로 추가하는 단계;

상기 제2 캐시 내 제2 스트라이드로 추가된 트랙들을 상기 제2 캐시 내 비수정 트랙들의 제2 비수정 리스트에 나타내는 단계;

상기 제2 캐시로부터 강등시킬 상기 제2 캐시 내 스트라이드들에서 상기 제2 비수정 리스트로부터의 트랙들을 결정하는 단계; 및

상기 제2 캐시에 대한 상기 제2 비수정 리스트로부터 강등시키기로 결정된 트랙들을 강등시키는 단계를 포함하

는, 시스템.

청구항 14

삭제

청구항 15

삭제

청구항 16

삭제

청구항 17

삭제

청구항 18

삭제

청구항 19

삭제

청구항 20

삭제

청구항 21

캐시 시스템에서 데이터를 관리하는 방법으로서,

제1 캐시가 풀(a first cache full)인 것에 응답하여, 수정 리스트 내에 나타난 수정 비순차 트랙들(modified non-sequential tracks)을 상기 제1 캐시로부터 스토리지(a storage)로 디스테이지하는 단계(destaging);

상기 제1 캐시로부터 상기 스토리지로 디스테이지된 상기 수정 비순차 트랙들을 상기 제1 캐시 내 비수정 비순차 트랙들(unmodified non-sequential tracks)로서 나타내는 단계(indicating);

상기 제1 캐시 내 비수정 비순차 트랙들을 제1 비수정 리스트(a first unmodified list)에 나타내는 단계 - 상기 제1 비수정 리스트는 오직 상기 제1 캐시 내 비수정 비순차 트랙들만 나타냄 -;

상기 제1 캐시로부터 강등시킬(demote), 스토리지 시스템(a storage system)에 저장된 상기 제1 캐시 내 비수정 트랙들의 제1 비수정 리스트에 나타난 트랙들을 결정하는 단계 - 상기 제1 캐시 및 제2 캐시 내의 각 스트라이드(each stride in the first cache and a second cache)는 데이터의 트랙들로 파퓰레이트되고(populated), 상기 트랙들은 상기 스토리지 시스템 내에 유지됨 -;

상기 제2 캐시의 구성에 기초하여 강등시킬, 상기 제1 비수정 리스트로부터 결정된 트랙들을 포함하는 제1 스트라이드(a first stride)를 형성하는 단계;

상기 제1 스트라이드 내 트랙들을 포함시킬, 상기 제2 캐시 내 제2 스트라이드를 결정하는 단계;

상기 제1 스트라이드로부터의 트랙들을 상기 제2 캐시 내 상기 제2 스트라이드에 추가하는 단계;

상기 제2 캐시 내 제2 스트라이드에 추가된 트랙들을 상기 제2 캐시 내 비수정 트랙들의 제2 비수정 리스트에 나타내는 단계;

상기 제2 캐시로부터 강등시킬, 상기 제2 캐시 내 스트라이드들에서 상기 제2 비수정 리스트로부터의 트랙들을 결정하는 단계; 및

상기 제2 캐시로부터 강등시키기로 상기 제2 비수정 리스트로부터 결정된 트랙들을 강등시키는 단계를 포함하는, 방법.

청구항 22

청구항 21에 있어서, 상기 제1 캐시는 상기 제2 캐시보다 더 빠른 억세스 디바이스이고, 상기 제2 캐시는 상기 스토리지 시스템보다 더 빠른 억세스 디바이스인, 방법.

청구항 23

청구항 21에 있어서, 상기 제2 캐시로부터 강등시키기로 결정된 트랙들은 상기 제2 캐시에서 서로 다른 스트라이드들(strides)로부터 나온 것인, 방법.

청구항 24

청구항 21에 있어서, 상기 방법은,

상기 제1 캐시 내 트랙에 대한 라이트(write to track)를 수신하는 단계;

상기 라이트를 수신하는 트랙이 상기 제2 캐시에 포함되는지를 결정하는 단계; 및

상기 제1 캐시에서 라이트된 트랙이 상기 제2 캐시에 포함된다는 결정에 응답하여, 상기 제1 캐시에서 업데이트된 상기 제2 캐시 내 트랙을 무효화(invalidate)시키는 단계를 더 포함하는, 방법.

청구항 25

청구항 21에 있어서, 상기 방법은,

상기 제2 캐시 내 스트라이드들을 통합시킬지를 결정하는 단계;

스트라이드들을 통합시키는 결정에 응답하여,

어떠한 트랙들도 갖지 않는 하나의 이용가능한 스트라이드를 결정하고;

유효 트랙과 무효 트랙 둘 다를 갖는 적어도 두 개의 스트라이드들을 결정하고;

적어도 두 개의 스트라이드들로부터의 유효 트랙들을 상기 결정된 이용가능한 스트라이드에 결합시키는 것 - 상기 적어도 두 개의 스트라이드들은 상기 제1 캐시로부터 강등된 스트라이드들로부터의 트랙들을 저장하기 위해 사용가능함 - 을 수행하는 단계를 더 포함하는, 방법.

청구항 26

청구항 21에 있어서, 트랙들의 상기 제1 스트라이드를 형성하는 단계는,

데이터의 트랙들을 저장하기 위한 m 개의 디바이스들 및 적어도 하나의 패리티 디바이스 - 이는 상기 m 개의 디바이스들을 위한 데이터의 트랙들로부터 계산된 패리티 데이터(parity data)를 저장하기 위한 것임 - 를 포함하는 n 개의 디바이스들을 갖는, 상기 제2 캐시를 위해 정의된 RAID(Redundant Array of Independent Disk) 구성에 기초하여 RAID 구성을 위한 스트라이드를 형성하는 단계를 포함하는, 방법.

청구항 27

청구항 21에 있어서,

상기 제1 비수정 리스트는 상기 제1 캐시 내 트랙들을 위한 제1 LRU(least recently used) 리스트를 포함하고,

상기 제2 비수정 리스트는 상기 제2 캐시 내 트랙들을 위한 제2 LRU 리스트를 포함하는, 방법.

청구항 28

삭제

청구항 29

삭제

청구항 30

삭제

청구항 31

삭제

청구항 32

삭제

발명의 설명

기 술 분 야

[0001] 본 발명은 제1 캐시(first cache)로부터의 트랙들의 제1 스트라이드(first stride of tracks)를 제2 캐시(second cache) 내 제2 스트라이드(second stride)에 라이트(write)하기 위해 파퓰레이트(populate)하는 컴퓨터 프로그램 제품(computer program product), 시스템, 및 방법과 관련된다.

배 경 기 술

[0002] 캐시 관리 시스템(cache management system)은, 요청된 트랙들을 저장하는 스토리지 디바이스보다 더 빠른 억세스 스토리지 디바이스(예컨대, 메모리)에서 리드(read) 및 라이트(write) 동작들의 결과로서 최근에 억세스된 스토리지 디바이스 내 트랙들을 버퍼링(buffer)한다. 더 빠른 억세스 캐시 메모리 내 트랙들에 대한 계속되는 리드 요청들(read requests)은 더 느린 억세스 스토리지로부터의 요청 트랙들을 리턴하는 것보다 더 빠른 속도(rate)로 리턴되고, 따라서 리드 지연(read latency)을 감소시킨다. 캐시 관리 시스템은 또한, 스토리지 디바이스로 전달(directed)될 수정 트랙(modified track)이 캐시 메모리에 라이트될 때 그리고 그 수정 트랙이 스토리지 디바이스(예컨대, 하드 디스크 드라이브)로 외부로 라이트되기 전에, 라이트 요청(write request)에 대한 완료(complete)를 리턴할 수 있다. 스토리지 디바이스에 대한 라이트 지연(write latency)은 일반적으로 캐시 메모리에 라이트하는 지연보다 상당히 더 길다. 따라서, 캐시를 사용하는 것은 라이트 지연도 감소시킨다.

[0003] 캐시 관리 시스템은 캐시에 저장된 각각의 트랙에 대한 하나의 엔트리(entry)를 갖는 링크된 리스트(linked list)를 유지할 수 있는데, 이는 스토리지 디바이스에 라이트하기 전에 캐시에 버퍼링된 라이트 데이터(write data) 또는 리드 데이터(read data)를 포함할 수 있다. 흔히 사용되는 LRU(Least Recently Used) 캐시 기술에서는, 만약 캐시 내 트랙이 억세스되면, 즉, 캐시 "히트(hit)" 이면, 억세스된 트랙에 대한 LRU 리스트 내 엔트리는 그 리스트의 MRU(Most Recently Used) 엔드(end)로 옮겨진다. 만약 요청된 트랙이 캐시 내에 있지 않으면, 즉, 캐시 미스(cache miss)이면, 엔트리가 그 리스트의 LRU 엔드에 있는 캐시 내 트랙은 제거(또는 스토리지로 다시 디스테이지(destage))될 수 있고, 스토리지로부터 캐시 내로 스테이지(stage)된 트랙 데이터에 대한 엔트리는 LRU 리스트의 MRU 엔드에 추가된다. 이 LRU 캐시 기술에서는, 더 자주 억세스되는 트랙들이 캐시에 남을 것이고, 반면에 덜 자주 억세스되는 데이터는, 새로 억세스되는 트랙들을 위한 캐시 내 자리를 마련하기 위해, 리스트의 LRU 엔드로부터 제거될 가능성이 더 클 것이다.

[0004] 당해 기술 분야에서는 스토리지 시스템에서 캐시를 사용하기 위한 향상된 기술들에 대한 요구가 있다.

발명의 내용

과제의 해결 수단

[0005] 제1 캐시, 제2 캐시 및 스토리지 시스템을 포함하는 캐시 시스템에서 데이터를 관리하기 위한 컴퓨터 프로그램 제품, 시스템 및 방법이 제공된다. 제1 캐시로부터 강등(demote)시킬, 스토리지 시스템에 저장된 트랙들에 대한 결정이 이뤄진다. 강등시키기로 결정된 트랙들을 포함하는 제1 스트라이드(first stride)가 형성된다. 제1 스트라이드 내 트랙들을 포함시키기 위한 제2 캐시 내 제2 스트라이드(second stride)에 대한 결정이 이뤄진다. 제1 스트라이드로부터의 트랙들(tracks from the first stride)은 제2 캐시 내 제2 스트라이드에 추가된다. 제2 캐시로부터 강등시킬, 제2 캐시 내 스트라이드들에서의 트랙들에 대한 결정이 이뤄진다. 제2 캐시로부터 강등시키기로 결정된 트랙들은 강등된다.

도면의 간단한 설명

[0006] 도 1은 컴퓨팅 환경의 일 실시예를 보여준다.

도 2는 제1 캐시 관리 정보의 일 실시예를 보여준다.

도 3은 제2 캐시 관리 정보의 일 실시예를 보여준다.

도 4는 제1 캐시 컨트롤 블록의 일 실시예를 보여준다.

도 5는 제2 캐시 컨트롤 블록의 일 실시예를 보여준다.

도 6은 스트라이드 정보의 일 실시예를 보여준다.

도 7은 제2 캐시 RAID 구성(configuration)의 일 실시예를 보여준다.

도 8은 스토리지 RAID 구성의 일 실시예를 보여준다.

도 9는 제2 캐시로 승격(promote)시키기 위해 제1 캐시로부터의 비수정(unmodified) 비순차(non-sequential) 트랙들을 강등시키는 동작들의 일 실시예를 보여준다.

도 10은 제1 캐시에 트랙을 추가하는 동작들의 일 실시예를 보여준다.

도 11은 제1 스트라이드로부터 제2 스트라이드로 트랙들을 추가하는 동작들의 일 실시예를 보여준다.

도 12는 제2 캐시 내 스페이스(space)를 프리(free)하게 하는 동작들의 일 실시예를 보여준다.

도 13은 제2 캐시 내 스트라이드들을 프리하게 하는 동작들의 일 실시예를 보여준다.

도 14는 리드 요청(read request)에 리턴하기 위해 트랙들에 대한 요청을 처리하는 동작들의 일 실시예를 보여준다.

발명을 실시하기 위한 구체적인 내용

[0007] 기술된 실시예들은 캐시 승격 동작들(cache promotion operations)의 효율을 향상시키기 위해 제2 캐시 내 스트라이드들에 대한 풀 스트라이드 라이트들로서 트랙들이 라이트될 수 있도록, 스트라이드들 내 제1 캐시로부터의 트랙들을 승격시키기 위한 기술들을 제공한다. 나아가, 트랙들이 제1 캐시(14)로부터 제2 캐시(18)로 스트라이드들로 승격되고 있는 한편, 트랙들은, LRU 알고리즘과 같은 캐시 강등 알고리즘(cache demotion algorithm)에 따라 트랙 단위로(on a track basis) 제2 캐시(18)로부터 강등된다. 나아가, 불완전하게 찬(partially full), 즉 유효 및 무효 트랙들을 갖는 제2 캐시 내 스트라이드들은, 제1 캐시로부터의 트랙들의 추가(further) 스트라이드들을 수신하기 위해 제2 캐시 내 프리(free)하게 하도록 하나의 스트라이드에 결합될 수 있고, 그래서 제2 캐시가 제1 캐시 내 트랙들로부터 형성된 스트라이드들을 위해 이용가능한 프리 스트라이드들을 유지하도록 한다.

[0008] 도 1은 컴퓨팅 환경의 일 실시예를 나타낸다. 복수의 호스트들(2a, 2b, ..., 2n)은 스토리지(10)에서 볼륨(8)(예컨대, 로지컬 유닛 넘버들(Logical Unit Numbers), 로지컬 디바이스들(Logical Devices), 로지컬 서브시스템들(Logical Subsystems) 등)에 있는 데이터를 억세스하기 위해 네트워크(6)를 통해 스토리지 컨트롤러(4)로 입력/출력(I/O) 요청을 내보낼(submit) 수 있다. 스토리지 컨트롤러(4)는 단일 또는 복수의 코어들을 갖는 하나 또는 그 이상의 프로세서들을 포함하는 프로세서 컴플렉스(12), 제1 캐시(14) 및 제2 캐시(18)를 포함한다. 제1 캐시(14) 및 제2 캐시(18)는 호스트들(2a, 2b, ..., 2n)과 스토리지(10) 사이에서 전송되는 캐시 데이터를 캐시(cache)한다.

[0009] 스토리지 컨트롤러(4)는 메모리(20)를 갖는다. 메모리(20)는 호스트들(2a, 2b, ..., 2n)과 스토리지(10) 사이에 전송되는 트랙들의 전송을 관리하기 위한 스토리지 매니저(22)와, 제1 캐시(14) 및 제2 캐시(18)에서 호스트들(2a, 2b, ...2n)과 스토리지(10) 사이에 전송되는 데이터를 관리하는 캐시 매니저(24)를 포함한다. 트랙은 트랙, 로지컬 블록 어드레스(Logical Block Address, LBA) 등과 같은 스토리지(10)에 구성된 데이터의 모든 유닛을 포함할 수 있고, 이는 볼륨(volume), 로지컬 디바이스 등과 같은 트랙들의 더 큰 그룹핑의 일부이다. 캐시 매니저(24)는 제1 캐시(14) 및 제2 캐시(18)에서 리드 (비수정) 트랙들 및 라이트 (수정) 트랙들을 관리하기 위해, 제1 캐시 관리 정보(26) 및 제2 캐시 관리 정보(28)를 유지한다.

[0010] 스토리지 매니저(22) 및 캐시 매니저(24)는 메모리(20) 내로 로드되고 프로세서 컴플렉스(12)에 의해 실행되는 프로그램 코드로서, 도 1에 도시되어 있다. 이와는 다르게, 기능들(functions)의 일부 또는 전부는, ASICs(Application Specific Integrated Circuits)에서와 같이, 스토리지 컨트롤러(4) 내 하드웨어 디바이스들

에 구현될 수 있다.

[0011] 제2 캐시(18)는 로그 구조 어레이(log structured array)(LSA)(32)에 트랙들을 저장할 수 있고, 여기서 트랙들은 수신되는 순서로 순차적으로 라이트되고, 이렇게 하여, 제2 캐시(18)에 라이트된 트랙들의 일시적인 순서화(temporal ordering)를 제공한다. LSA에서, LSA에 이미 존재하는 트랙들의 나중 버젼들은 LSA(32)의 엔드(end)에 라이트된다. 다른 실시예들에서, 제2 캐시(18)는 LSA에서와는 다른 포맷들로 데이터를 저장할 수 있다.

[0012] 메모리(20)는 제2 캐시 RAID 구성 정보(34)를 더 포함한다. 제2 캐시 RAID 구성 정보(34)는 제2 캐시(18)에 저장할 트랙들의 스트라이드를 형성하는 방법을 결정하기 위해 사용되는 RAID 구성(configuration)에 관한 정보를 제공한다. 일 실시예에서, 제2 캐시(18)는 독립된 솔리드 스테이트 스토리지 디바이스들(SSDs, solid state storage devices))과 같은 복수의 스토리지 디바이스들을 포함할 수 있다. 그리하여, 제1 캐시(14)로부터의 트랙들로 형성된 스트라이드들이, 플래쉬 메모리들과 같은 제2 캐시(18)를 형성하는 독립된 스토리지 디바이스들에 걸쳐 스트라이프(striped across) 되도록 한다. 또 다른 실시예에서, 제2 캐시(18)는 하나의 플래쉬 메모리와 같은 단일 스토리지 디바이스를 포함할 수 있다. 그리하여, 트랙들이 제2 캐시 RAID 구성(34)에 의해 정의된 바와 같이 스트라이드들에 그룹핑되도록 하지만, 그 트랙들은, 제2 캐시(18)를 구현하는 하나의 플래쉬 메모리와 같은 단일 디바이스에 대한 스트라이드들로 라이트되도록 한다. 제2 캐시 RAID 구성(34)을 위해 구성된 스트라이드들의 트랙들은 제2 캐시(18) 디바이스 내 LSA(32)에 라이트될 수 있다. 제2 캐시 RAID 구성(34)은 서로 다른 RAID 레벨들 - 예컨대, 레벨들 5, 10 등 - 을 명시할 수 있다.

[0013] 메모리(20)는 스토리지 RAID 구성 정보(36)를 더 포함할 수 있다. 스토리지 RAID 구성 정보(36)는, 만약 제2 캐시(18)가 수정 데이터(modified data)를 저장해야 한다면, 제1 캐시(14) 또는 제2 캐시(18)로부터의 트랙들을, 스토리지 시스템(10)으로 라이트하는 방법을 결정하기 위해 사용되는 RAID 구성에 관한 정보를 제공한다. 여기서 디스테이지되는 스트라이드(destaged stride) 내 트랙들은, 스토리지 시스템(10)에서, 디스크 드라이브들과 같은 스토리지 디바이스들에 걸쳐 스트라이프된다.

[0014] 일 실시예에서, 제1 캐시(14)는, DRAM(Dynamic Random Access Memory)과 같은 RAM(Random Access Memory)을 포함할 수 있고, 제2 캐시(18)는 솔리드 스테이트 디바이스와 같은 플래쉬 메모리를 포함할 수 있고, 스토리지(10)는 하드 디스크 드라이브들 및 자기 테이프와 같은 하나 또는 그 이상의 순차 억세스 스토리지 디바이스들을 포함한다. 스토리지(10)는 단일의 순차 억세스 스토리지 디바이스를 포함할 수 있고, 또는 JBOD(Just a Bunch of Disks), DASD(Direct Access Storage Device), RAID(Redundant Array of Independent Disks) 어레이, 가상 디바이스(virtualization device) 등과 같은 스토리지 디바이스들의 어레이를 포함할 수 있다. 일 실시예에서, 제1 캐시(14)는 제2 캐시(18)보다 더 빠른 억세스 디바이스이고, 제2 캐시(18)는 스토리지(10)보다 더 빠른 억세스 디바이스이다. 나아가, 제1 캐시(14)는 제2 캐시(18)보다 더 큰 스토리지의 유닛 당 코스트를 가질 수 있고, 제2 캐시(18)는 스토리지(10) 내 스토리지 디바이스들보다 더 큰 스토리지의 유닛 당 코스트를 가질 수 있다.

[0015] 제1 캐시(14)는 메모리(20)의 일부일 수 있고, 또는 DRAM과 같이 독립된 메모리 디바이스에 구현될 수 있다.

[0016] 네트워크(6)는 SAN(Storage Area Network), LAN(Local Area Network), WAN(Wide Area Network), 인터넷, 및 인트라넷 등을 포함할 수 있다.

[0017] 도 2는 제1 캐시 관리 정보(26)의 일 실시예를 나타낸다. 제1 캐시 관리 정보(26)는, 컨트롤 블록 디렉토리(52)에서 블록들을 컨트롤하기 위해 제1 캐시(14) 내 트랙들의 인덱스를 제공하는 트랙 인덱스(50); 제1 캐시(14) 내 비수정(unmodified) 순차(sequential) 트랙들의 일시적인 순서화를 제공하는 비수정 순차 LRU 리스트(54); 제1 캐시(14) 내 수정 순차 및 비순차 트랙들의 일시적인 순서화를 제공하는 수정 LRU 리스트(56); 제1 캐시(14) 내 비수정 비순차 트랙들의 일시적인 순서화를 제공하는 비수정 비순차 LRU 리스트(58); 및 풀 스트라이드 라이트(full stride write)와 같이 제2 캐시(18)에 라이트할 제1 캐시(14) 내 비수정 비순차 트랙들로 형성된 스트라이드들에 관한 정보를 제공하는 스트라이드 정보(60)를 포함한다.

[0018] 어떤 실시예들에서, 제1 캐시(18)를 풀(full)로 결정할 때, 수정 LRU 리스트(56)는 제1 캐시(14)로부터 스토리지(10)로 수정 트랙들을 디스테이지하기 위해 사용되며, 그래서 그것들의 디스테이지된 트랙들의 사본이 제1 캐시(18)에 있도록 한다.

[0019] 수정 비순차 트랙이 제1 캐시(14)로부터 스토리지(10)로 디스테이지되는 경우, 캐시 매니저(24)는 제1 캐시(14) 내 비수정 비순차 트랙으로 그 디스테이지된 트랙들을 지정할 수 있고, 새롭게 지정된 비수정 트랙의 인디케이션(indication)을 비수정 비순차 LRU 리스트(58)에 추가하며, 이로부터 그것은 제2 캐시(14)로 승격될 대상이

된다. 디스테이지된 수정 트랙의 상태는, 그 디스테이지된 수정 비순차 트랙을 필드(106)에서 비수정으로 나타내기 위해 제1 캐시 컨트롤 블록(104)을 업데이트함으로써 변경될 수 있다. 따라서, 제1 캐시(14) 내 비수정 비순차 트랙들은 수정 LRU 리스트(56)에 따라 스토리지(10)에 디스테이지된 수정 비순차 트랙들 또는 리드 데이터를 포함할 수 있다. 따라서, LRU 리스트(58)에서 비수정 트랙들이 되는 디스테이지된 수정 트랙들은 후속되는 리드 요청들을 위해 이용가능하게 되도록 제2 캐시(14)로 승격(promote)될 수 있다. 이들 실시예들에서, 제2 캐시(14)는 비수정 비순차 트랙들을 캐시하기 위한 읽기 전용 캐시(read only cache)를 포함한다.

[0020] 도 3은 제2 캐시 관리 정보(28)의 일 실시예를 나타낸다. 제2 캐시 관리 정보(28)는 컨트롤 블록 디렉토리(72)에서 블록들을 컨트롤하기 위해 제2 캐시(18) 내 트랙들의 인덱스를 제공하는 트랙 인덱스(70); 제2 캐시(18) 내 비수정 트랙들의 일시적인 순서화를 제공하는 비수정 리스트(74); 및 제2 캐시(18)에 라이트된 트랙들의 스트라이드들에 관한 정보를 제공하는 스트라이드 정보(78)를 포함한다. 일 실시예에서, 제2 캐시(18)는 단지 비수정, 비순차 트랙들을 저장한다. 다른 실시예들에서, 제2 캐시(18)는 수정 및/또는 순차 트랙들도 저장할 수 있다.

[0021] 모든 LRU 리스트들(54, 56, 58 및 74)은, 식별된 트랙이 마지막으로 억세스된 순서대로 정렬된 제1 캐시(14) 및 제2 캐시(18) 내 트랙들의 트랙 ID들을 포함할 수 있다. LRU 리스트들(54, 56, 58, 및 74)은 가장 최근에 억세스된 트랙을 나타내는 MRU(most recently used) 엔드 및 최근에 가장 덜 사용되거나 가장 적게 억세스된 트랙을 나타내는 LRU 엔드를 갖는다. 캐시들(14 및 18)에 추가된 트랙들의 트랙 ID들은 LRU 리스트의 MRU 엔드에 추가되고, 캐시들(14 및 18)로부터 강등된 트랙들은 LRU 엔드로부터 억세스된다. 트랙 인덱스들(50 및 70)은 스캐터 인덱스 테이블(scatter index table, SIT)를 포함할 수 있다. 캐시들(14 및 18)에서 트랙들의 일시적인 순서화를 제공하기 위해 또 다른 유형의 데이터 구조들이 사용될 수 있다.

[0022] 비순차 트랙들은 OLTP(Online Line Transaction Processing) 트랙들을 포함할 수 있고, 이는 흔히 완전히 랜덤하지 않고(not fully random) 어느 정도의 참조 구역성(locality)을 갖는, 즉 반복적으로 억세스될 가능성을 갖는, 작은 블록 라이트들을 포함한다.

[0023] 도 4는 컨트롤 블록 디렉토리(52)에서 제1 캐시 컨트롤 블록(100)의 일 실시예를 나타낸다. 제1 캐시 컨트롤 블록(100)은 컨트롤 블록 식별자(ID)(102), 제1 캐시(14) 내 트랙의 물리적 위치의 제1 캐시 위치(104), 트랙이 수정(modified)인지 비수정(unmodified)을 나타내는 정보(106), 트랙이 순차(sequential) 억세스인지 비순차(non-sequential) 억세스인지를 나타내는 정보(108), 예를 들어, 강등 없음(no demotion), 강등 준비(ready to demote), 그리고 강등 완료(demote complete)와 같이, 트랙에 대한 강등 상태(demote status)를 나타내는 정보(110)를 포함한다.

[0024] 도 5는 제2 캐시 컨트롤 블록 디렉토리(72)에서 제2 캐시 컨트롤 블록(120)의 일 실시예를 나타낸다. 제2 캐시 컨트롤 블록(120)은 컨트롤 블록 식별자(ID)(122); LSA(32)에 트랙이 위치하는 LSA 위치(124); 트랙이 수정인지 비수정인지를 나타내는 수정/비수정 정보(126); 트랙이 유효인지 무효인지를 나타내는 유효/무효 플래그(128)를 포함한다. 제2 캐시(18) 내 트랙은, 그 트랙이 제1 캐시(14)에서 업데이트된다면, 또는 그 트랙이 제2 캐시(18)로부터 강등된다면, 무효로 나타내어진다.

[0025] 수정 비순차 트랙이 제1 캐시(14)로부터 스토리지(10)로 디스테이지되는 경우, 캐시 매니저(24)는 그 디스테이지된 트랙들을 제1 캐시(14) 내 비수정 비순차 트랙으로 지정할 수 있고, 비수정 비순차 LRU 리스트(50)에 그 새롭게 지정된 비수정 트랙의 인디케이션(indication)을 추가할 수 있으며, 이로부터 그것은 제2 캐시(14)로 승격될 대상이 된다. 그 디스테이지된 수정 트랙의 상태는, 그 디스테이지된 수정 비순차 트랙을 필드(106)에서 비수정으로 나타내기 위해 제1 캐시 컨트롤 블록(100)을 업데이트함으로써 변경될 수 있다. 따라서, 제1 캐시(14) 내 비수정 비순차 트랙들은 수정 LRU 리스트(56)에 따라 스토리지(10)로 디스테이지된 수정 비순차 트랙들 또는 리드 데이터를 포함할 수 있다. 따라서, LRU 리스트(58)에서 비수정 트랙들이 되는 디스테이지된 수정 트랙들은, 후속되는 리드 요청들을 위해 이용가능하게 되도록 제2 캐시(14)로 승격될 수 있다. 이들 실시예들에서, 제2 캐시(14)는 비수정 비순차 트랙들을 캐시하기 위한 읽기 전용 캐시(read only cache)를 포함한다.

[0026] 도 6은 제2 캐시(18)에 형성될 하나의 스트라이드를 위한 스트라이드 정보(60, 78)의 인스턴스(instance)(130)를 나타낸다. 인스턴스(130)는 스트라이드 식별자(ID)(132), 스트라이드(132)에 포함된 스토리지(10)의 트랙들(134), 트랙들의 총 수 중 스트라이드 내 유효 트랙들의 수를 나타내는 점유율(occupancy)(136)을 포함하며, 여기서 유효하지 않은 스트라이드 내 트랙들은 가비지 수집 동작들(garbage collection operations)의 대상이 된다.

[0027] 도 7은 제1 캐시(14) 내 트랙들로부터 제2 캐시(18) 내 트랙들의 스트라이드들을 형성하는 방법을 결정하기 위해 유지되는 제2 캐시 RAID 구성(34)의 일 실시예를 나타낸다. RAID 레벨(140)은, 예컨대, RAID 1, RAID 5, RAID 6, RAID 10 등을 사용하기 위한 RAID 구성, 사용자 데이터의 트랙들을 저장하는 데이터 디스크들의 수(m)(142), 그리고 데이터 디스크들(142)로부터 계산된 패리티(parity)를 저장하는 패리티 디스크들의 수(p)(144)를 나타내며, 여기서 p는 하나 또는 그 이상일 수 있고, 이는 계산된 패리티 블록들을 저장하기 위한 디스크들의 수를 나타낸다. 비수정 패리티 선택적 플래그(146)는, 제2 캐시(18)로 승격되고 있는 제1 캐시(14) 내 비수정 비순차 트랙들을 위해 패리티가 계산되어야 하는지 여부를 나타낸다. 이 선택적 플래그(146)는, 스트라이드 내 비수정 비순차 트랙들을 단지 포함시키기 위해, 비수정 비순차 트랙들만으로 그 스트라이드를 채우도록 허용한다. 제1 캐시(14) 내 비수정 비순차 트랙들의 스트라이드는 LSA(32)에 나타내어질 수 있고, 여기서 스트라이드의 트랙들은 제2 캐시(18)를 형성하는 m 플러스 p 개의 스토리지 디바이스들에 걸쳐 스트라이프된다. 이와는 다르게, 제2 캐시(18)는 n 개 보다 더 적은 디바이스들을 포함할 수 있다.

[0028] 도 8은 스토리지(10)의 디스크들에 걸쳐 스트라이프하도록 제2 캐시(18) 내 수정 트랙들의 스트라이드들을 형성하는 방법을 결정하기 위해 유지되는 스토리지 RAID 구성(36)의 일 실시예를 나타낸다. RAID 레벨(150)은 사용할 RAID 구성, 사용자 데이터의 트랙들을 저장하는 데이터 디스크들의 수(m)(152), 및 데이터 디스크들(152)로부터 계산된 패리티를 저장하는 패리티 디스크들의 수(p)(154)를 나타내며, 여기서 p는 하나 또는 그 이상일 수 있으며, 계산된 패리티 블록들을 저장하기 위한 디스크들의 수를 나타낸다. 제2 캐시(18)로부터의 트랙들의 스트라이드는 스토리지 시스템(10) 내 디스크들에 걸쳐 스트라이프될 수 있다.

[0029] 일 실시예에서, 제2 캐시 RAID 구성(34) 및 스토리지 RAID 구성(36)은, 서로 다른 RAID 레벨들, 데이터 디스크들, 패리티 디스크들 등과 같은 서로 다른 파라미터들을 제공할 수도 있고, 또는 동일한 파라미터들을 가질 수 있다.

[0030] 도 9는 제2 캐시(18)로 승격시키기 위해 제1 캐시(14)로부터의 비수정 비순차 트랙들을 강등시키도록 캐시 매니저(24)에 의해 수행되는 동작들의 일 실시예를 나타내며, 여기서 비수정 비순차 트랙들은 스페이스가 필요할 때, 비수정 비순차 LRU 리스트(58)의 LRU 엔드로부터 선택될 수 있다. 선택된 비수정 비순차 트랙들을 강등시키기 위한 동작들을 개시할 때(블록 200에서), 강등시키기로 선택되는 비수정 비순차 트랙들의 강등 상태(110, 도 4)는 "준비(ready)"로 세팅된다(블록 202에서). 캐시 매니저(24)는 제2 캐시(18) 내 스트라이드로 승격시키기 위해 제1 캐시(114)로부터의 트랙들의 제1 스트라이드를 형성하도록 제2 캐시 RAID 구성 정보(34)를 사용한다(블록 204에서). 예를 들어, 트랙들의 제1 스트라이드를 형성하는 것은, 데이터의 트랙들을 저장하기 위한 m 개의 디바이스들을 포함하는 n 개의 디바이스들과, m 개의 디바이스들에 대해 데이터의 트랙들로부터 계산된 패리티 데이터를 저장하기 위한 적어도 하나의 패리티 디바이스(p)를 갖는 것과 같이, 제2 캐시를 위해 정의된 RAID 구성(34)에 기초하여 RAID 구성을 위한 스트라이드를 형성하는 것을 포함한다. 나아가, 제2 캐시가 적어도 n 개의 솔리드 스테이트 스토리지 디바이스들을 포함하는 실시예들에서, 트랙들의 제1 스트라이드는, 제2 스트라이드를 형성하기 위한 패리티 없이, n 개의 솔리드 스테이트 스토리지 디바이스들에 걸쳐 스트라이프될 수 있다.

[0031] 캐시 매니저(24)는 컨트롤 블록들(100)에서 "준비(ready)" 강등 상태(110)를 갖는 비수정 비순차 트랙들의 수를 결정하기 위해 비수정 비순차 LRU(58) 리스트를 처리한다(블록 206에서). 만약 캐시 매니저(24)가, 비수정 비순차 트랙들의 수가 스트라이드를 형성하기에 충분하다고 결정하면(블록 208에서), 캐시 매니저(24)는 "준비" 강등 상태(110)를 갖는 비수정 비순차 트랙들의 제1 스트라이드를 파퓰레이트한다(블록 210에서). 일 실시예에서, 제1 스트라이드는 비수정 비순차 LRU 리스트(58)의 LRU 엔드로부터 시작하여 파퓰레이트될 수 있고, 스트라이드에서 데이터 디스크들을 위한 충분한 트랙들을 사용한다. 만약 (블록 212에서) RAID 구성이 패리티 디스크들을 명시한다면, 캐시 매니저(24)는 스트라이드에 포함된 비수정 비순차 트랙들을 위한 패리티를 계산하고(블록 212에서), 스트라이드에 패리티 데이터(p 패리티 디스크들을 위해)를 포함시킨다. 만약 (블록 208에서) 제1 캐시(14) 내 비수정 비순차 트랙들이 제1 스트라이드를 채우기에 충분하지 않다면, 제1 스트라이드를 파퓰레이트하기 위해 이용가능한 강등 준비 상태를 갖는 비수정 비순차 트랙들의 충분한 수가 있을 때까지 컨트롤은 종료된다.

[0032] 제1 스트라이드를 파퓰레이트한 후(블록들 210 및 212에서), 캐시 매니저(14)는 제1 스트라이드로부터의 트랙들을 포함시키기 위해 제2 캐시(18)에서 프리한 제2 스트라이드를 결정한다(블록 214에서). 제1 스트라이드로부터의 트랙들은 제2 캐시(18)를 형성하는 디바이스들에 걸쳐(across) 제2 스트라이드에 대한 풀 스트라이드 라이트(full stride write)로서 라이트되거나 스트라이프된다(블록 216에서). 제2 캐시(18) 내 제2 스트라이드를 제1 스트라이드로부터의 트랙들로 채울 때, 캐시 매니저(14)는 제2 스트라이드에 대한 스트라이드 정보(130)의 점유율(occupancy)(136)을 풀(full)로 나타낸다(블록 218에서). 캐시 매니저(24)는 스트라이드에 포함된 비수정 비

순차 트랙들에 대한 강등 상태(110)를 강등 "완료"로 업데이트한다(블록 220에서).

[0033] 비록 도 9의 동작들은 제2 캐시(18)로 승격시키기 위해 제1 캐시(14)로부터 비수정 비순차 트랙들을 강등시키는 것으로 기술되고 있지만, 다른 실시예들에 있어서, 동작들은 수정(modified), 순차(sequential) 등과 같은 다른 유형의 트랙들을 강등시키는 것에 적용될 수 있다.

[0034] 기술된 실시예들에 따라, 제1 캐시(14)로부터의 비수정 트랙들이 모이고 제2 캐시(18)에 스트라이드로 라이트되어, 하나의 입력/출력(I/O) 동작이 다수의 트랙들을 전송하는 데에 사용되도록 한다.

[0035] 도 10은 제1 캐시(14)에 트랙을 추가, 즉 승격시키기 위해 캐시 매니저(24)에 의해 수행되는 동작들의 일 실시예를 나타내며, 여기서 트랙은, 호스트(2a, 2b, ..., 2n)로부터의 라이트 또는 수정 트랙, 리드 요청 대상이 되고 제1 캐시(14)로 이동된 결과로서의 제2 캐시(18) 내 비순차 트랙, 또는 캐시(14 또는 18)에서 발견되지 않고 스토리지(10)로부터 리트리브되는 리드 요청된 데이터를 포함할 수 있다. 제1 캐시(14)에 추가할 트랙을 수신할 때(블록 250에서), 만약 트랙의 사본이 제1 캐시(14)에 이미 포함되어 있다면, 즉, 그 수신된 트랙이 라이트이면, 캐시 매니저(24)는 제1 캐시(14) 내 트랙을 업데이트한다(블록 254에서). 만약 (블록 252에서) 트랙들이 캐시에 이미 있지 않으면, 캐시 매니저(24)는 제1 캐시(14) 내 위치(104) 그리고 그 트랙이 수정/비수정(106) 인지 그리고 순차/비순차(108)인지를 나타내는 것을 추가하기 위해 그 트랙에 대한 컨트롤 블록(100, 도 4)을 생성한다(블록 256에서). 이 컨트롤 블록(100)은 제1 캐시(14)의 컨트롤 블록 디렉토리(52)에 추가된다. 캐시 매니저(24)는, 추가할 트랙의 트랙 ID 및 컨트롤 블록 디렉토리(52)에서 생성된 캐시 컨트롤 블록(100)에 대한 인덱스를 갖는 제1 캐시 트랙 인덱스(50)에 엔트리를 추가한다(블록 258에서). 엔트리는 추가할 트랙의 트랙 유형의 LRU 리스트(54, 56 또는 58)의 MRU 엔드(end)에 추가된다(블록 260에서). 만약 (블록 262에서) 추가할 트랙이 수정 비순차 트랙이면 그리고 만약, 제2 캐시 트랙 인덱스(70)로부터 결정되는 것에 따라, (블록 264에서) 추가할 트랙의 사본이 제2 캐시(18)에 있다면, 제2 캐시(18) 내 트랙의 사본은, 예를 들어, 제2 캐시(18) 내 트랙을 위한 캐시 컨트롤 블록(120)에서 유효/무효 플래그(128)를 무효로 세팅하는 것에 의한 것과 같이, 무효화된다(블록 266에서). 만약 (블록 306에서) 추가할 트랙이 비수정 비순차이면, 컨트롤은 종료된다.

[0036] 도 11은 제1 캐시(14)로부터의 제1 스트라이드로부터 제2 캐시(18) 내 제2 스트라이드로 트랙들을 추가하기 위해 캐시 매니저(24)에 의해 수행되는 동작들의 일 실시예를 나타낸다. 캐시 매니저(24)는 추가되고 있는 제1 스트라이드로부터의 트랙들(134)을 나타내고 점유율(occupancy)(136)을 풀(full)로 나타내는 제2 스트라이드를 위한 스트라이드 정보(130, 도 6)를 생성한다(블록 302). 추가되고 있는 제1 스트라이드 내 각각의 트랙에 대해, 동작들의 루프는 블록들 304 내지 318에서 수행된다. 캐시 매니저(24)는 제2 캐시(18)에서 LSA(32)로 승격되고 있는 트랙의 인디케이션(indication), 예컨대, 트랙 ID와 같은 것을 추가한다(블록 302에서). 만약 (블록 308에서) 추가되고 있는 트랙이 이미 제2 캐시(18)에 있으면, 캐시 매니저(24)는, LSA(32) 내 위치(124), 데이터가 비수정(126), 그리고 트랙이 유효(128)라는 것을 나타내는 트랙에 대해 캐시 컨트롤 블록(120)을 업데이트한다(블록 310에서). 만약 (블록 308에서) 트랙이 이미 제2 캐시(18)에 있지 않으면, 캐시 매니저(24)는, LSA(32)에서의 트랙 위치(124), 그리고 트랙이 수정/비수정(126)인지를 나타내는 것을 추가하기 위해 트랙에 대해 컨트롤 블록(120, 도 5)을 생성한다(블록 312에서). 하나의 엔트리가 승격된 트랙의 트랙 ID 그리고 제2 캐시(18)를 위한 컨트롤 블록 디렉토리(72)에서의 생성된 캐시 컨트롤 블록(120)에 대한 인덱스를 갖는, 제2 캐시 트랙 인덱스(70)에 추가된다. 블록 310 내지 316으로부터, 캐시 매니저(24)는 트랙 ID를 MRU 엔드에 추가하는 것에 의한 것과 같이, 비수정 LRU 리스트(74)의 MRU 엔드에서 승격된 트랙을 나타낸다(블록 316에서).

[0037] 도 12는 제2 캐시(18)에 추가할 새로운 트랙들, 즉, 제1 캐시(14)로부터 강등되어 있는 트랙들을 위해 제2 캐시(18) 내 스페이스를 프리(free)하게 하기 위해 캐시 매니저(24)에 의해 수행되는 동작들의 일 실시예를 나타낸다. 이 동작을 개시할 때(블록 350에서), 캐시 매니저(24)는, 비수정 LRU 리스트(74)의 LRU 엔드로부터 제2 캐시(18) 내 비수정 트랙들을 결정하고(블록 352에서), 무효화된 비수정 트랙들을 스토리지(10)로 디스테이지하지 않고 결정된 비수정 트랙들을 무효화시키며(블록 354에서), 또한 비수정 LRU 리스트(74)로부터 무효화된 비수정 트랙들을 제거하고 그 트랙에 대한 캐시 컨트롤 블록(120)에서 무효(128)인 것으로 그 트랙을 나타낸다. 제2 캐시(18)에서 비수정 트랙들은, 제1 캐시(14)에 추가된 리드 트랙들 또는 제1 캐시(14)로부터 디스테이지된 수정 트랙들을 포함할 수 있다. 나아가, 제2 캐시(18)로부터의 강등을 위해 캐시 매니저(24)에 의해 선택된 트랙들은 제2 캐시(18)에 형성된 서로 다른 스트라이드들로부터 나온 것(from)일 수 있다. 나아가, 제2 캐시 내 스트라이드들은 유효 트랙과 무효 트랙 둘 다를 포함할 수 있고, 여기서 트랙들은 제2 캐시(18)로부터의 강등에 의해, 또는 제1 캐시(18)에서 업데이트되고 있는 트랙에 의해, 무효화된다.

[0038] 어떤 실시예들에서, 캐시 매니저(24)는, 강등시킬 트랙들을 결정하기 위해, 각각 제1 캐시(14) 및 제2 캐시(1

8)를 위한 독립된 LRU 리스트들(58 및 74)을 사용함으로써 제1 캐시(14) 및 제2 캐시(18)로부터 강등시킬 트랙들을 결정하기 위한 서로 다른 트랙 강등 알고리즘들을 사용한다. 제1 캐시(14) 및 제2 캐시(18)에서 강등을 위한 트랙들을 선택하기 위해 사용되는 알고리즘들은, 먼저 강등시킬 트랙들을 결정하기 위해 제1 캐시(14) 및 제2 캐시(18)에서 트랙들의 특징들을 고려할 수 있다.

[0039] 도 13은 제1 캐시(14)에서 트랙들의 스트라이드들을 위해 이용가능하도록 하기 위해 제2 캐시(18) 내 스트라이드들을 프리(free)하게 하기 위한 캐시 매니저(24)에 의해 수행되는 동작들의 일 실시예를 나타낸다. 제2 캐시(18) 내 스트라이드들을 프리하게 하기 위한 동작을 개시할 때(블록 370에서), 캐시 매니저는, 프리 스트라이드들(free strides)의 수, 즉 0의 점유율(136)을 갖는 스트라이드들이 프리 스트라이드 스레쉬홀드(free stride threshold)보다 작은지를 결정한다(블록 372에서). 예를 들어, 캐시 매니저(24)는 제1 캐시(14) 트랙들로부터 형성된 스트라이드들을 위해 이용가능하게 될 프리 스트라이드들 중 적어도 둘 또는 다른 어떤 개수가 항상 있다는 것을 보장할 수 있다. 만약 프리 스트라이드들의 수가 스레쉬홀드보다 작지 않다면, 컨트롤은 종료된다. 그렇지 않고, 만약 프리 스트라이드들의 수가 스레쉬홀드보다 작다면(블록 372에서), 캐시 매니저(24)는 0의 점유율(136)을 갖는 이용가능한 스트라이드(136)를 결정하고(블록 374에서), 불완전하게 찬(partially full), 즉, 유효 트랙들이 프리 스트라이드에 적합(fit into)할 수 있는 유효 및 무효 트랙들을 갖는, 적어도 두 개의 스트라이드들을 결정한다(블록 376에서). 캐시 매니저(24)는 상기 결정된 이용가능한 스트라이드에 상기 결정된 적어도 두 개의 불완전하게 찬 스트라이드들로부터의 유효 트랙들을 결합한다(블록 378에서). 그런 다음, 캐시 매니저(24)는 적어도 두 개의 스트라이드들 - 이들로부터 트랙들은 0의 점유율(136)을 갖는 것으로 병합됨 - 을 나타낸다(블록 380에서). 그래서 그것들은 제1 캐시(14)로부터의 스트라이드들로부터 트랙들을 수신하기 위해 이용가능하다.

[0040] 도 14는 캐시들(14 및 18) 및 스토리지(10)로부터의 리드 요청에 대해 요청된 트랙들을 리트리브(retrieve)하기 위한 캐시 매니저(24)에 의해 수행되는 동작들의 일 실시예를 나타낸다. 리드 요청을 처리하는 스토리지 매니저(22)는 요청된 트랙들을 위해 캐시 매니저(24)에 요청들을 내보낼 수 있다. 트랙들에 대한 요청을 수신할 때(블록 450에서), 캐시 매니저(24)는 요청된 트랙들 모두가 제1 캐시(14)에 있는지를 결정하기 위해 제1 캐시 트랙 인덱스(50)를 사용한다(블록 454에서). 만약 (블록 454에서) 모든 요청된 트랙들이 제1 캐시(14)에 있지 않으면, 캐시 매니저(24)는 제1 캐시(14)에서가 아니라 제2 캐시(18)에서, 요청된 트랙들(any of the requested tracks)을 결정하기 위해 제2 캐시 트랙 인덱스(70)를 사용한다(블록 456에서). 만약 (블록 458에서) 제1 캐시(14) 및 제2 캐시(18)에서 어떠한 요청된 트랙들도 발견되지 않으면, 캐시 매니저(24)는, 제1 캐시(14) 및 제2 캐시(18)에서가 아니라, 제2 캐시 트랙 인덱스(70)로부터, 스토리지(10)에서, 요청된 트랙들(any of the requested tracks)을 결정한다(블록 460에서). 그런 다음, 캐시 매니저(24)는 제2 캐시(18) 및 스토리지(10)에서의 결정된 트랙들(any of the determined tracks)을 제1 캐시(14)로 승격시킨다(블록 462에서). 캐시 매니저(24)는 상기 리드 요청에 대해 리턴하기 위해 제1 캐시(14)로부터의 요청된 트랙들을 리트리브하도록 제1 캐시 트랙 인덱스(50)를 사용한다(블록 464에서). 상기 리트리브된 트랙들을 위한 엔트리들은 상기 리트리브된 트랙들을 위한 엔트리들을 포함하는 LRU 리스트(54, 56, 58)의 MRU 엔드로 이동된다(블록 466에서).

[0041] 도 13의 동작들에서, 캐시 매니저(24)는 가장 높은 레벨의 캐시(14)로부터 요청된 트랙들을 리트리브하며, 그런 다음 스토리지(10)로 가기 전에 제2 캐시(18)로부터 리트리브한다. 왜냐하면 캐시들(14 및 18)이 요청된 트랙의 가장 최근에 수정된 버젼을 가질 것이기 때문이다. 가장 최근의 버젼은 제1 캐시(14)에서 먼저 발견되고, 그런 다음, 만약 제1 캐시(14)에 있지 않으면 제2 캐시(18)에서 발견되고, 그런 다음 어느 캐시(14, 18)에도 있지 않으면 스토리지(10)에서 발견된다.

[0042] 기술된 실시예들은 제2 캐시를 위한 RAID 구성에 따라 정의된 스트라이드들에서 제1 캐시 내 트랙들을 그룹핑하기 위한 기술들을 제공한다. 그래서 제1 캐시 내 트랙들이 제2 캐시에 대하여 스트라이드들 내에 그룹핑될 수 있도록 한다. 그런 다음 제2 캐시에 캐시된 트랙들은 스토리지를 위한 RAID 구성에 따라 정의된 스트라이드들 내로 그룹핑될 수 있고, 그런 다음, 스토리지 시스템으로 라이트될 수 있다.

[0043] 기술된 실시예들은 스트라이드들에서 제1 캐시로부터 트랙들을 승격시키기 위한 기술들을 제공하며, 그래서 캐시 승격 동작들의 효율을 향상시키기 위해 트랙들이 제2 캐시 내 스트라이드들에 대한 풀 스트라이드 라이트들(full stride writes)로서 라이트될 수 있도록 한다. 기술된 실시예들은, 단일 I/O 동작으로 제2 캐시로 전체 스트라이드를 승격시킴에 의해 리소스들을 보존하기 위해, 풀 스트라이드 라이트들이 제1 캐시에서 강등된 트랙들을 제2 캐시로 승격시키기 위해 사용될 수 있도록 한다.

[0044] 나아가, 스트라이드들로 트랙들이 제1 캐시(14)로부터 제2 캐시(18)로 승격되고 있을 때, 트랙들은 LRU 알고리

즘과 같은 캐시 강등 알고리즘(cache demotion algorithm)에 따라, 트랙 단위로(on a track-by-track basis) 제2 캐시(18)로부터 강등된다.

[0045] 기술된 동작들은 소프트웨어, 펌웨어, 하드웨어, 또는 이것들의 어떤 조합을 만들어 내기 위해 표준 프로그래밍 및/또는 엔지니어링 기술들을 사용하는 방법, 장치 또는 컴퓨터 프로그램 제품으로 구현될 수 있다. 따라서, 실시예들의 측면들은 전적으로 하드웨어 실시예의 형태를 취할 수도 있고, 전적으로 소프트웨어 실시예(펌웨어, 상주 소프트웨어(resident software), 마이크로코드(micro-code) 등을 포함함)의 형태를 취할 수도 있고, 또는 소프트웨어와 하드웨어 측면들을 결합하는 실시예(이들 모두는 본 명세서 내에서 일반적으로, "회로", "모듈" 또는 "시스템"으로 언급될 수 있음)의 형태를 취할 수도 있다. 더 나아가, 실시예들의 측면들은 그 내부에 구현된 컴퓨터 판독가능 프로그램 코드를 갖는 하나 또는 그 이상의 컴퓨터 판독가능 매체(들)에 구현된 컴퓨터 프로그램 제품의 형태를 취할 수도 있다.

[0046] 하나 또는 그 이상의 컴퓨터 판독가능 매체(들)의 어떤 조합이든지 이용될 수 있다. 컴퓨터 판독가능 매체는 컴퓨터 판독가능 신호 매체 또는 컴퓨터 판독가능 스토리지 매체일 수 있다. 컴퓨터 판독가능 스토리지 매체는, 예를 들어, 전자, 자기, 광학, 전자기, 적외선, 또는 반도체 시스템, 장치, 또는 디바이스, 또는 이것들의 어떤 적절한 조합이 사용될 수 있다. 그러나 이러한 것들로 한정되는 것은 아니다. 컴퓨터 판독가능 스토리지 매체의 더 구체적인 예들(모든 것들을 총 망라하는 것은 아님)은 하나 또는 그 이상의 와이어들을 갖는 전기적 연결(electrical connection), 휴대용 컴퓨터 디스켓, 하드 디스크, 랜덤 억세스 메모리(RAM, random access memory), 읽기 전용 메모리(ROM, read-only memory), 소거형 프로그래밍가능 읽기 전용 메모리(erasable programmable read-only memory, EPROM or Flash memory), 광 섬유, 휴대용 컴팩트 디스크 읽기 전용 메모리(CD-ROM), 광 스토리지 디바이스, 자기 스토리지 디바이스, 또는 이것들의 어떤 적절한 조합을 포함할 것이다. 본 문서의 맥락에서, 컴퓨터 판독가능 스토리지 매체는 명령 실행 시스템, 장치, 또는 디바이스에 의해 사용하기 위한 또는 명령 실행 시스템, 장치, 또는 디바이스와 함께 사용하기 위한 프로그램을 포함 또는 저장할 수 있는 어떤 실체적인 매체(tangible medium)일 수 있다.

[0047] 컴퓨터 판독가능 신호 매체는, 예를 들어, 기저대역으로 또는 반송파의 일부로서, 그 내부에 구현된 컴퓨터 판독가능 프로그램 코드를 갖는 전파되는 데이터 신호(propagated data signal)를 포함할 수 있다. 이러한 전파되는 신호는, 전자기, 광, 또는 이것들의 어떤 적절한 조합을 포함하는 어떤 다양한 형태들의 취할 수 있으나, 이러한 것들로 한정되는 것은 아니다. 컴퓨터 판독가능 신호 매체는, 컴퓨터 판독가능 스토리지 매체가 아니면서 명령 실행 시스템, 장치, 또는 디바이스에 의해 사용하기 위한 또는 명령 실행 시스템, 장치, 또는 디바이스와 함께 사용하기 위한 프로그램을 전달, 전파, 또는 이송할 수 있는 어떤 컴퓨터 판독가능 매체일 수 있다.

[0048] 컴퓨터 판독가능 매체 상에 구현된 프로그램 코드는, 무선, 유선, 광섬유 케이블, RF 등, 또는 이것들의 어떤 적절한 조합(그러나, 이러한 예들로 한정되는 것은 아님)을 포함하는 어떤 적절한 매체를 사용하여 전송될 수 있다.

[0049] 본 발명의 측면들을 위한 동작들을 수행하기 위한 컴퓨터 프로그램 코드는, 하나 또는 그 이상의 프로그래밍 언어들의 어떤 조합으로 작성될 수 있는데, 이러한 프로그래밍 언어들에는, 자바(Java), 스몰토크(Smalltalk), C++ 등과 같은 객체 지향형 프로그래밍 언어와, "C" 프로그래밍 언어 또는 유사 프로그래밍 언어들과 같은 전통적인 절차형 프로그래밍 언어들이 포함된다. 프로그램 코드는, 독립형 소프트웨어 패키지(stand-alone software package)와 같이, 사용자의 컴퓨터 상에서 전적으로 실행될 수도 있고, 부분적으로 사용자의 컴퓨터 상에서 실행될 수도 있으며, 부분적으로 사용자의 컴퓨터 상에서 그리고 부분적으로 원격 컴퓨터 상에서 실행될 수도 있으며, 또는 원격 컴퓨터 또는 서버 상에서 전적으로 실행될 수도 있다. 후자의 경우, 원격 컴퓨터는, LAN 또는 WAN을 포함하는 네트워크의 어떤 유형을 통해 사용자의 컴퓨터 상에 연결될 수 있고, 이 연결은 외부 컴퓨터(예를 들어, 인터넷 서비스 공급자를 사용하여 인터넷을 통해)에 대해 이뤄질 수 있다.

[0050] 본 발명의 측면들은 위에서 본 발명의 실시예들에 따른 방법들, 장치들(시스템들) 및 컴퓨터 프로그램 제품들의 흐름도들 및/또는 블록도들을 참조하여 기술되어 있다. 흐름도들 및/또는 블록도들의 각각의 블록, 및 흐름도들 및/또는 블록도들에서 블록들의 조합들은, 컴퓨터 프로그램 명령들에 의해 구현될 수 있다는 것을 이해하여야 할 것이다. 이들 컴퓨터 프로그램 명령들은, 범용 컴퓨터, 전용 컴퓨터, 또는 프로그램가능 데이터 처리 장치의 프로세서로 제공되어 하나의 머신을 생성하도록 하여, 컴퓨터 또는 다른 프로그램가능 데이터 처리 장치의 프로세서를 통해 실행될 때, 이러한 명령들이, 흐름도 및/또는 블록도의 블록 또는 블록들에 명시된 기능들/동작들을 구현하기 위한 수단을 생성하도록 한다.

[0051] 이들 컴퓨터 프로그램 명령들은 또한, 컴퓨터, 다른 프로그램가능 데이터 처리 장치, 또는 다른 디바이스들이

특정 방식으로 기능하도록 할 수 있는 컴퓨터 판독가능 매체에 저장될 수 있으며, 그리하여, 컴퓨터 판독가능 매체에 저장된 명령들은, 흐름도 및/또는 블록도의 블록 또는 블록들에 명시된 기능/동작을 구현하는 명령들을 포함하는 제조 물품(article of manufacture)을 생성하도록 한다.

[0052] 컴퓨터 프로그램 명령들은 또한, 컴퓨터, 다른 프로그램가능 데이터 처리 장치, 또는 다른 디바이스들 상에 로드되어, 컴퓨터, 다른 프로그램가능 장치 또는 다른 디바이스들 상에 일련의 동작 단계들이 수행되도록 하여, 컴퓨터로 구현되는 프로세스를 생성할 수 있다. 그리하여, 컴퓨터 또는 다른 프로그램가능 장치 상에서 실행되는 명령들이 흐름도 및/또는 블록도의 블록 또는 블록들에 명시된 기능들/동작들을 구현하기 위한 프로세서들을 제공하도록 한다.

[0053] "일 실시예", "실시예", "실시예들", "상기 실시예", "상기 실시예들", "하나 또는 그 이상의 실시예들", "몇몇 실시예들", 및 "하나의 실시예" 등의 용어들은, 특별히 그렇지 않은 것으로 표현되지 않는다면, "본 발명(들)의 하나 또는 그 이상의(모두는 아님) 실시예들"을 의미한다.

[0054] 본 명세서 내에서 "포함하는", "갖는" 및 이것들의 변형된 용어들과 같은 용어들은, 특별히 그렇지 않은 것으로 표현되지 않는다면, "포함하지만, 이러한 것들로 한정되는 것은 아님" 이라는 것을 의미한다.

[0055] 항목들의 열거된 목록은, 특별히 그렇지 않은 것으로 표현되지 않는다면, 그 항목들 중 어떤 것 또는 모두가 서로 배타적이라는 것을 의미하지는 않는다.

[0056] 본 명세서 내에서, "일", "하나" 및 "상기"라는 용어는, 특별히 그렇지 않은 것으로 표현되지 않는다면, "하나 또는 그 이상" 을 의미한다.

[0057] 서로 간에 통신하는 디바이스들은, 특별히 그렇지 않은 것으로 표현되지 않는다면, 서로 연속적으로 통신할 필요는 없다. 또한, 서로 통신하는 디바이스들은 직접적으로 통신할 수도 있고, 하나 또는 그 이상의 중간 매체들을 통해 간접적으로 통신할 수 있다.

[0058] 서로 간에 통신하는 몇몇 컴포넌트들을 갖는 실시예에 관한 설명은 이러한 모든 컴포넌트들이 요구된다는 것을 의미하지는 않는다. 그보다는 오히려, 본 발명의 폭넓은 가능한 실시예들을 설명하기 위해서, 다양한 선택적인 컴포넌트들이 기술된다.

[0059] 나아가, 비록 프로세스 단계들, 방법 단계들, 알고리즘들 등이, 순차적인 순서로 기술될 수 있으나, 이러한 프로세스들, 방법들 및 알고리즘들은 격순으로(in alternate orders) 구성될 수 있다. 바꿔 말하면, 단계들의 시퀀스 또는 순서는, 반드시 그 순서에서 단계들이 수행되는 요건을 나타내는 것은 아니다. 본 명세서 내에 기술되는 프로세스들의 단계들은 어떤 실제적인 순서로 수행될 수 있다. 나아가, 어떤 단계들은 동시에 수행될 수도 있다.

[0060] 본 명세서에서 하나의 디바이스 또는 물품이 기술될 때, 그것은 하나 이상의 디바이스/물품(그것들이 협력을 하든 그렇지 않든)은 하나의 디바이스/물품을 대신하여 사용될 수 있다는 것은 자명할 것이다. 이와 유사하게, 본 명세서에서 하나 이상의 디바이스 또는 물품이 기술되는 경우(그것들이 협력을 하든 그렇지 않든), 하나의 디바이스/물품이 하나 이상의 디바이스 또는 물품을 대신하여 사용될 수 있고 또는 다른 개수의 디바이스들/물품들이, 제시된 개수의 디바이스들 또는 프로그램들을 대신하여 사용될 수 있다는 것도 자명할 것이다. 또 다르게는, 디바이스의 기능(functionality) 및/또는 특징들은 이러한 기능/특징들을 갖는 것으로서 분명하게 기술되지 않은 하나 또는 그 이상의 다른 디바이스들에 의해 구현될 수 있다. 따라서, 본 발명의 다른 실시예들은 디바이스 자체를 포함할 필요가 없다.

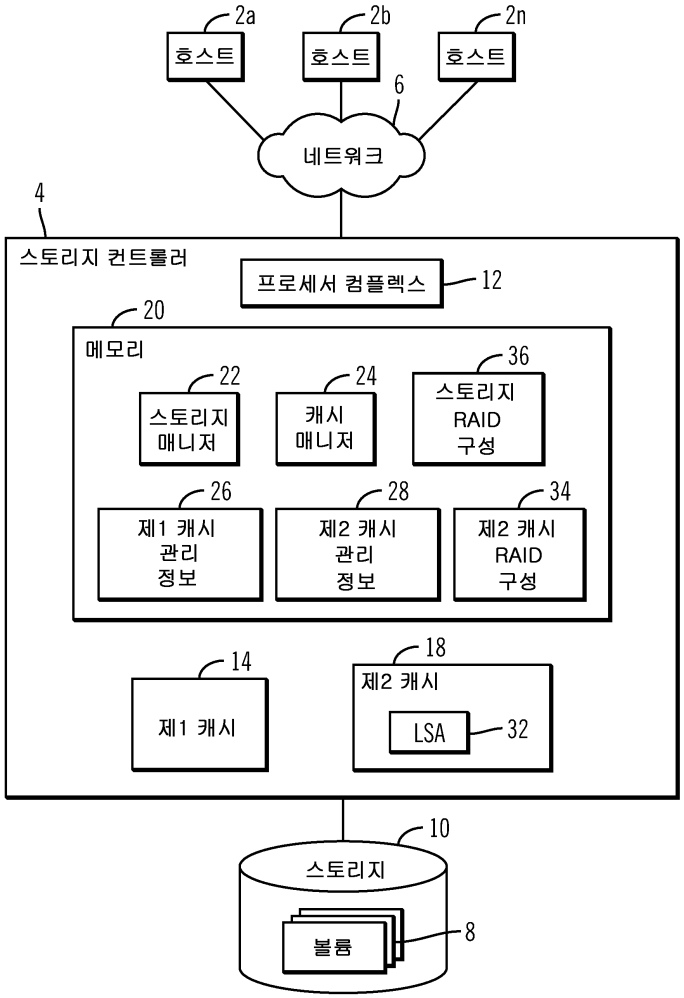
[0061] 도면들의 예시된 동작들은 특정 순서로 일어나는 특정 이벤트들을 보여준다. 또 다른 실시예들에서, 특정 동작들은 다른 순서로 수행될 수도 있고, 수정될 수도 있으며, 또는 제거될 수도 있다. 더욱이, 위에서 기술된 로직에 추가하여 단계들이 추가될 수 있으며, 이는 기술된 실시예들을 따른다. 나아가, 본 명세서에 기술된 동작들은 순차적으로 일어날 수도 있고, 또는 특정 동작들은 병렬로 처리될 수도 있다. 더 나아가, 동작들은 하나의 처리장치에 의해 수행될 수도 있고, 또는 분산형 처리 장치(distributed processing units)에 의해 수행될 수도 있다.

[0062] 본 발명의 여러 가지 실시예들에 관한 앞서의 설명은 예시 및 설명의 목적으로 제공되는 것이다. 따라서, 본 발명을 정확히 개시된 그 형태로 한정하려는 의도가 아니고, 또한 모든 실시예들을 빠짐없이 총 망라하려는 의도도 아니다. 앞서의 가르침에 비추어 많은 변형 및 변경 예들이 가능하다. 본 발명의 범위는 이 상세한 설명에 의해 한정되는 것이 아니라, 후속되는 청구항들에 의해 한정되도록 의도된다. 위의 설명, 예들 및 데이터는 제

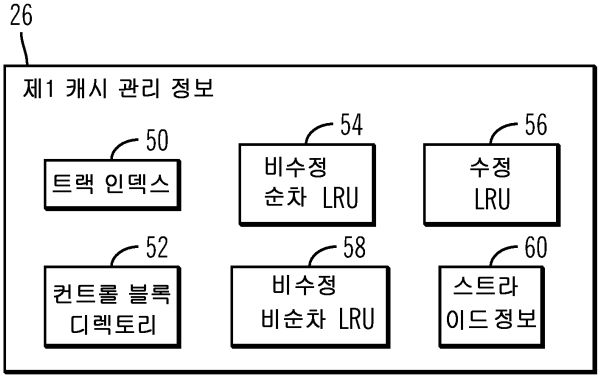
조에 관한 완전한 설명 및 발명의 구성의 사용을 제공한다. 발명의 사상 및 범위를 벗어남이 없이 발명의 많은 실시예들이 만들어질 수 있으므로, 발명은 후속되는 청구항들에 있다.

도면

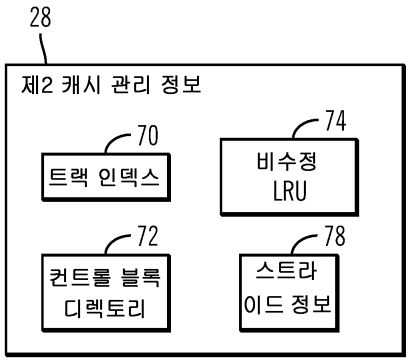
도면1



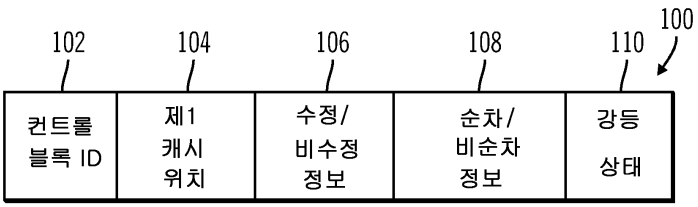
도면2



도면3

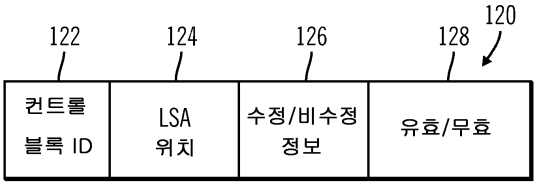


도면4



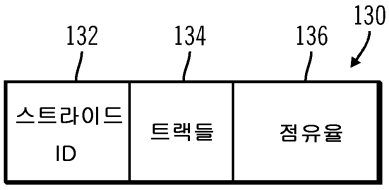
제1 캐시 컨트롤 블록

도면5



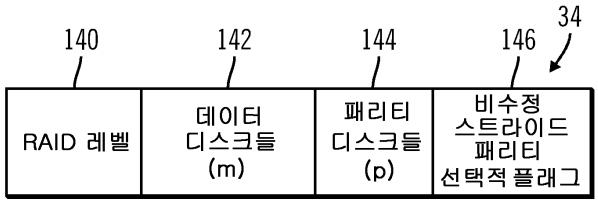
제2 캐시 컨트롤 블록

도면6



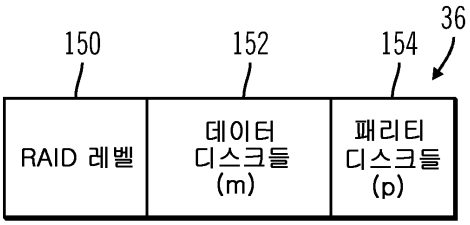
스트라이드 정보

도면7



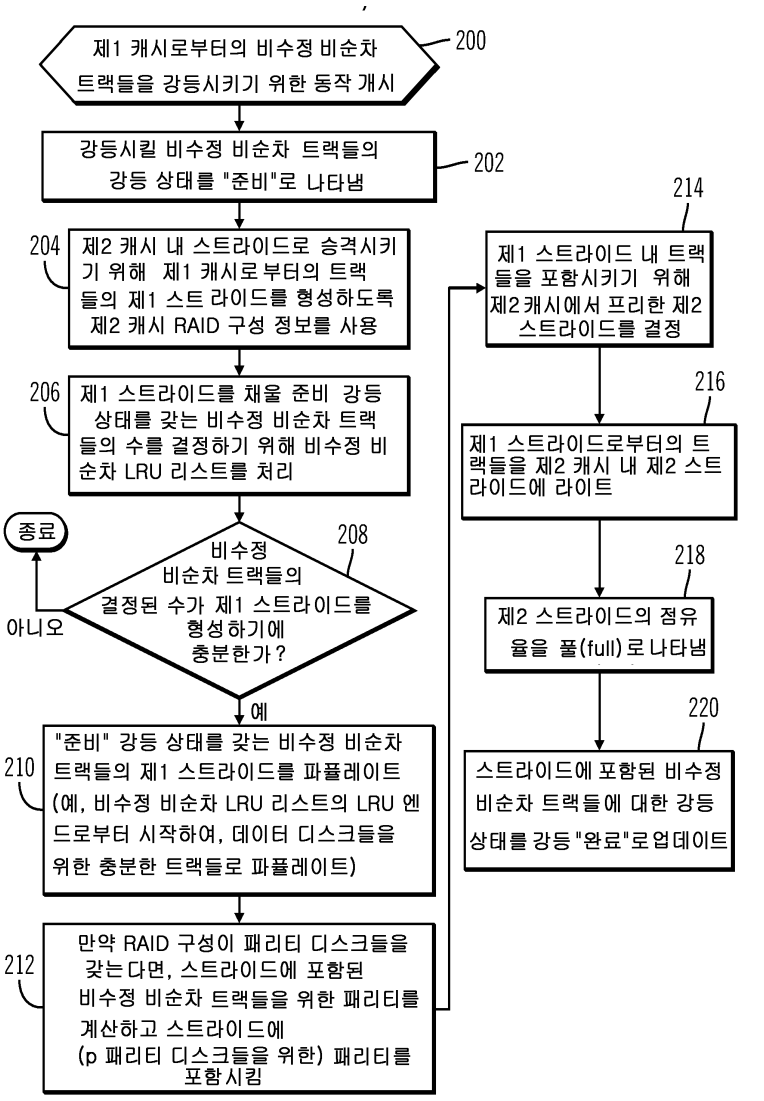
제2 캐시 RAID 구성

도면8

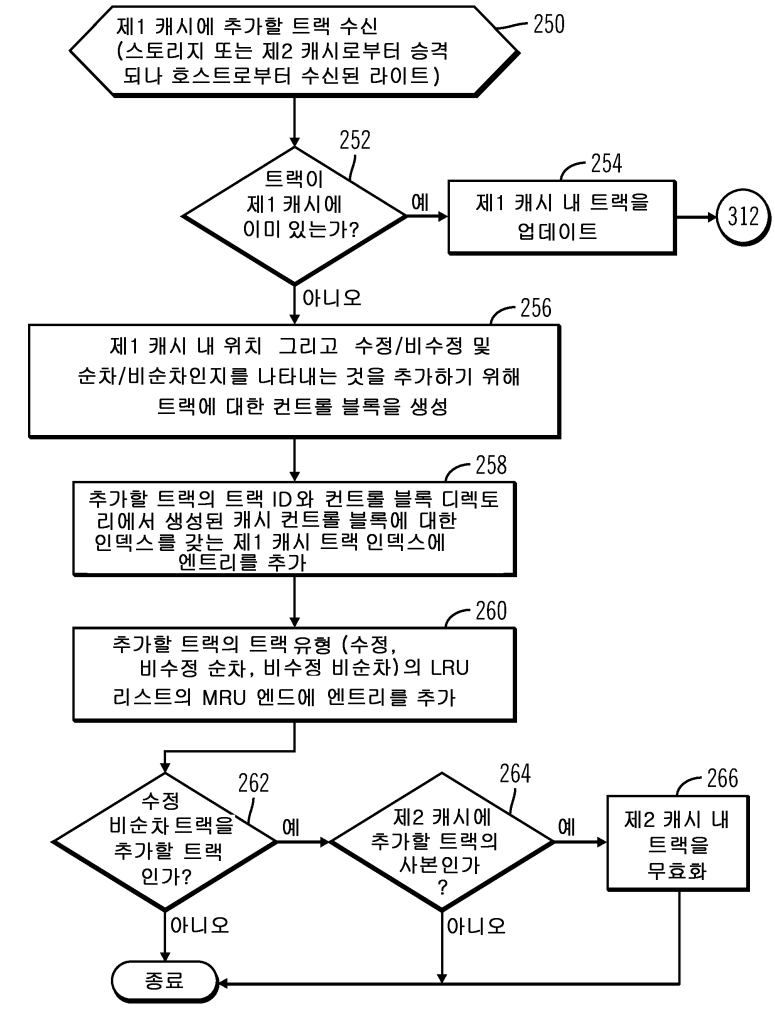


스토리지 시스템 RAID 구성

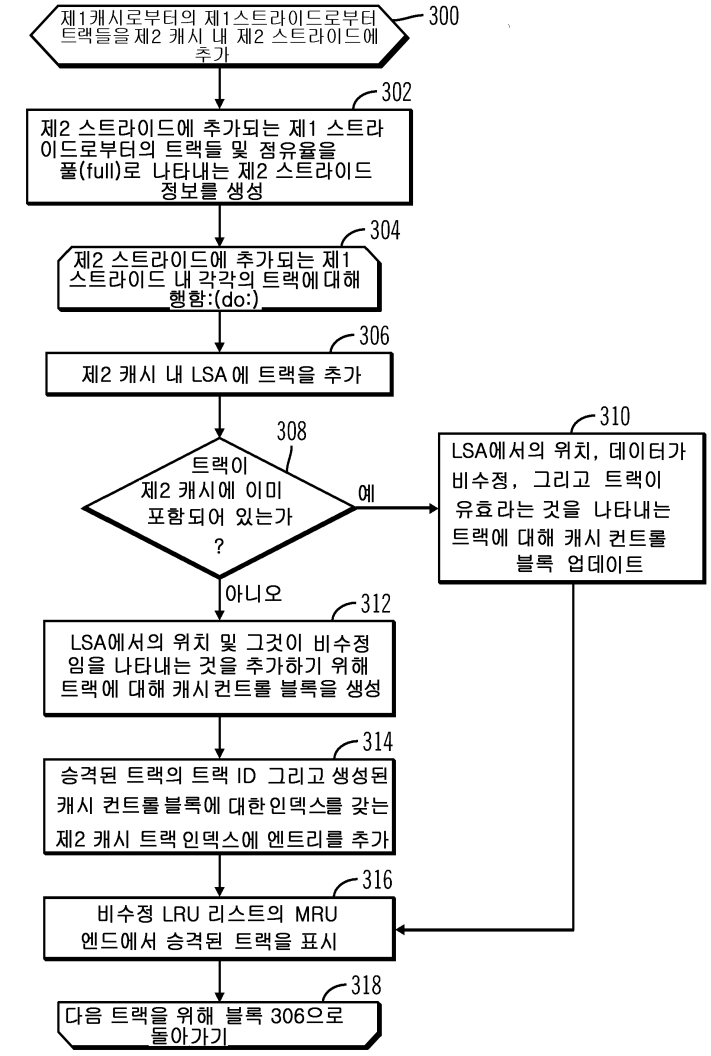
도면9


제1 캐시로부터의 비수정 비순차 트랙들을 강등시키기 위한 동작 개시
200
강등시킬 비수정 비순차 트랙들의 강등 상태를 "준비"로 나타냄
202
204
제2 캐시 내 스트라이드로 승격시키기 위해 제1 캐시로부터의 트랙들의 제1 스트 라이드를 형성하도록 제2 캐시 RAID 구성 정보를 사용
206
제1 스트라이드를 채울 준비 강등 상태를 갖는 비수정 비순차 트랙들의 수를 결정하기 위해 비수정 비순차 LRU 리스트를 처리
종료
208
비수정 비순차 트랙들의 결정된 수가 제1 스트라이드를 형성하기에 충분한가?
아니오
예
210
"준비" 강등 상태를 갖는 비수정 비순차 트랙들의 제1 스트라이드를 파퓰레이트 (예, 비수정 비순차 LRU 리스트의 LRU 엔드로부터 시작하여, 데이터 디스크들을 위한 충분한 트랙들로 파퓰레이트)
212
만약 RAID 구성이 패리티 디스크들을 갖는다면, 스트라이드에 포함된 비수정 비순차 트랙들을 위한 패리티를 계산하고 스트라이드에 (p 패리티 디스크들을 위한) 패리티를 포함시킴
214
제1 스트라이드 내 트랙들을 포함시키기 위해 제2캐시에서 프리한 제2 스트라이드를 결정
216
제1 스트라이드로부터의 트랙들을 제2 캐시 내 제2 스트라이드에 라이트
218
제2 스트라이드의 점유율을 풀(full)로 나타냄
220
스트라이드에 포함된 비수정 비순차 트랙들에 대한 강등 상태를 강등"완료"로 업데이트

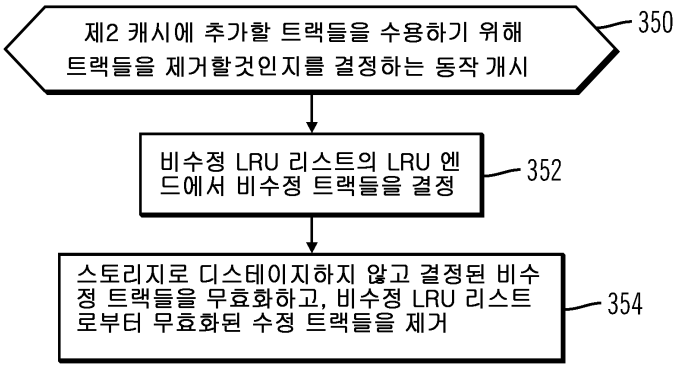
도면10


제1 캐시에 추가할 트랙 수신
(스토리지 또는 제2 캐시로부터 승격
되나 호스트로부터 수신된 라이트)
250
252
트랙이
제1 캐시에
이미 있는가?
예
254
제1 캐시 내 트랙을
업데이트
312
아니오
256
제1 캐시 내 위치 그리고 수정/비수정 및
순차/비순차인지를 나타내는 것을 추가하기 위해
트랙에 대한 컨트롤 블록을 생성
258
추가할 트랙의 트랙 ID와 컨트롤 블록 디렉토
리에서 생성된 캐시 컨트롤 블록에 대한
인덱스를 갖는 제1 캐시 트랙 인덱스에
엔트리를 추가
260
추가할 트랙의 트랙 유형 (수정,
비수정 순차, 비수정 비순차)의 LRU
리스트의 MRU 엔드에 엔트리를 추가
262
수정
비순차 트랙을
추가할 트랙
인가?
예
264
제2 캐시에
추가할 트랙의
사본인가
?
예
266
제2 캐시 내
트랙을
무효화
아니오
아니오
종료

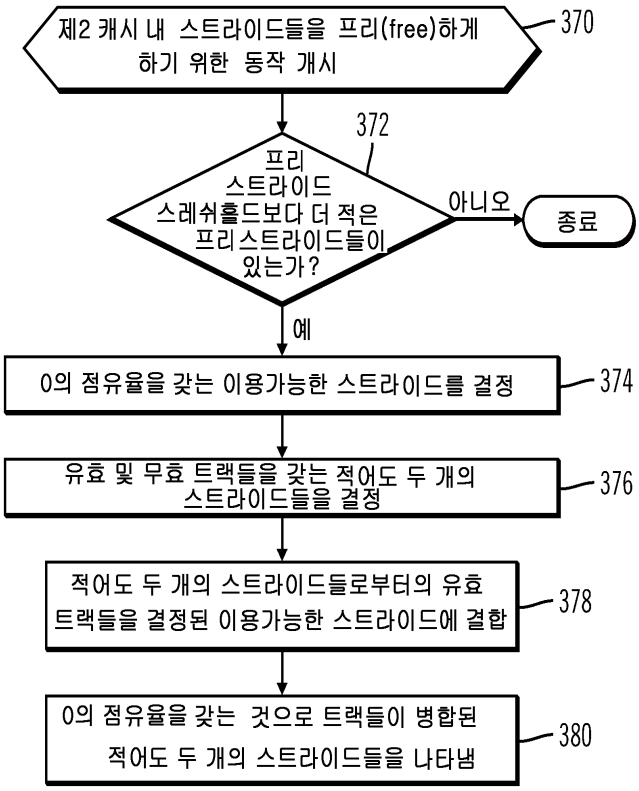
도면11


제1캐시로부터의 제1스트라이드로부터 트랙들을 제2 캐시 내 제2 스트라이드에 추가
300
제2 스트라이드에 추가되는 제1 스트라이드로부터의 트랙들 및 점유율을 풀(full)로 나타내는 제2 스트라이드 정보를 생성
302
제2 스트라이드에 추가되는 제1 스트라이드 내 각각의 트랙에 대해 행함:(do:)
304
제2 캐시 내 LSA에 트랙을 추가
306
트랙이 제2 캐시에 이미 포함되어 있는가 ?
308
예
LSA에서의 위치, 데이터가 비수정, 그리고 트랙이 유효라는 것을 나타내는 트랙에 대해 캐시 컨트롤 블록 업데이트
310
아니오
LSA에서의 위치 및 그것이 비수정임을 나타내는 것을 추가하기 위해 트랙에 대해 캐시컨트롤 블록을 생성
312
승격된 트랙의 트랙 ID 그리고 생성된 캐시 컨트롤블록에 대한인덱스를 갖는 제2 캐시 트랙 인덱스에 엔트리를 추가
314
비수정 LRU 리스트의 MRU 엔드에서 승격된 트랙을 표시
316
다음 트랙을 위해 블록 306으로 돌아가기
318

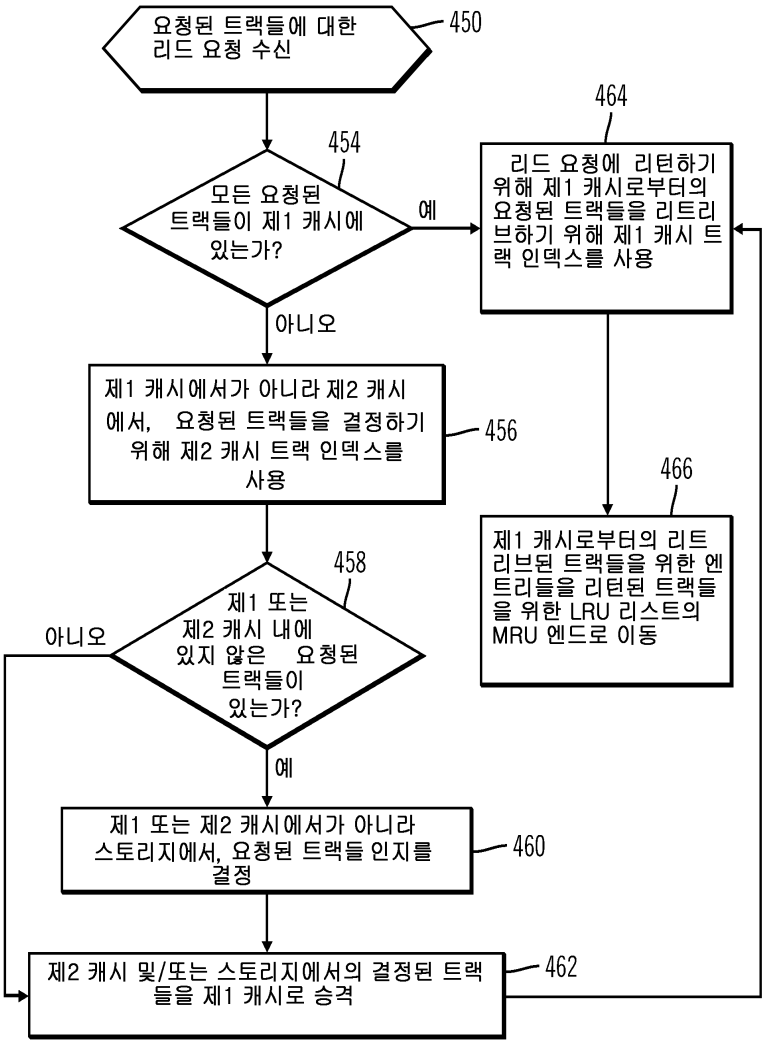
도면12


제2 캐시에 추가할 트랙들을 수용하기 위해 트랙들을 제거할것인지를 결정하는 동작 개시
350
비수정 LRU 리스트의 LRU 엔드에서 비수정 트랙들을 결정
352
스토리지로 디스테이지하지 않고 결정된 비수정 트랙들을 무효화하고, 비수정 LRU 리스트로부터 무효화된 수정 트랙들을 제거
354

도면13


제2 캐시 내 스트라이드들을 프리(free)하게 하기 위한 동작 개시
370
372
프리 스트라이드 스레쉬홀드보다 더 적은 프리스트라이드들이 있는가?
아니오
종료
예
0의 점유율을 갖는 이용가능한 스트라이드를 결정
374
유효 및 무효 트랙들을 갖는 적어도 두 개의 스트라이드들을 결정
376
적어도 두 개의 스트라이드들로부터의 유효 트랙들을 결정된 이용가능한 스트라이드에 결합
378
0의 점유율을 갖는 것으로 트랙들이 병합된 적어도 두 개의 스트라이드들을 나타냄
380

도면14


요청된 트랙들에 대한 리드 요청 수신
450
모든 요청된 트랙들이 제1 캐시에 있는가?
454
예
아니오
리드 요청에 리턴하기 위해 제1 캐시로부터의 요청된 트랙들을 리트리브하기 위해 제1 캐시 트랙 인덱스를 사용
464
제1 캐시에서가 아니라 제2 캐시에서, 요청된 트랙들을 결정하기 위해 제2 캐시 트랙 인덱스를 사용
456
제1 캐시로부터의 리트리브된 트랙들을 위한 엔트리들을 리턴된 트랙들을 위한 LRU 리스트의 MRU 엔드로 이동
466
제1 또는 제2 캐시 내에 있지 않은 요청된 트랙들이 있는가?
458
아니오
예
제1 또는 제2 캐시에서가 아니라 스토리지에서, 요청된 트랙들 인지를 결정
460
제2 캐시 및/또는 스토리지에서의 결정된 트랙들을 제1 캐시로 승격
462