



(19) 대한민국특허청(KR)

(12) 등록특허공보(B1)

(45) 공고일자 2016년03월18일

(11) 등록번호 10-1604610

(24) 등록일자 2016년03월14일

(51) 국제특허분류(Int. Cl.)  
 G06F 12/08 (2016.01) G06F 12/10 (2016.01)  
 G06F 9/06 (2006.01)  
 (21) 출원번호 10-2014-0023435  
 (22) 출원일자 2014년02월27일  
 심사청구일자 2014년02월27일  
 (65) 공개번호 10-2014-0112397  
 (43) 공개일자 2014년09월23일  
 (30) 우선권주장  
 13/801,490 2013년03월13일 미국(US)  
 (56) 선행기술조사문헌  
 US20120297140 A1

(73) 특허권자  
 시게이트 테크놀로지 엘엘씨  
 미국 캘리포니아 95104 쿠퍼티노 사우스 디 엔자  
 블러바드 10200  
 (72) 발명자  
 코우레스, 앤드류 마이클  
 미국 80308 콜로라도 볼더 피오비 17103  
 (74) 대리인  
 특허법인 남앤드남

전체 청구항 수 : 총 17 항

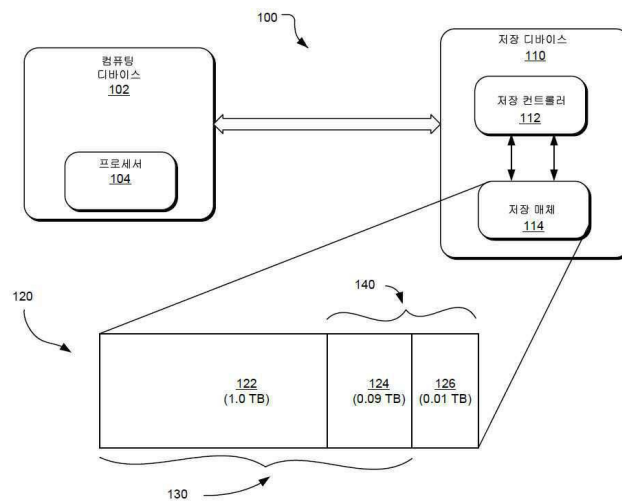
심사관 : 김대성

(54) 발명의 명칭 동적 저장 디바이스 프로비저닝

(57) 요약

방법 또는 시스템은 저장 매체의 저장 공간을 영구적으로 할당된 매체 캐시 저장 영역; 동적으로 매핑된 매체 캐시 저장 영역, 및 정적으로 매핑된 저장 영역에 할당하며, 상기 동적으로 매핑된 매체 캐시 저장 영역은 성능 및/또는 신뢰성 강화 기능들에 사용된다.

대표도 - 도1



**명세서**

**청구범위**

**청구항 1**

저장 디바이스로서,  
저장 매체; 및

상기 저장 매체의 저장 공간을 (i) 영구적으로 할당된 매체 캐시 저장 영역; (ii) 동적으로 할당된 매체 캐시 저장 영역; 및 (iii) 정적으로 할당된 저장 영역에 할당하도록 구성되는 프로세서를 포함하고,

상기 프로세서는 상기 동적으로 할당된 매체 캐시 저장 영역의 매핑된 부분의 크기 대 상기 동적으로 할당된 매체 캐시 저장 영역의 매핑되지 않은 부분의 크기의 비율을 동적으로 변경하도록 구성되는, 저장 디바이스.

**청구항 2**

청구항 1에 있어서, 상기 프로세서는 상기 동적으로 할당된 매체 캐시 저장 영역의 크기 및 상기 정적으로 할당된 저장 영역의 크기의 합과 같은 사용가능 저장 용량을 보고하도록 더 구성되는, 저장 디바이스.

**청구항 3**

청구항 1에 있어서, 상기 프로세서는 성능 강화 기능들에 대해 상기 영구적으로 할당된 매체 캐시 저장 영역 및 상기 동적으로 할당된 매체 캐시 저장 영역 각각을 사용하도록 더 구성되는, 저장 디바이스.

**청구항 4**

청구항 1에 있어서, 상기 프로세서는,

논리 블록 어드레스(LBA) 맵의 상이한 부분의 사용 패턴들을 결정하고;

상기 LBA 맵의 다른 부분들과 비교하여 덜 빈번히 사용되는 상기 LBA 맵의 제 1 부분을 상기 동적으로 할당된 매체 캐시 저장 영역에 매핑하도록 더 구성되는, 저장 디바이스.

**청구항 5**

청구항 1에 있어서, 상기 프로세서는,

상기 정적으로 할당된 저장 영역들에 피닝되는 기록 요청 데이터와 상기 동적으로 할당된 매체 캐시 저장 영역에 피닝되는 기록 요청 데이터를 분리하고; 그리고

상기 동적으로 할당된 매체 캐시 저장 영역에 피닝되는 기록 요청 데이터를 상기 동적으로 할당된 매체 캐시 저장 영역에 동적으로 매핑하도록 더 구성되는, 저장 디바이스.

**청구항 6**

삭제

**청구항 7**

청구항 1에 있어서, 상기 프로세서는 상기 동적으로 할당된 매체 캐시 저장 영역의 매핑된 부분의 크기 대 상기 동적으로 할당된 매체 캐시 저장 영역의 매핑되지 않은 부분의 크기의 비율에 기초하여 성능 강화 기능을 변경하도록 더 구성되는, 저장 디바이스.

**청구항 8**

저장 디바이스의 동적 오버프로비저닝(overprovisioning)에 대한 방법으로서,

비휘발성 저장 매체의 저장 공간을 (i) 영구적으로 할당된 매체 캐시 저장 영역; (ii) 동적으로 할당된 매체 캐시 저장 영역; 및 (iii) 정적으로 할당된 저장 영역에 할당하는 단계 - 상기 동적으로 할당된 매체 캐시 저장

영역은 성능 강화 기능들에 사용됨 - ; 및

상기 동적으로 할당된 매체 캐시 저장 영역의 매핑된 부분의 크기 대 상기 동적으로 할당된 매체 캐시 저장 영역의 매핑되지 않은 부분의 크기의 비율에 기초하여 성능 강화 기능을 변경하는 단계를 포함하는, 저장 디바이스의 동적 오버프로비저닝에 대한 방법.

**청구항 9**

청구항 8에 있어서, 상기 비휘발성 저장 매체는 (1) 디스크 드라이브 저장 매체, 및 (2) 하이브리드 디스크 드라이브 저장 매체 중 적어도 하나인, 저장 디바이스의 동적 오버프로비저닝에 대한 방법.

**청구항 10**

청구항 8에 있어서, 논리 블록 어드레스들(LBA들)의 초기 부분을 상기 동적으로 할당된 매체 캐시 저장 영역으로부터 언매핑(unmapping)하는 단계 및 상기 LBA들의 하나 이상의 상이한 부분들을 상기 동적으로 할당된 매체 캐시 저장 영역에 매핑하는 단계를 더 포함하는, 저장 디바이스의 동적 오버프로비저닝에 대한 방법.

**청구항 11**

청구항 10에 있어서, 상기 LBA들의 초기 부분을 상기 동적으로 할당된 매체 캐시 저장 영역으로부터 언매핑하는 단계 및 상기 LBA들의 상이한 부분들을 상기 동적으로 할당된 매체 캐시 저장 영역에 매핑하는 단계는 상기 LBA들의 초기 부분의 서브세트를 방출하는 호스트로부터 트립 커맨드를 수신하는 것에 대응하여 수행되는, 저장 디바이스의 동적 오버프로비저닝에 대한 방법.

**청구항 12**

청구항 8에 있어서,

LBA 맵의 복수의 부분들의 사용 패턴들을 결정하는 단계; 및

LBA들의 제 1 부분을 상기 동적으로 할당된 매체 캐시 저장 영역으로부터 언매핑하는 단계 및 상기 LBA들의 제 2 부분을 상기 동적으로 할당된 매체 캐시 저장 영역에 매핑하는 단계를 더 포함하며,

상기 LBA들의 제 1 부분의 적어도 일부는 상기 LBA들의 제 2 부분의 적어도 일부보다 더 빈번히 사용되도록 결정되는, 저장 디바이스의 동적 오버프로비저닝에 대한 방법.

**청구항 13**

청구항 8에 있어서,

기록 요청을 수신하는 단계;

상기 정적으로 할당된 저장 영역들에 피닝되는 기록 요청 데이터와 상기 동적으로 할당된 매체 캐시 저장 영역에 피닝되는 기록 요청 데이터를 분리하는 단계; 및

상기 동적으로 할당된 매체 캐시 저장 영역에 피닝되는 기록 요청 데이터를 상기 동적으로 할당된 매체 캐시 저장 영역에 동적으로 매핑하는 단계를 더 포함하는, 저장 디바이스의 동적 오버프로비저닝에 대한 방법.

**청구항 14**

청구항 13에 있어서, 정적으로 할당된 저장 영역들에 피닝되는 기록 요청 데이터를 상기 정적으로 할당된 저장 영역들에 정적으로 매핑하는 단계를 더 포함하는, 저장 디바이스의 동적 오버프로비저닝에 대한 방법.

**청구항 15**

삭제

**청구항 16**

청구항 8에 있어서,

상기 동적으로 할당된 매체 캐시 저장 영역의 매핑된 부분의 크기 대 상기 동적으로 할당된 매체 캐시 저장 영

역의 매핑되지 않은 부분의 크기의 비율을 비교하는 단계; 및

상기 비율이 임계값보다 위이면, 성능 강화 기능들에 대해 상기 동적으로 할당된 매체 캐시 저장 영역의 사용을 종결하는 단계를 더 포함하는, 저장 디바이스의 동적 오버프로비저닝에 대한 방법.

**청구항 17**

저장 시스템으로서,

저장 매체;

상기 저장 매체의 예상된 사용 패턴을 결정하고 상기 예상된 사용 패턴에 기초하여 오버프로비저닝의 크기를 결정하도록 구성되는 프로세서를 포함하며,

상기 프로세서는 상기 저장 매체의 저장 공간을, 상기 예상된 사용 패턴에 기초하여, (i) 영구적으로 할당된 매체 캐시 저장 영역; (ii) 동적으로 할당된 매체 캐시 저장 영역; 및 (iii) 정적으로 할당된 저장 영역에 할당하도록 구성되며,

상기 프로세서는 상기 동적으로 할당된 매체 캐시 저장 영역의 매핑된 부분의 크기 대 상기 동적으로 할당된 매체 캐시 저장 영역의 매핑되지 않은 부분의 크기의 비율을 동적으로 변경하도록 구성되는,

저장 시스템.

**청구항 18**

삭제

**청구항 19**

청구항 17에 있어서, 상기 프로세서는 상기 동적으로 할당된 매체 캐시 저장 영역의 크기 및 상기 정적으로 할당된 저장 영역의 크기의 합과 같은 사용가능 저장 용량을 보고하도록 더 구성되는, 저장 시스템.

**청구항 20**

청구항 17에 있어서, 상기 프로세서는 성능 강화 기능들에 대해 상기 영구적으로 할당된 매체 캐시 저장 영역 및 상기 동적으로 할당된 매체 캐시 저장 영역 각각을 사용하도록 더 구성되는, 저장 시스템.

**발명의 설명**

**배경 기술**

[0001]

데이터 저장, 검색, 및 분석에 대한 요구들이 증가했으므로, 더 싸고 더 빠른 데이터 저장 디바이스들에 대한 요구가 또한 증가했다. 디스크 드라이브들, 고체 상태 드라이브들 등과 같은 데이터 저장 디바이스들은 자기 기록, 광 기록 등과 같은 상이한 저장 기술들을 사용하여 상이한 기록 매체 상에 데이터를 저장한다. 디스크 드라이브들은 컴퓨팅 시스템들 내의 데이터 저장 디바이스로 광범위하게 사용된다. 증가하는 요구들을 충족시키기 위해, 디스크 드라이브들 제공자들은 디스크 드라이브들의 용량 뿐만 아니라 디스크 드라이브들 상의 기록의 밀도를 증가시켰다. 동시에, 대다수의 데이터가 사용자들에 의해 저장 및 처리되는 것을 고려하면, 저장 디바이스 제공자들은 저장 용량을 더싼 가격으로 제공하라는 압력을 받고 있다.

**발명의 내용**

**해결하려는 과제**

[0002]

본 요약은 이하 상세한 설명에 더 설명되는 개념들의 선택을 간략한 형태로 소개하기 위해 제공된다. 본 요약은 청구된 발명 대상의 중요 특징들 또는 본질적 특징들을 식별하도록 의도되지 않으며, 청구된 발명 대상의 범위를 제한하기 위해 사용되도록 의도되지 않는다. 청구된 발명 대상의 다른 특징들, 상세들, 유용성들, 및 장점들은 다양한 구현들 및 첨부 도면들에 더 예시되고 첨부된 청구항들에 정의된 바와 같은 구현들의 이하의 보다 상세히 기록된 상세한 설명으로부터 분명할 것이다.

**과제의 해결 수단**

[0003] 방법 또는 시스템은 저장 매체의 저장 공간을 영구적으로 할당된 "매체 캐시" 저장 영역; 동적으로 매핑된 매체 캐시 저장 영역; 및 정적으로 매핑된 저장 영역에 할당하며, 동적으로 매핑된 매체 캐시 저장 영역은 성능 강화 기능들에 사용된다.

[0004] 이들 및 다양한 다른 특징들 및 장점들은 이하의 상세한 설명의 판독으로부터 분명할 것이다.

**도면의 간단한 설명**

[0005] 설명된 기술은 첨부 도면들과 관련하여 판독되는 다양한 구현들을 설명하는 이하의 상세한 설명으로부터 가장 잘 이해된다.

도 1은 매체의 동적 오버프로비저닝을 위한 시스템의 구현의 블록도를 예시한다.

도 2는 매체의 동적 오버프로비저닝을 위한 시스템의 대체 구현의 블록도를 예시한다.

도 3은 동적 오버프로비저닝 of 매체의 을 위한 시스템의 대체 구현의 블록도를 예시한다.

도 4는 매체의 동적 오버프로비저닝을 데이터 저장 시스템에 제공하는 동작들의 예를 예시한다.

도 5는 매체의 동적 오버프로비저닝을 데이터 저장 시스템에 제공하는 동작들의 다른 예를 예시한다.

도 6은 성능 강화 기능들에 사용되는 동적으로 매핑된 저장 영역에 타겟화된 기록들의 효과를 나타내는 하나의 예시적 그래피를 예시한다.

**발명을 실시하기 위한 구체적인 내용**

[0006] 이하의 설명에서, 설명의 목적들을 위해, 다수의 특정 상세들은 본 명세서에 설명되는 다양한 구현들의 완전한 이해를 제공하기 위해 진술된다. 다양한 특징들이 특정 구현들에 속하는 것으로 생각되지만, 일 구현에 관하여 설명되는 특징들은 다른 구현들과 통합될 수 있다는 점이 이해되어야 한다. 그러나, 동일한 표시에 의해, 임의의 설명된 구현의 어떤 단일 특징 또는 특징들도 다른 구현들이 그러한 특징들을 생략할 수 있으므로, 필수적인 것으로 간주되지 않아야 한다.

[0007] 데이터 저장, 검색, 및 분석에 대한 요구들이 증가했으므로, 더 싸고 더 빠른 데이터 저장 디바이스들에 대한 요구가 또한 증가해왔다. 디스크 드라이브들은 데이터 저장 디바이스 컴퓨팅 시스템들로서 광범위하게 사용된다. 그러한 데이터 저장 디바이스들은 자기 매체, 광 매체 등 상에 데이터를 저장하는 디스크 드라이브들을 포함한다. 증가하는 요구를 충족시키기 위해, 데이터 저장 제공자들은 저장 디바이스들의 크기 뿐만 아니라 저장 디바이스들의 밀도를 증가시켰다. 실질적으로 저장 제공자들이 그들의 고객들에게 제공하는 저장 용량의 모든 바이트는 거기에 부여되는 비용을 갖는다. 예를 들어, 그러한 비용은 테라바이트 당 달러 등의 면에서 지정될 수 있다. 일반적으로, 디스크 드라이브 제공자들 및 다른 저장 용량 제공자들에 대해, 더 많은 저장 용량을 제조하는 것은 더 많이 비용이 든다. 그러나, 저장 산업의 경쟁 본질을 고려하면, 저장 제공자들이 그들의 고객들로부터 추출할 수 있는 가격은 시장의 힘에 의해 제한된다. 그러므로, 저장 제공자들은 그들의 이윤 폭을 증가시키기 위해 및/또는 경쟁력을 유지하기 위해 그들의 고객들에게 판매될수록 저장의 비용을 감소시킬 동기를 갖는다.

[0008] 저장 용량 당 저장 제공자들의 비용을 증가시키거나 고객들에게 판매되는 저장 용량 당 비용을 수익을 감소시키는 요인들 중 하나는 저장 제공자들이 그들의 고객들에게 판매되는 또는 보고되는 것보다 더 많은 물리 저장 용량을 제공하는 저장 용량의 오버프로비저닝이다. 이것의 일 예는 2개의 상이한 제조업자로부터 둘 다가 동일한 용량을 갖고 거래되는, 2개의 상이한 디바이스를 구매할 때 인식될 수 있다: 디바이스들이 근소하게 상이한 용량들을 보고한다는 점이 관측될 수 있다. 이 맥락에서, 오버프로비저닝은 고객들에게 또는 시장에 판매되거나 거래되는 저장 용량보다 더 많은 실제 저장 용량을 제조하고 출하하는 것을 포함한다. 따라서, 예를 들어 1 TB의 저장 용량이 고객에게 판매되면, 저장 제공자는 실제로 10 퍼센트 더 많은 용량을 제조하며, 따라서 1.1 TB의 저장 용량이 고객에게 제공된다. 그러한 경우에, 저장 제공자는 실질적으로 1 TB만의 수익을 얻을 수 있지만, 그것은 1.1 TB의 저장 용량을 제조하고 제공하기 위한 비용을 발생시킨다.

[0009] 서비스 제공자들이 오버프로비저닝을 사용하는 다수의 이유들이 있다. 예를 들어, NAND 드라이브들의 형태로의 저장을 위해, 오버프로비저닝은 저장 매체의 궁극적인 웨어링(NAND 플래시가 마모되어 버리는) 후에도, 고객이 여전히 고객에게 판매되었던 약간의 최소 스토리지를 갖는다는 점을 보장한다. 디스크 드라이브들의 형태로의 스토리지에 대해, '오버프로비저닝'이라는 일반적 용어는 많은 이유들로 사용된다. 여기에 논의되는 오버프로비

저널은 디스크 드라이브들의 평균 액세스 시간 등과 같은 성능 메트릭스를 증대하기 위해 사용된다. 디스크 드라이브 스토리지 제공자들은 디스크 드라이브들의 성능을 더 증가시키기 위해 오버프로비저닝의 크기를 증가시킬 동기를 가진다. 그러나, 언제든지 오버프로비저닝의 양은 증가되며, 단위 저장 용량 당 비용이 또한 증가한다. 따라서, 1 TB의 저장 용량을 갖는 것으로 판매되는 디스크 드라이브에 대해, 오버프로비저닝이 10%에서 20%로 증가되면, 저장 용량의 TB 당 비용 또한 20%만큼 증가하는 반면, 제공자는 고객으로부터 그러한 비용을 회수 가능하지 않을 수 있다. 따라서, 오버프로비저닝의 비용과 오버프로비저닝에 의해 획득되는 성능 이득들 사이에서 오버프로비저닝의 증가를 제공하는 트레이드오프가 있다.

[0010]

본 명세서에 제공되는 구현은 오버프로비저닝에 사용되는 저장 공간의 적어도 일부를 동적으로 매핑한다. 예를 들어, 0.1 TB(100 GB)가 오버프로비저닝에 사용되는 1.1 TB(1100 GB)의 전체 저장 용량을 갖는 디스크 드라이브에 대해, 0.1 TB 오버프로비저닝된 저장 공간의 적어도 일부는 동적으로 매핑된다. 예를 들어, 1.1 TB의 전체 용량을 갖는 디스크 드라이브에 대해, 오버프로비저닝을 위해 0.1 TB의 저장 영역들을 제공하는 것 대신에, 실질적으로 0.01 TB(10 GB)의 저장 영역만이 오버프로비저닝을 위해 제공된다. 다른 1.9 TB(90 GB)의 저장 용량은 동적으로 매핑되고 그것은 성능 강화 기능들을 위해 사용된다. 그러한 구현에서, 저장 디바이스의 드물게 또는 조금 사용된 영역은 동적으로 매핑된다. 예를 들어, 오버프로비저닝을 위한 0.01 TB의 매핑되지 않은 영역 및 1.09 TB의 매핑된 저장 영역들이 주어지면, 0.09 TB의 매핑된 저장 영역은 동적으로 매핑되고 그러한 동적으로 매핑된 저장 영역은 오버프로비저닝된 저장 영역들을 사용하여 일반적으로 수행되는 성능 증대 기능성들을 위해 사용된다.

[0011]

도 1은 저장 매체의 동적 오버프로비저닝에 대한 시스템(100)을 개시한다. 구체적으로, 시스템(100)은 저장 디바이스(110)에 통신 연결된 프로세서(104)를 갖는 컴퓨팅 디바이스(102)를 포함한다. 시스템(100)이 컴퓨팅 디바이스(102) 외부에 저장 디바이스(110)를 갖도록 예시되지만, 대안적 구현에서, 저장 디바이스(110)는 컴퓨팅 디바이스(102) 내부에 있을 수 있다. 저장 디바이스(110)는 예를 들어 자기 디스크 드라이브, 광 디스크 드라이브, 플래시 드라이브 등일 수 있다. 저장 디바이스(110)는 저장 컨트롤러(112) 및 저장 매체(114)를 포함한다. 저장 컨트롤러(112)는 컴퓨팅 디바이스(102)로부터 저장 커맨드들을 수신하고 저장 커맨드들을 처리하여 저장 매체(114)로부터 데이터를 판독하고, 데이터를 저장 매체(114)에 기록하고, 판독/기록 헤드(필요하다면)의 이동을 제어하도록 구성되는 등이다.

[0012]

일 구현에서, 저장 매체(114)는 다수의 저장 영역들로 분할된다. 예를 들어, 저장 디바이스(110)가 자기 디스크 드라이브인 경우, 저장 매체(114)는 각각의 영역이 디스크 드라이브들 상의 하나 이상의 섹터들로 이루어진 다수의 영역들로 분할된다. 시스템(100)은 저장 영역(120)에 의한 저장 매체(114)의 전체 저장 용량을 예시한다. 시스템(100)에 예시된 바와 같이, 저장 영역(120)은 다수의 상이한 저장 영역들(122, 124, 126) 등을 포함한다. 일 구현에서, 저장 컨트롤러(112)는 저장 영역(120) 내에 지정되는 다양한 저장 영역들에 저장 매체(114) 상의 트랙들 및 섹터들의 내부 매핑을 유지함으로써 다양한 저장 영역들로 저장 매체(114)의 분할을 유지한다. 예를 들어, 디스크 드라이브 저장 디바이스에 대해, 디스크들 상의 특정 섹터들은 저장 영역(122)에 할당될 수 있는 반면, 디스크들의 다른 섹터들은 저장 영역(126)에 할당될 수 있다.

[0013]

일 구현에서, 저장 컨트롤러(112)는 또한 저장 영역들(122, 124, 126)의 매핑을 논리 블록 어드레스들(LBA들)에 발생시키고 저장한다. 구체적으로, 저장 매체(114)의 사용가능 저장 영역들은 LBA들에 매핑된다. 그러한 구현에서, 저장 영역(120)은 또한 다수의 블록들로 분할되고 각각의 블록에는 논리 블록 어드레스(LBA)가 주어진다. 예를 들어, 저장 영역들(122-126) 각각은 각각의 블록들이 LBA들에 매핑되는 상태에서 복수의 블록들을 포함할 수 있다.

[0014]

더욱이, 저장 컨트롤러(112)는 LBA들의 매핑을 프로세서(104)에 이용가능한 저장 영역들에 또는 저장 매체(114)에 액세스를 요구하는 다른 디바이스들에 행할 수도 있다. 또 다른 대안적으로, 저장 컨트롤러(112)는 저장 매체(114)의 저장 용량을 그러한 외부 디바이스들에 전달할 수도 있다. 데이터가 저장 매체(114)에 기록되므로, 저장 컨트롤러는 어느 저장 영역들이 이미 사용되는지 및 어느 저장 영역들이 여전히 부가 저장에 이용가능한지를 기록한다. 부가 데이터를 저장 매체(114)에 기록하는 요청에 대응하여, 저장 컨트롤러(112)는 하나 이상의 저장 영역들(122, 124, 126) 상의 저장을 위한 이용가능성을 결정하고, 그러한 부가 데이터를 저장하는 목적지 저장 영역을 결정한다. 예를 들어, 저장 컨트롤러(112)는 저장 영역(126)이 데이터를 저장하는데 이미 사용된 것을 결정하면, 부가 데이터의 저장에 대한 새로운 요청들은 저장 영역(122, 124) 등에 전송된다.

[0015]

일 구현에서, 저장 영역들(122, 124, 126) 등은 정적으로 매핑된 또는 동적으로 매핑된 저장 영역들로 지정된다. 그러한 구현에서, 저장 영역들에 LBA들의 매핑은 주어진 영역이 정적으로 할당되었는지 또는 동적으로

로 할당되었는지에 의존한다. 정적으로 할당된 저장 영역에서, 주어진 물리 스토리지 어드레스가 LBA에 매핑되면, 그러한 매핑은 장래에 변화하지 않는다. 다른 한편, 동적으로 매핑된 영역에서, 저장 컨트롤러(112)는 LBA의 매핑을 물리 어드레스로 동적으로 변경할 수 있다.

[0016]

정적으로 및 동적으로 할당된 영역들로 저장 영역(120)의 분할은 저장 컨트롤러(112)가 저장 매체(114)의 오버프로비저닝을 동적으로 변경하는 것을 허용한다. 예를 들어, 저장 영역(120)의 전체 저장 용량이 1.1 TB인 것을 고려하면, 저장 컨트롤러(112)는 1.09 TB를 사용가능 저장 용량으로 보고할 수 있는 한편, 저장 디바이스(110)의 전체 수명을 위해 최소 0.1 TB의 성능 증대 매체 캐시를 여전히 제공한다. 저장 컨트롤러(112)는 저장 영역(120)의 적어도 일부를 동적으로 매핑함으로써 그러한 오버프로비저닝을 달성한다. 도 1에 예시된 예에서, 저장 영역(120)의 전체 저장 용량이 1.1 TB인 경우, 1.0 TB을 갖는 저장 영역(122)은 정적으로 매핑된 것으로 지정된다. 0.09 TB 용량을 갖는 저장 영역(124) 및 0.01 TB 용량을 갖는 저장 영역(126)은 동적으로 매핑된다. 더욱이, 저장 컨트롤러(112)는 저장 영역들(122 및 124)(도 1에서의 130으로 언급됨)의 결합된 용량을 저장 디바이스(110)에 대한 전체 이용가능 용량으로서 보고한다. 그러한 경우에, 영역들(124 및 126)의 결합된 저장 영역(도 1에서의 140으로 언급됨)은 데이터의 일시적 카핑, 에러 체크, 데이터의 분류 등과 같은 다양한 성능 강화 목적들을 위한 매체 캐시로서 사용된다. 매체 캐시의 다른 사용들은 특정 호스트 워크로드들에서 랜덤 기록 성능을 증대하는 것을 포함하며, 이는 관리하기가 점점 더 어렵다. 대안적으로, 매체 캐시는 저장되고 있는 호스트 데이터에 대한 추가 패리티(에러 정정 정보)를 분류함으로써 신뢰성을 증대하기 위해 사용될 수도 있다. 또 다른 대안적으로, 매체 캐시는 특정 호스트 워크로드가 신뢰성을 추진하는데 특히 열악한 소위 "워크 애플리케이션(work amplification)"의 효과들을 최소화함으로써 보증 요건들을 충족시키기 위해 사용될 수도 있다. 결과적으로, 동적으로 매핑된 저장 영역(124)은 성능 강화, 및 다른 기능들을 위한 매체 캐시로서 사용가능하고 동시에 사용되는 것 둘 다로서 보고된다.

[0017]

그러한 구현에서, 동적으로 매핑된 저장 영역(124)이 하나 이상의 성능 또는 신뢰성 강화 기능들에 사용되지만, 저장 영역(124)의 저장 용량은 또한 사용가능 공간으로서 호스트에 보고된다. 즉, 저장 영역(124)은 데이터를 저장하는 것 뿐만 아니라 성능 강화 기능들 둘 다에 사용된다. 또 다른 구현에서, 성능 강화 기능들에 사용되는 동적으로 매핑된 저장 영역(124)은 드물게 사용되는 또는 거의 사용되지 않는 저장 매체의 저장 영역들에 매핑된다. 그러한 구현에서, 더 많이 사용될 것 같은 LBA 공간의 부분들은 정적으로 매핑된 저장 영역(122)에 매핑된다. 호스트 어드레스가 가능 LBA 공간의 덜 빈번하게 액세스된 부분들을 동적으로 매핑된 저장 영역(124)에 매핑하는 것은 동적으로 매핑된 저장 영역(124)이 저장 디바이스(110)의 수명의 대부분에 대한 성능 강화 기능들에 이용가능한 것을 보장한다.

[0018]

일 구현에서, 동적 오버프로비저닝의 양은 저장 디바이스(110)를 사용하는 고객의 타입, 저장 디바이스(110)가 사용되는 시장 등에 기초할 수 있다. 예를 들어, 저장 디바이스(110)가 가정에서 개인적 사용을 위해 소비자에게 의해 사용되게 되면, 매우 사용될 것 같지 않거나 매우 드물게 사용되는 LBA 공간의 더 큰 부분들이 있을 것 같다. 그러한 경우에, 저장 영역(120)의 더 큰 부분은 동적으로 매핑된 저장 영역(124)의 일부가 되는 것으로서 동적으로 매핑되고 매우 작은 부분은 오버프로비저닝된 저장 영역(126)으로 지정된다. 즉, 저장 디바이스의 그러한 타겟 사용에서, 고객에게 보고되는 전체 저장 공간(120)의 백분율은 매우 높다. 다른 한편, 저장 디바이스(110)가 LBA 공간의 대부분이 매우 빈번하게 사용될 것 같은 산업에서 사용되게 되면, 지정된 오버프로비저닝된 저장 영역(126)은 성능 레벨들을 유지하기 위해 더 커야 한다. 결과적으로, 그러한 경우에, 고객에게 보고되는 전체 저장 공간(120)의 백분율은 소비자에 의해 사용되고 있는 저장 디바이스(110)의 경우에서보다 더 낮다.

[0019]

도 2는 최대 용량의 저장 디바이스의 매핑(200)을 예시한다. 구체적으로, 도 2는 LBA 맵(210)과 저장 디바이스의 실제 물리적 용량(220) 사이의 매핑을 예시한다. 예시된 구현에서, 저장 컨트롤러는 LBA 맵(210)의 다양한 영역들의 사용 패턴들을 결정한다. 따라서, 예를 들어 도 2에서, 영역(212)은 평균의 사용을 갖도록 결정되며, 영역(214)은 매우 조금 사용되도록 결정되는 반면, 영역들(216 및 218)은 빈번한 또는 평균을 넘는 사용을 갖도록 결정된다. 일 구현에서, 다양한 영역들에 대한 초기 사용 패턴들은 기존 제품들의 워크로드들을 감시하는 것에 기초한 이력 사용 패턴들에 기초하여 결정된다. 대안적 구현에서, 사용 패턴들, 및 따라서 동적 영역에 특정 LBA들의 매핑은 저장 디바이스의 수명을 넘어 동적으로 결정되지만, 매핑의 초기 상태는 언급된 바와 같은 이력 사용에 기초할 수 있다.

[0020]

LBA 맵(210)에 의해 보고된 바와 같은 저장 디바이스의 저장 용량은 저장 매체의 물리적 용량(220)보다 근소하게 더 낮다. 예를 들어 물리적 용량(220)이 1.10 TB이면, 보고된 저장 용량은 LBA 맵(210)에 따라 1.09 TB로 설정된다. 물리 저장 용량의 0.01 TB의 차이는 성능 강화 및 시스템 자원 관리를 포함하는 다양한 기능들에 대한 최소 매체 캐시 크기로서 사용된다. 도 2는 그러한 0.01 TB의 최소 매체 캐시(226)를 예시한다. 매체 캐시(22

6)의 저장 공간은 동적으로 매핑되고 이용가능 저장 용량으로서 보고되지 않는다. 더욱이, 저장 용량(220)의 잔여분은 정적으로 매핑된 영역들(222) 및 동적으로 매핑된 영역(224)으로 분할된다. 저장 컨트롤러는 240에 의해 예시된 바와 같이, LBA 맵(210)의 조금 사용된 영역(214)을 동적으로 매핑된 영역(224)에 매핑한다. 예를 들어, 동적으로 매핑된 영역(224)의 크기는 0.09 TB일 수 있고, 매체 캐시로서 지정된 저장 영역(226)의 크기는 0.01 TB와 같고, 정적으로 매핑된 영역들(222)의 크기는 1.0 TB와 같다. 그러한 경우에, 저장 디바이스가 우선 사용하게 될 때(디바이스에 기록된 사용자 데이터 없음), 1.09 TB와 같은 저장 영역들(222 및 224)의 결합된 용량은 호스트/사용자에게 용량으로서 보고될 수 있다. 동시에, 0.1 TB와 같은 저장 영역들(224 및 226)의 결합된 용량은 성능 강화 및/또는 신뢰성 강화 기능들을 위한 매체 캐시로서 사용될 수 있다.

[0021]

호스트들에 의해 사용되는 다양한 운영 체제들에서, 호스트는 "호스트 트림 커맨드", 또는 균등물을 이용할 수 있으며, 이는 호스트 디바이스가 더 이상 그러한 LBA들을 "유효한" 또는 "기록된" 것으로 간주할 필요가 없도록 호스트가 특정 LBA들에 기록하고 그 다음 그들을 "트림"할 수 있는 것을 의미한다. 예를 들어, 도 2에서, 운영 체제는 초기에 데이터를 빈번하게 LBA 공간(218)에 기록할 수 있다. 이러한 경우에, LBA 공간(218)은 동적으로 매핑된 영역(224)에 매핑되지 않는다. LBA 공간(210)이 디바이스의 수명을 넘어 사용되므로, 조금 사용된 영역(214)은 점점 더 적은 동적으로 사용된 영역이 다양한 매체 캐시 또는 성능 강화 기능들에 이용가능하도록 점점 더 많이 사용될 수 있다. 그러한 시나리오에서, 호스트는 트림 커맨드를 빈번하게 사용되는 것으로 여겨진 공간(218)의 비어있는 일부에 발행하면, 컨트롤러는 동적으로 매핑된 저장 영역(224)에 LBA 공간(214)의 이전 매핑(240) 대신에 이제 새로이 비게 된 LBA 공간(218)이 동적으로 매핑된 저장 영역(224)에 매핑되도록 물리 저장 용량(220)에 LBA 공간(210)의 매핑을 동적으로 변경할 수 있다. 물리 저장 용량(220)에 LBA 공간(210)의 다양한 부분들의 매핑의 그러한 동적 변경은 또한 LBA 공간(210)의 사용 패턴들의 다른 변경들에 대응하여 수행될 수 있다.

[0022]

대안적 구현에서, 물리 저장 용량(220)에 LBA 공간(210)의 다양한 부분들의 그러한 동적 매핑은 호스트에 의해 LBA 공간(210)의 하나 이상의 부분들의 비어 있는 것에 기초하지 않을 수 있다. 그 대신에, 시스템은 LBA 공간(210)의 사용을 감시하고 그러한 부분들에 대한 사용의 패턴의 변경들에 기초하여 그러한 리매핑을 결정할 수 있다. 따라서, 예를 들어 컨트롤러는 동적으로 매핑된 저장 영역(224)에 매핑되는 제 1 LBA 공간이 그것의 사용 패턴의 변경을 갖는 것을 결정하면, 컨트롤러는 제 1 LBA 공간을 물리 저장 용량(220)의 일부 다른 부분에 동적으로 리매핑하고 그 다음 제 2 LBA 공간을 LBA 공간(210)으로부터 동적으로 매핑된 저장 영역(224)으로 매핑할 수 있다. 이러한 방식으로, 동적으로 매핑된 저장 영역(224)에 매핑되는 LBA들은 그들 자체에 동적으로 매핑된다. 이것은 조금 사용된 것으로 예상되고 따라서 동적으로 매핑된 저장 영역(224)에 매핑되는 LBA 공간이 갑자기, 아주 많이 사용되는 상황들을 시스템이 수용하는 것을 허용한다. 그러한 동적 할당의 결과로, 동적으로 매핑된 저장 영역(224)을 제공하는 성능 강화 및 다른 장점들은 그러한 예상되지 않은 상황들에서 유지된다. 더욱이, 동적으로 매핑된 영역(224)으로부터 LBA 공간(210)의 특정 부분들의 동적 언매핑 및 동적으로 매핑된 영역(224)에 LBA 공간(210)의 일부 다른 부분들의 후속 매핑은 저장 디바이스의 사용의 수명을 넘어 지속할 수 있다. 그러한 지속적 언매핑 및 리매핑은 다양한 성능 강화 기능들, 신뢰성 강화 기능들 등을 위해 동적으로 매핑된 영역(224)의 지속적인 사용을 제공한다.

[0023]

사용자가 저장 매체를 사용하기 시작할 때, 전체 정적으로 매핑된 1.0 TB의 저장 용량이 사용자/호스트에 의해 유효한 것으로 간주되지 않는다면 그리고 간주될 때까지, 동적으로 매핑된 영역들(224 및 226)의 둘 다는 함께 다양한 성능 강화 기능들에 사용된다. 따라서, 0.1 TB의 저장 용량은 전체 정적으로 매핑된 1.0 TB의 용량이 유효한 것으로 간주되지 않는 한, 실질적으로 0.01 TB의 공간만을 영구 매체 캐시(226)로서 제공하는 데도 불구하고 성능 강화 기능들에 이용가능하다. 따라서, 사용자 또는 호스트 디바이스는 작은 크기의 영구적으로 할당된 매체 캐시(226)로도 임의의 저하가 레벨 성능의 레벨인 것을 인식하지 않는다. 대체로, 사용자들(특히 소비자 시장에서의 사용자들)이 그들의 저장 디바이스들의 전체 용량을 결국 사용하게 되지 않는 것을 고려하면, 저장 디바이스들의 성능은 저장 디바이스의 수명의 대부분을 초과하는 것을 겪지 않는다.

[0024]

사용자가 데이터를 애드호크(ad-hoc) 방식으로 저장 디바이스에 기록할 때, 동적으로 매핑된 영역들(224) 뿐만 아니라 매체 캐시(226)는 데이터가 이러한 영역들에서 그것의 단말 목적지로 이동되기 전에, 데이터를 일시적으로 저장하고, 데이터를 처리하고, 데이터를 분류하는 등을 위해 사용될 수 있다. 그러한 단말 목적지는 정적으로 매핑된 영역들(222) 내에 있을 수 있다. 결과적으로, 전체 동적으로 매핑된 용량의 크기가 매체 캐시(226)의 크기보다 훨씬 더 큰 것을 고려하면, 저장 디바이스의 성능은 사용자에게 의해 인지되는 바와 같이 저하되지 않는다. 동시에, 저장 디바이스 제조자는 더 크게 이용가능한 용량을 사용자에게 보고할 수 있으며, 따라서 스토리지를 제공하는 유효한 비용을 감소시킨다.

- [0025] 호스트 디바이스와 같은 사용자가 저장 용량 중의 점점 더 많은 것을 사용하고 유효하게 유지하는 것을 시작하므로, LBA 맵의 드물게 사용된 부분(214)이 동적으로 매핑된 저장 영역(224)에 매핑될 때에도, 동적으로 매핑된 영역(224)의 일부 사용이 저장을 위해 발생할 것 같다. 예를 들어, 호스트 디바이스는 LBA 부분(214) 내에 있는 LBA에의 기록을 배타적으로 요청할 수 있으며, 이는 동적으로 공급된 저장 영역(224)에 기록을 야기한다. 그러한 경우에, 호스트에 의해 제공되는 데이터는 동적으로 매핑된 영역(224) 내의 일부 영역(230)에 "피닝"된다. 결과적으로, 성능 강화 기능들에 이용가능한 동적으로 매핑된 영역(224) 내의 저장 공간의 양은 영역(230) 내의 데이터의 크기에 의해 감소되며, 이는 매체 캐시 내에 "피닝"된다. 피닝은 데이터가 영구적으로 매체 캐시에서 상주하는 것을 의미한다. 점점 더 많은 데이터가 동적으로 매핑된 영역(224)에서 피닝되므로, 동적으로 매핑된 저장 영역(224)을 사용하여 성능 강화 기능들을 제공할 시에 저장 디바이스의 능력이 감소하여, 일부 가능한 성능 저하를 야기한다.
- [0026] 일 구현에서, 데이터가 동적으로 매핑된 영역(224) 내의 임의의 위치에 피닝될 때, 컨트롤러는 동적으로 매핑된 영역(224)에서 그 데이터의 단말 레스팅 플레이스(resting place)를 관리한다. 예를 들어, 영역(230)에 기록된 데이터는 동적으로 매핑된 영역(224) 내의 대안적 위치로 이용될 수 있다. 동적으로 매핑된 영역(224)에 피닝되는 데이터의 그러한 동적 매핑은 성능 강화 기능들을 위해 동적으로 매핑된 영역(224)의 보다 효율적인 사용을 허용한다. 컨트롤러가 피닝된 데이터를 여전히 관리하고 있으면, 그러한 피닝된 데이터는 단말 레스팅 플레이스로부터 주변으로 여전히 이동가능해야 된다는 점을 주목한다.
- [0027] 도 3은 성능 강화 기능들에 사용될 수 있는 동적으로 매핑된 영역(300)의 확대도의 일 예를 예시한다. 구체적으로, 동적으로 매핑된 영역(300)은 다수의 상이한 위치들(302, 304, 306) 등에 호스트 기록들을 수용할 수 있다. 예시된 바와 같이, 그러한 호스트 기록들은 동적으로 매핑된 영역(300) 내에 이질적인 위치에 위치된다. 결과적으로, 데이터를 분류하는 것, 데이터를 미리 조사 분석하는 것 등과 같은 성능 증대 기능들을 위한 동적으로 매핑된 영역(300)의 사용은 불리하게 작용된다. 일 구현에서, 호스트 기록 요청들로부터 위치들(302, 304, 306)로 데이터는 동적으로 매핑된 영역(320)에 의해 예시된 바와 같이, 위치들(302a, 302b, 302c)에 동적으로 매핑된다. 그러한 동적 매핑의 결과로, 동적으로 매핑된 영역(320)의 큰 부분(330)은 성능 강화 기능들에 여전히 이용가능하다. 결과적으로, 부분(330) 오버프로비저닝되고 매체 캐시로서 지정되는 저장 영역과 함께, 다양한 성능 및 신뢰성 강화 기능들에 사용될 수 있다. 동적 매핑 영역(320)의 매핑에 대한 정보는 매체 캐시로서 지정되는 오버프로비저닝된 저장 영역 뿐만 아니라 저장 디바이스상의 다양한 다른 위치들에 저장될 수 있다.
- [0028] 동적 매핑의 성능/신뢰성 강화 용량은 동적으로 매핑된 영역(300)의 매핑된 부분 대 동적으로 매핑된 영역(300)의 매핑되지 않은 영역의 비율에 기초하여 결정될 수 있다. 동적으로 매핑된 영역(300)에 호스트에 의한 타겟화된 기록들로 인해 매핑되게 되는 동적으로 매핑된 영역(300)의 양이 더 커질수록, 매핑된 비율 대 매핑되지 않은 비율은 더 커지고 동적으로 매핑된 영역(300)의 성능 강화 용량은 더 작아진다. 반면에, 더 작은 매핑된 비율 대 매핑되지 않은 비율은 동적으로 매핑된 영역(300)의 대부분이 성능 강화 동작들에 이용가능한 것을 암시한다. 매핑된 및 매핑되지 않은 둘 다의 임의의 데이터에 대해 가능하다.
- [0029] 동적으로 매핑된 영역(300)이 드물게 사용되거나 사용되지 않는 LBA 공간에 매핑될 때, 동적으로 매핑된 영역(300)의 매핑된 영역 대 매핑되지 않은 영역의 비율은 동적으로 매핑된 영역(300)에 의해 제공되는 성능 강화 기능들이 실질적으로 일정할 만큼 충분히 낮게 유지될 수 있다. 동시에, 동적으로 매핑된 영역(300)을 제공함으로써, 저장 디바이스 제공자는 매체 캐시로서 오버프로비저닝하는 저장 영역의 크기를 감소시킬 수 있고, 따라서 더 큰 용량을 제공하기 위해 더 낮은 비용을 제공한다. 더욱이, 동적으로 매핑된 영역(300)을 제공함으로써, 저장 제공자는 만족스러운 매체 성능 및 신뢰성에 요구되는 매체 캐시 크기와 오버프로비저닝 사이의 관계를 실질적으로 분리시킬 수 있다.
- [0030] 도 4는 성능 강화 기능들에 사용되는 동적으로 할당된 저장 영역의 크기를 결정하는 방법의 하나의 예시적 흐름도(400)를 예시한다. 동작(402)은 저장 디바이스에 대한 타겟 사용을 결정한다. 예를 들어, 저장 디바이스의 타겟 사용은 파이버 채널 네트워크 저장 디바이스 등으로서 소비자의 개인용 컴퓨터 내에 있을 수 있다. 타겟 사용자 타입에 의존하여, 저장 디바이스는 상이한 방식으로 사용될 수 있다. 동작(404)은 타겟 사용자의 사용 패턴을 결정한다. 예를 들어, 개인적 사용을 위해 저장 디바이스를 사용하는 소비자인 최종 사용자는 제공되는 저장 공간 모두를 실질적으로 사용할 것 같지 않다. 다른 한편, 저장 디바이스를 데이터 서버로서 사용하는 기업의 최종 사용자 저장 공간의 대부분을 매우 빈번히 액세스할 수 있다. 더욱이, 동작(404)은 또한 최종 사용자에 대한 성능 강화 요건들을 결정한다.
- [0031] 동작(406)은 타겟 사용자의 사용 패턴들, 성능 강화 요건들 등에 기초하여 저장 디바이스에 제공될 오버프로비

저널의 양을 결정한다. 예를 들어, 보다 고속의 성능에 대한 요건을 갖는 타겟 사용자에게 대해, 동작(406)은 다른 사용자들 등에 대해서보다 더 많은 오버프로비저닝을 제공하는 것을 결정할 수 있다. 그 후에, 동작(408)은 저장 공간의 무슨 부분이 동적으로 매핑된 저장 영역으로서 구성되게 될지를 결정한다. 동작(408)은 타겟 사용자 등의 사용 패턴들에 기초하여 그러한 결정을 행할 수 있다.

[0032]

동작(410)은 LBA 공간의 드물게 사용된 부분들을 결정한다. 예를 들어, 그러한 드물게 사용되는 공간은 이력 사용 패턴 데이터에 기초할 수 있다. 대안적으로, 드물게 사용되는 또는 사용되지 않는 공간은 저장 디바이스가 필드에 있고 사용자 액세스 패턴들에 노출된 후 런타임에서 동적으로 결정될 수 있다. 또 대안적으로, "드물게 사용되는" 영역은 각각의 사용자/호스트에 대해 변화할 수 있다. 동작(412)은 LBA 공간의 드물게 사용된 부분들을 동적으로 매핑된 저장 영역들에 매핑한다. LBA 공간의 이러한 부분들이 드물게 사용되므로, 저장 공간의 동적으로 매핑된 영역은 빈번한 액세스 요청들을 가질 가능성이 더 낮다. 이것은 성능 강화 기능들을 위해 동적으로 할당된 저장 영역의 보다 효율적 사용을 허용한다. 동작(414)은 동적으로 매핑된 저장 영역의 사용을 감시한다. 구체적으로, 동작(414)은 동적으로 매핑된 저장 영역의 매핑된 부분 대 매핑되지 않은 부분의 비율을 감시할 수 있다. 일 구현에서, 동작(416)은 동적으로 매핑된 저장 영역의 매핑되지 않은 부분의 크기 대 동적으로 매핑된 저장 영역의 매핑된 부분의 크기의 비율에 기초하여 하나 이상의 성능 강화 동작들을 변경한다. 예를 들어, 비율에 기초하여, 동작(416)은 조금 사용된 영역들에 매핑된 데이터를 언매핑하는 것을 시작하고 그것을 아주 많이 사용되는 영역들에 매핑된 데이터로 스왑(swap)할 수 있어, 아주 많이 사용되는 영역들에서의 데이터는 매체 캐시에 매핑되고 조금 사용된 영역에서의 데이터는 그것의 단말 레스팅 플레이스에 매핑된다. 예를 들어, 동적으로 매핑된 저장 영역은 비율이 임계값 위에 있는 것으로 결정되면, 임의의 성능 강화 기능들에 더 이상 사용되지 않는다.

[0033]

도 5는 데이터 저장 시스템 내에 매체의 동적 오버프로비저닝을 제공하는 동작들의 하나의 예시적 흐름도(500)를 예시한다. 구체적으로, 동작(502)은 기록 요청을 저장 매체에 수신한다. 동작(504)은 또한 성능 강화 동작들을 위해 사용되는 저장 매체의 동적으로 매핑된 저장 영역에의 타겟화된 기록에 대한 것인지를 결정하도록 기록 요청을 평가한다. 기록 요청이 저장 매체의 동적으로 매핑된 저장 영역에 임의의 그러한 타겟화된 기록들을 포함하지 않으면, 그 다음 기록 요청이 서빙되고, 필요한 것으로서 초기의 미리 조사 분석하는 것 후에, 동작(506)은 데이터를 저장 매체의 정적으로 매핑된 영역에서의 그것의 단말 레스팅 플레이스에 저장한다.

[0034]

그러나, 동작(504)이 기록 요청의 적어도 일부가 타겟 기록을 저장 매체의 동적으로 매핑된 저장 영역에 포함하면, 동작(508)은 저장 매체의 동적으로 매핑된 저장 영역에 기록되게 될 데이터를 다른 데이터로부터 분리한다. 그 후에, 정적으로 피닝된 데이터는 정적으로 매핑된 저장 영역들 내의 그것의 단말 레스팅 플레이스에 저장된다. 동작(510)은 동적으로 피닝된 데이터의 동적 매핑을 결정하기 위해 그러한 데이터를 평가하며, 이는 동적으로 피닝된 저장 영역 내에 저장되게 될 데이터이다. 동작(510)은 그러한 동적으로 피닝된 데이터를 동적으로 매핑된 저장 영역의 시작 부분, 동적으로 매핑된 저장 영역의 종료 부분 등에 매핑하는 것을 결정할 수 있다. 일 구현에서, 동작(512)은 동적 매핑에 대한 정보를 저장한다. 그 후에, 저장 동작(514)은 동작(510)에서 결정된 동적 매핑에 따라 데이터를 동적으로 매핑된 저장 영역에 저장한다.

[0035]

도 6은 성능 강화 기능들을 위해 이용가능한 전체 매체 캐시의 크기와 매체 캐시의 동적으로 매핑된 부분에 기록된 LBA들 사이의 가능한 관계를 예시하는 그래프(600)를 예시한다. 이러한 구현에서, 0.01 TB의 매체 캐시가 오버프로비저닝된다는 점 및 동적으로 매핑된 영역이 한번 충전되면, 필연적으로 비게 될 수 없다는 점이 추정된다. 다른 0.09 TB의 저장 공간은 이용가능할 때 동적으로 매핑되고 성능 강화 기능들을 위해 사용된다. 라인(610)에 의해 예시된 바와 같이, 매체 캐시의 동적으로 매핑된 부분에 기록된 어떤 LBA들도 없을 때, 실질적으로 0.1 TB(0.01 TB + 0.09 TB)의 매체 캐시 모두가 성능 강화 기능들을 위해 이용가능하다. 점점 더 많은 매체 캐시의 동적으로 매핑된 부분이 기록됨에 따라, 점점 더 적은 전체 매체 캐시가 성능 강화를 위해 이용가능하다. 그래프(600)는 또한 라인(612)에 의한 드라이브의 성능의 레벨을 도시한다. 예시된 바와 같이, 점점 더 많은 매체 캐시의 동적으로 매핑된 부분이 기록됨에 따라, 저장 디바이스의 성능이 불리하게 작용된다.

[0036]

본 명세서에 설명되는 구현들은 하나 이상의 컴퓨터 시스템들에서 논리 단계들로 구현될 수 있다. 본 명세서에 설명되는 다양한 구현들의 논리 동작들은 (1) 하나 이상의 컴퓨터 시스템들에서 실행되는 프로세서 구현 단계들의 시퀀스로서 및 (2) 하나 이상의 컴퓨터 시스템들 내에서 상호연결된 기계 또는 회로 모듈들로서 구현된다. 구현은 컴퓨터 시스템의 성능 요건들에 의존하는 선택의 문제이다. 따라서, 본 명세서에 설명되는 구현들의 구현들을 구성하는 논리 동작들은 동작들, 스텝들, 객체들, 또는 모듈들로 다양하게 지칭된다. 더욱이, 논리 동작들은 다르게 명백히 청구되지 않으면 임의의 순서로 수행될 수 있거나 특정 순서는 본래 청구 언어에 의해 요구

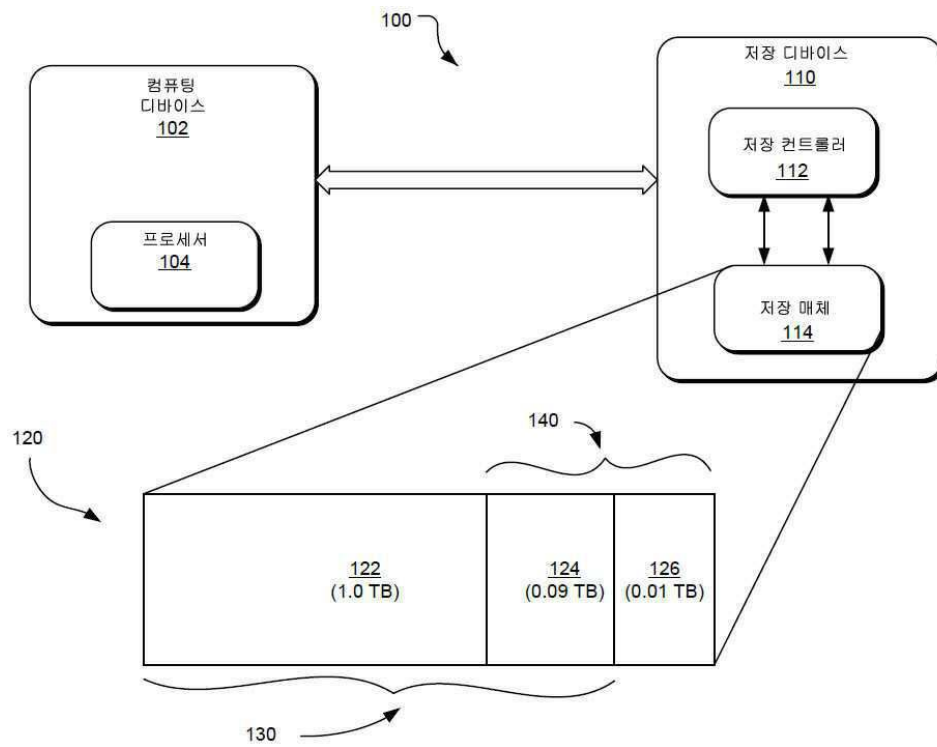
된다는 점이 이해되어야 한다.

[0037] 명료성을 위해, 본 명세서에 설명되는 구현들의 루틴 기능들은 도시되고 설명되는 것은 아니다. 임의의 그러한 실제 구현의 개발에서, 다수의 구현 특정 결정들은 응용에 따름 - 및 사업 관련 제약과 같은 개발자의 특정 목적들을 달성하기 위해 이루어지고, 그의 특정 목적들은 일 구현으로부터 다른 구현으로 그리고 한 개발자로부터 다른 개발자로 변화된다는 점이 이해될 것이다.

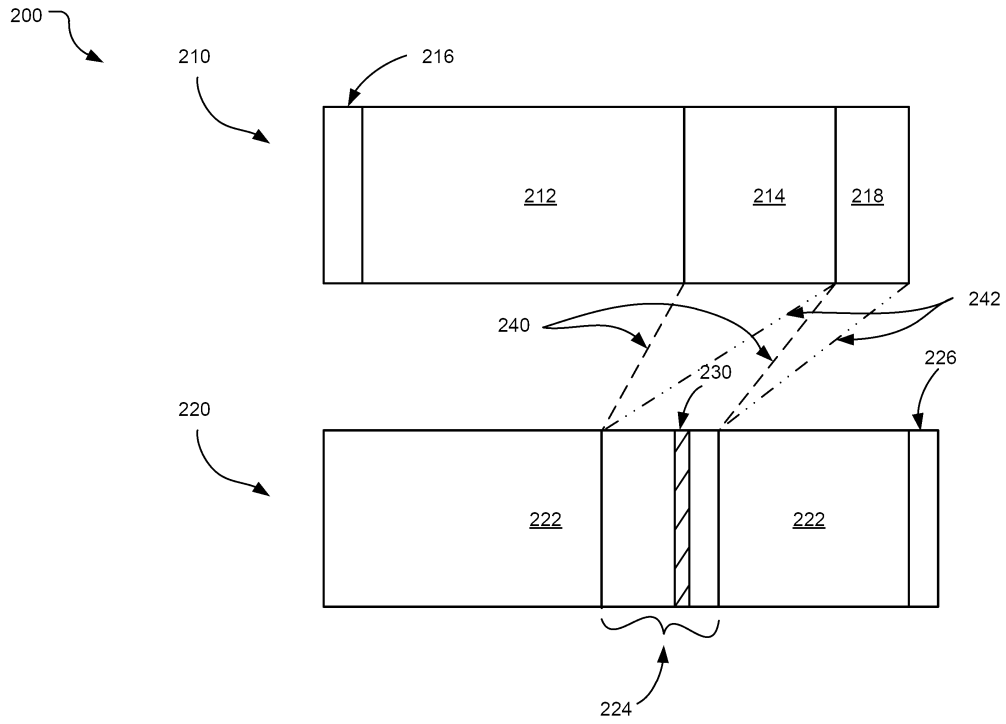
[0038] 상기 명세서, 예들, 및 데이터는 예시적 구현들의 구조 및 사용의 완전한 설명을 제공한다. 많은 대체 구현들은 본 발명의 사상 및 범위로부터 벗어나는 것 없이 이루어질 수 있기 때문에, 본 발명은 이하에 첨부된 청구항들에 있다. 더욱이, 상이한 구현들의 구조적 특징들은 열거된 청구항들로부터 벗어나는 것 없이 또 다른 구현에 결합될 수 있다.

도면

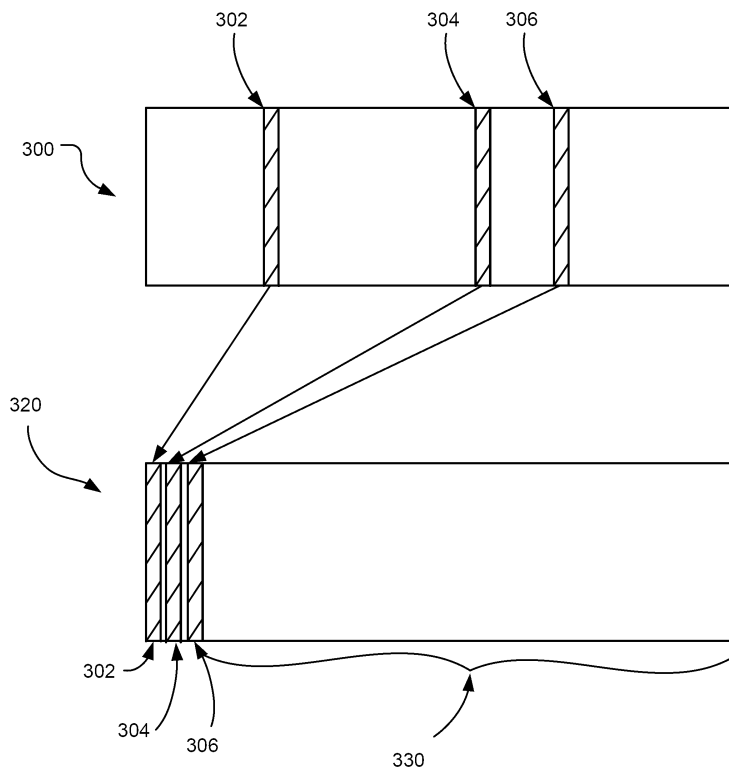
도면1



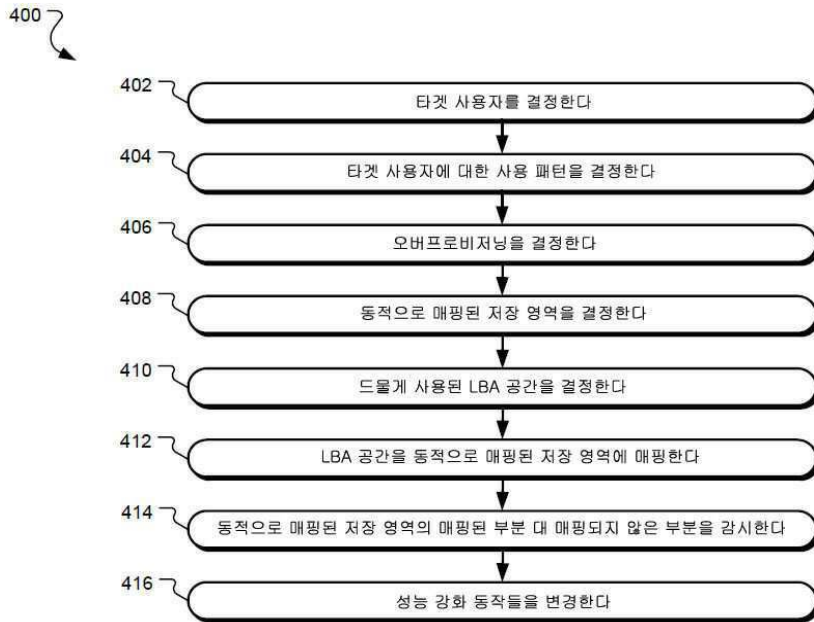
도면2



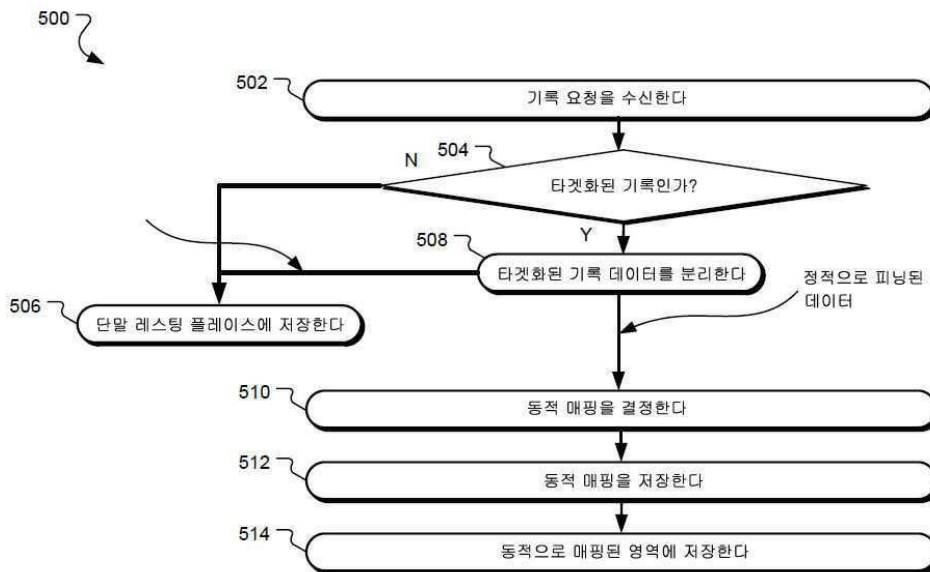
도면3



도면4



도면5



도면6

