



(19) 대한민국특허청(KR)  
(12) 등록특허공보(B1)

(45) 공고일자 2023년06월08일  
(11) 등록번호 10-2541458  
(24) 등록일자 2023년06월02일

(51) 국제특허분류(Int. Cl.)  
G06F 3/06 (2006.01) G06F 12/02 (2018.01)  
(52) CPC특허분류  
G06F 3/064 (2013.01)  
G06F 12/0246 (2013.01)  
(21) 출원번호 10-2016-0076697  
(22) 출원일자 2016년06월20일  
심사청구일자 2021년06월07일  
(65) 공개번호 10-2017-0008152  
(43) 공개일자 2017년01월23일  
(30) 우선권주장  
62/192,045 2015년07월13일 미국(US)  
(뒷면에 계속)  
(56) 선행기술조사문헌  
JP2010018004 A  
KR1020140033099 A

(73) 특허권자  
삼성전자주식회사  
경기도 수원시 영통구 삼성로 129 (매탄동)  
(72) 발명자  
최창호  
미국 캘리포니아주 95120 산호세 마운트 홀리 드  
라이브 6622  
후오, 지안지안  
미국 캘리포니아주 95120 산호세 올드 밀 코트  
6602  
(뒷면에 계속)  
(74) 대리인  
특허법인 고려

전체 청구항 수 : 총 20 항

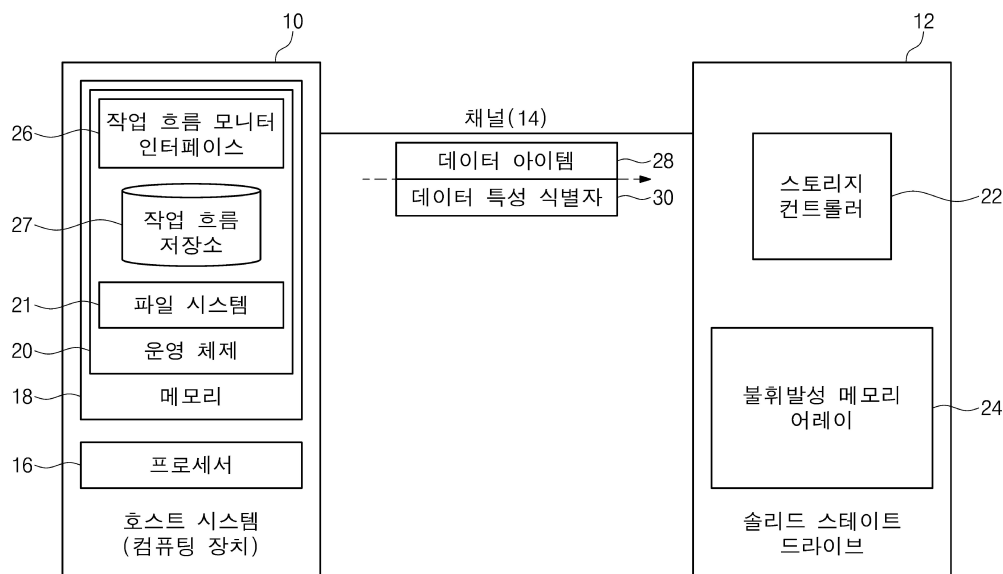
심사관 : 신현상

(54) 발명의 명칭 불휘발성 메모리 장치에서의 데이터 특성 기반 데이터 배치

(57) 요약

실시 예들은 불휘발성 메모리 장치의 스토리지 컨트롤러에 의해 수행되는 불휘발성 메모리 장치 내부의 데이터 특성 기반 데이터 배치를 위해 설명된다. 본 발명의 실시 예는, 실행 중인 응용 및 컴퓨터 장치에 연결된 하드웨어 장치 중 적어도 하나를 검출하는 소프트웨어 구성 요소를 컴퓨터 장치에서 실행하는 단계, 실행 중인 응용 및 (뒷면에 계속)

대표도



하드웨어 장치 중 적어도 하나를 검출하는 것에 응답하여, 소프트웨어 구성 요소가 작업 흐름 저장소를 검색하여 실행 중인 응용 및 하드웨어 장치 중 적어도 하나와 연관된 미리 정해진 작업 흐름을 찾는 단계, 소프트웨어 구성 요소가 실행 중인 응용 및 하드웨어 장치 중 적어도 하나의 활동을 미리 정해진 작업 흐름과 비교하는 단계, 그리고 데이터 아이템들 및 할당된 데이터 특성 식별자들이 채널을 통해 불휘발성 메모리 장치에 전송되어 저장되도록 실행 중인 응용 또는 하드웨어 장치에 의해 사용되는 데이터 아이템들에 데이터 특성 식별자들을 자동적으로 할당하도록 미리 정해진 작업 흐름을 이용하는 단계를 포함한다. 미리 정해진 작업 흐름은 미리 정해진 데이터 특성 식별자들을 실행 중인 응용 또는 하드웨어 장치에 의해 불휘발성 메모리에 기입되는 데이터 아이템들의 상이한 타입들과 연관한다. 불휘발성 메모리 장치는 데이터 특성 식별자들을 읽고, 그리고 동일한 데이터 특성 식별자들을 갖는 데이터 아이템들이 동일한 블록에 저장되도록, 데이터 아이템을 저장할 불휘발성 메모리 장치의 블록을 식별한다.

(52) CPC특허분류

**G06F 12/0253** (2013.01)

**G06F 3/0679** (2013.01)

**G06F 3/0689** (2013.01)

(30) 우선권주장

62/245,100 2015년10월22일 미국(US)

15/144,588 2016년05월02일 미국(US)

(72) 발명자

**췁, 데릭**

미국 캘리포니아주 94587 유니온 시티 치펜데일 드  
라이브 4315

**크리쉬나무티, 프라빈**

미국 캘리포니아주 94086 서니배일 애스터 애비뉴  
1035 2187호

**후엔, 형관**

미국 캘리포니아주 94015 손힐 드라이브 달리 시티  
864

## 명세서

### 청구범위

#### 청구항 1

컴퓨터 장치가 불휘발성 메모리 장치 내에서 데이터 특성 기반 데이터 배치를 활용하게 하는 인터페이스를 제공하는 방법에 있어서:

실행 중인 응용 및 상기 컴퓨터 장치에 연결된 하드웨어 장치 중 적어도 하나를 검출하는 소프트웨어 구성 요소를 상기 컴퓨터 장치에서 실행하는 단계;

상기 실행 중인 응용 및 상기 하드웨어 장치 중 적어도 하나를 검출하는 것에 응답하여, 상기 소프트웨어 구성 요소가 작업 흐름 저장소를 검색하여 상기 실행 중인 응용 및 상기 하드웨어 장치 중 적어도 하나와 연관된 미리 정해진 작업 흐름을 찾는 단계;

상기 소프트웨어 구성 요소가 상기 실행 중인 응용 및 상기 하드웨어 장치 중 적어도 하나의 활동을 상기 미리 정해진 작업 흐름과 비교하는 단계; 그리고

데이터 아이템들 및 할당된 데이터 특성 식별자들이 채널을 통해 상기 불휘발성 메모리 장치에 전송되어 저장되도록 상기 실행 중인 응용 또는 상기 하드웨어 장치에 의해 사용되는 상기 데이터 아이템들에 상기 데이터 특성 식별자들을 자동으로 할당하도록 상기 미리 정해진 작업 흐름을 이용하는 단계를 포함하고,

상기 미리 정해진 작업 흐름은 미리 정해진 데이터 특성 식별자들을 상기 실행 중인 응용 또는 상기 하드웨어 장치에 의해 상기 불휘발성 메모리에 기입되는 상기 데이터 아이템들의 상이한 타입들과 연관시키고,

상기 불휘발성 메모리 장치는 상기 데이터 특성 식별자들을 읽고, 그리고 동일한 데이터 특성 식별자들을 갖는 데이터 아이템들이 동일한 블록에 저장되도록, 상기 데이터 아이템을 저장할 상기 불휘발성 메모리 장치의 블록을 식별하는 방법.

#### 청구항 2

제1 항에 있어서,

상기 하드웨어 장치는 카메라, 마이크, 온도계, GPS, 시계, 센서, 스위치, 라우터, RAID 시스템, 또는 호스트 버스 어댑터(HBA) 시스템 중 적어도 하나를 포함하는 방법.

#### 청구항 3

제1 항에 있어서,

상기 실행 중인 응용 및 하드웨어 장치 중 적어도 하나를 검출하는 것은 새로운 데이터 배치 클라이언트를 검출하는 것을 포함하고, 상기 소프트웨어 구성 요소는 상기 실행 중인 응용 및 상기 하드웨어 장치 중 적어도 하나가 해당 기능을 갖도록 설계되지 않았더라도 데이터 특성 기반 데이터 배치를 제공하는 방법.

#### 청구항 4

제1 항에 있어서,

상기 소프트웨어 구성 요소는 상기 컴퓨터 장치가 상기 불휘발성 메모리 장치의 데이터 특성 기반 데이터 배치를 활용하게 하는 작업 흐름 모니터 인터페이스를 포함하는 방법.

#### 청구항 5

제4 항에 있어서,

상기 작업 흐름 모니터 인터페이스가 상기 작업 흐름 저장소에 대한 새로운 갱신 작업 흐름들을 수신하여 새로운 응용들 및 하드웨어 장치들을 수용하는 단계를 더 포함하는 방법.

#### 청구항 6

제5 항에 있어서,

상기 실행 중인 응용 및 상기 하드웨어 장치 중 적어도 하나의 활동을 상기 미리 정해진 작업 흐름과 비교하는 것은, 상기 실행 중인 응용 및 상기 하드웨어 장치 중 적어도 하나에 의해 수행되는 하나 또는 그보다 많은 현재 프로세스들 또는 단계들을 평가하고 그리고 상기 현재 프로세스들 또는 단계들을 상기 미리 정해진 작업 흐름 내의 유사한 일련의 프로세스들 또는 단계들과 비교하는 것에 의해 수행되는 방법.

#### 청구항 7

불휘발성 메모리 장치; 그리고

상기 불휘발성 메모리 장치에 연결된 컴퓨터 장치를 포함하고,

상기 컴퓨터 장치는 상기 컴퓨터 장치가 상기 불휘발성 메모리 장치 내에서 데이터 특성 기반 데이터 배치를 활용하게 하는 인터페이스로 기능하는 소프트웨어 구성 요소를 실행하고,

상기 소프트웨어 구성 요소는,

실행 중인 응용 및 상기 컴퓨터 장치에 연결된 하드웨어 장치 중 적어도 하나를 검출하는 단계;

상기 실행 중인 응용 및 상기 하드웨어 장치 중 하나를 검출하는 것에 응답하여, 작업 흐름 저장소를 검색하여 상기 실행 중인 응용 및 상기 하드웨어 장치 중 적어도 하나와 연관된 미리 정해진 작업 흐름을 찾는 단계;

상기 소프트웨어 구성 요소가 상기 실행 중인 응용 및 상기 하드웨어 장치 중 적어도 하나의 활동을 상기 미리 정해진 작업 흐름과 비교하는 단계; 그리고

데이터 아이템들 및 할당된 데이터 특성 식별자들이 채널을 통해 상기 불휘발성 메모리 장치에 전송되어 저장되도록 상기 응용 또는 상기 하드웨어 장치에 의해 사용되는 상기 데이터 아이템들에 상기 데이터 특성 식별자들을 자동으로 할당하도록 상기 미리 정해진 작업 흐름을 이용하는 단계를 수행하도록 구성되고,

상기 미리 정해진 작업 흐름은 상기 실행 중인 응용 또는 상기 하드웨어 장치에 의해 상기 불휘발성 메모리 장치에 기입되는 데이터 아이템들의 상이한 타입들에 미리 정해진 데이터 특성 식별자들을 연관하고,

상기 불휘발성 메모리 장치는 상기 데이터 특성 식별자들을 읽고, 그리고 동일한 데이터 특성 식별자들을 갖는 데이터 아이템들이 동일한 블록에 저장되도록, 상기 데이터 아이템을 저장할 상기 불휘발성 메모리 장치의 블록을 식별하는 시스템.

#### 청구항 8

제7 항에 있어서,

상기 하드웨어 장치는 카메라, 마이크, 온도계, GPS, 시계, 센서, 스위치, 라우터, RAID 시스템, 또는 호스트 버스 어댑터(HBA) 시스템 중 적어도 하나를 포함하는 시스템.

#### 청구항 9

제7 항에 있어서,

상기 실행 중인 응용 및 하드웨어 장치 중 적어도 하나를 검출하는 것은 새로운 데이터 배치 클라이언트를 검출하는 것을 포함하고, 상기 소프트웨어 구성 요소는 상기 실행 중인 응용 및 상기 하드웨어 장치 중 적어도 하나가 해당 기능을 갖도록 설계되지 않았더라도 데이터 특성 기반 데이터 배치를 제공하는 시스템.

#### 청구항 10

제7 항에 있어서,

상기 소프트웨어 구성 요소는 상기 컴퓨터 장치가 상기 불휘발성 메모리 장치의 데이터 특성 기반 데이터 배치를 활용하게 하는 작업 흐름 모니터 인터페이스를 포함하는 시스템

#### 청구항 11

제10 항에 있어서,

상기 작업 흐름 모니터 인터페이스가 상기 작업 흐름 저장소에 대한 새로운 갱신 작업 흐름들을 수신하여 새로운 응용들 및 하드웨어 장치들을 수용하는 단계를 더 포함하는 시스템.

#### 청구항 12

제11 항에 있어서,

상기 실행 중인 응용 및 상기 하드웨어 장치 중 적어도 하나의 활동을 상기 미리 정해진 작업 흐름과 비교하는 것은, 상기 실행 중인 응용 및 상기 하드웨어 장치 중 적어도 하나에 의해 수행되는 하나 또는 그보다 많은 현재 프로세스들 또는 단계들을 평가하고 그리고 상기 현재 프로세스들 또는 단계들을 상기 미리 정해진 작업 흐름 내의 유사한 일련의 프로세스들 또는 단계들과 비교하는 것에 의해 수행되는 시스템.

#### 청구항 13

메모리 어레이; 그리고

상기 메모리 어레이 및 메모리 채널에 연결되는 스토리지 컨트롤러를 포함하고,

상기 스토리지 컨트롤러는,

운영 체제 및 실행 중인 응용 중 적어도 하나로부터 상기 메모리 채널을 통해 저장될 제1 시리즈의 데이터 아이템들을 수신하고;

상기 제1 시리즈의 데이터 아이템들 각각은 데이터 유사도를 가리키는 하나 또는 그보다 많은 데이터 특성들에 기반하여 상기 제1 시리즈의 데이터 아이템들과 연관되는 제1 데이터 특성 식별자를 포함하고,

하드웨어 장치로부터 상기 메모리 채널을 통해 저장될 제2 시리즈의 데이터 아이템들을 수신하고;

상기 제2 시리즈의 데이터 아이템들 각각은 데이터 유사도를 가리키는 하나 또는 그보다 많은 데이터 특성들에 기반하여 상기 제2 시리즈의 데이터 아이템들과 연관되는 제2 데이터 특성 식별자를 포함하고,

상기 제1 및 제2 데이터 특성 식별자들을 읽고, 동일한 데이터 특성 식별자들을 갖는 데이터 아이템들이 동일한 블록에 저장되도록, 상기 제1 및 제2 시리즈의 데이터 아이템들을 저장할 블록들을 식별하고; 그리고

상기 제1 및 제2 시리즈의 데이터 아이템들을 상기 식별된 블록들에 저장하도록 구성되는 불휘발성 메모리 장치.

#### 청구항 14

제13 항에 있어서,

상기 하드웨어 장치는 카메라, 마이크, 온도계, GPS, 시계, 센서, 스위치, 라우터, RAID 시스템, 또는 호스트 버스 어댑터(HBA) 시스템 중 적어도 하나를 포함하는 불휘발성 메모리 장치.

#### 청구항 15

제13 항에 있어서,

상기 운영 체제 및 상기 실행 중인 응용 중 적어도 하나 또는 상기 하드웨어 장치는 컴퓨터 장치의 일부인 불휘발성 메모리 장치.

#### 청구항 16

제15 항에 있어서,

상기 컴퓨터 장치는 상기 컴퓨터 장치가 상기 불휘발성 메모리 장치의 데이터 특성 기반 데이터 배치를 활용하여 하는 작업 흐름 모니터 인터페이스를 포함하는 불휘발성 메모리 장치.

#### 청구항 17

제16 항에 있어서,

상기 작업 흐름 모니터 인터페이스는 상기 컴퓨터 장치에서 운영 체제 레벨에서 실행되고,

상기 작업 흐름 모니터 인터페이스는,

상기 실행 중인 응용 및 상기 컴퓨터 장치에 연결된 상기 하드웨어 장치 중 적어도 하나를 검출하는 단계;

상기 실행 중인 응용 및 상기 하드웨어 장치 중 적어도 하나를 검출하는 것에 응답하여, 작업 흐름 저장소를 검색하여 상기 실행 중인 응용 및 상기 하드웨어 장치 중 적어도 하나와 연관된 미리 정해진 작업 흐름을 찾는 단계;

상기 실행 중인 응용 및 상기 하드웨어 장치 중 적어도 하나의 활동을 상기 미리 정해진 작업 흐름과 비교하는 단계; 그리고

데이터 아이템들 및 할당된 데이터 특성 식별자들이 채널을 통해 상기 불휘발성 메모리 장치에 전송되어 저장되도록 상기 실행 중인 응용 또는 상기 하드웨어 장치에 의해 사용되는 상기 데이터 아이템들에 상기 데이터 특성 식별자들을 자동으로 할당하도록 상기 미리 정해진 작업 흐름을 이용하는 단계를 수행하도록 구성되고,

상기 미리 정해진 작업 흐름은 미리 정해진 데이터 특성 식별자들을 상기 실행 중인 응용 또는 상기 하드웨어 장치에 의해 상기 불휘발성 메모리에 기입되는 상기 데이터 아이템들의 상이한 타입들과 연관하는 불휘발성 메모리 장치.

#### 청구항 18

불휘발성 메모리 장치의 스토리지 컨트롤러에 의해 수행되는 상기 불휘발성 메모리 장치 내부의 데이터 특성 기반 데이터 배치를 위한 방법에 있어서:

운영 체제 및 실행 중인 응용 중 적어도 하나로부터 메모리 채널을 통해 저장될 제1 시리즈의 데이터 아이템들을 수신하는 단계;

상기 제1 시리즈의 데이터 아이템들 각각은 데이터 유사도를 가리키는 하나 또는 그보다 많은 데이터 아이템들에 기반하여 연관되는 제1 데이터 특성 식별자를 포함하고,

하드웨어 장치로부터 상기 메모리 채널을 통해 저장될 제2 시리즈의 데이터 아이템들을 수신하는 단계;

상기 제2 시리즈의 데이터 아이템들 각각은 데이터 유사도를 가리키는 하나 또는 그보다 많은 데이터 아이템들에 기반하여 연관되는 제2 데이터 특성 식별자를 포함하고,

상기 제1 및 제2 데이터 특성 식별자들을 읽고, 동일한 데이터 특성 식별자를 갖는 데이터 아이템들이 동일한 블록에 저장되도록 상기 제1 및 제2 시리즈의 데이터 아이템들을 저장할 상기 불휘발성 메모리 장치의 블록들을 식별하는 단계; 그리고

상기 제1 및 제2 시리즈의 데이터 아이템들을 상기 식별된 블록들에 저장하는 단계를 포함하는 방법.

#### 청구항 19

제18 항에 있어서,

상기 하드웨어 장치는 카메라, 마이크, 온도계, GPS, 시계, 센서, 스위치, 라우터, RAID 시스템, 또는 호스트 버스 어댑터(HBA) 시스템 중 적어도 하나를 포함하는 방법.

#### 청구항 20

컴퓨터 장치가 불휘발성 메모리 장치 내에서 데이터 특성 기반 데이터 배치를 활용하게 하는 인터페이스를 제공하는 방법에 있어서:

상기 컴퓨터 장치에서 소프트웨어 구성 요소를 실행하는 단계;

상기 소프트웨어 구성 요소가 상기 소프트웨어 구성 요소와 연관된 미리 정해진 작업 흐름을 찾도록 작업 흐름 저장소에 요청하는 단계;

상기 미리 정해진 작업 흐름은 상기 불휘발성 메모리 장치에 기입될 데이터 아이템들의 상이한 특성들에 미리 정해진 데이터 특성 식별자들을 연관하고,

상기 소프트웨어 구성 요소의 동작 동안에, 상기 소프트웨어 구성 요소에 의해 사용되는 데이터 아이템들과 연관되는 하나 또는 그보다 많은 데이터 특성 식별자들을 위해 상기 미리 정해진 작업 흐름에 하나 또는 그보다

많은 요청들을 전송하는 단계; 그리고

상기 소프트웨어 구성 요소가 상기 하나 또는 그보다 많은 데이터 특성 식별자들을 상기 미리 정해진 작업 흐름 으로부터 수신하는 것에 응답하여, 상기 소프트웨어 구성 요소에 의해 사용되는 상기 데이터 아이템들에 상기 데이터 특성 식별자들을 자동으로 할당하고 그리고 상기 데이터 아이템들 및 상기 할당된 데이터 특성 식별자들을 저장하기 위해 상기 불휘발성 메모리 장치에 전송하는 단계를 포함하고,

상기 불휘발성 메모리 장치는 상기 데이터 특성 식별자들을 읽고, 동일한 데이터 특성 식별자들을 갖는 데이터 아이템들이 동일한 블록에 저장되도록, 상기 데이터 아이템들이 저장될 상기 불휘발성 메모리 장치의 블록들을 식별하는 방법.

## 발명의 설명

### 기술 분야

[0001] 본 발명은 반도체 메모리에 관한 것으로, 더 상세하게는 컴퓨터 장치가 불휘발성 메모리 장치 내에서 데이터 특성 기반 데이터 배치를 활용할 수 있게 하는 인터페이스를 제공하는 방법, 불휘발성 메모리 장치 내에서의 데이터 특성 기반 데이터 배치를 위한 방법, 그리고 불휘발성 메모리 장치에 관한 것이다.

### 배경 기술

[0002] 플래시 메모리 기반의 솔리드 스테이트 드라이브(SSD)는 소비자 컴퓨터들 및 기업 서버들 모두에서 널리 사용되어 왔다. 플래시 메모리는 낸드(NAND) 및 노어(NOR) 논리 게이트들을 본따 명명된 두 가지 타입을 갖는다. NAND 타입 플래시 메모리는 블록들에서 기입되고 읽히지며, 각 블록은 복수의 페이지들을 포함한다.

[0003] SSD의 NAND 플래시 스토리지 셀들은 매우 고유한 특성들 가지므로, SSD의 통상적인 사용은 매우 비효율적이다. 예를 들어, 한 번에 하나의 페이지에 대해 읽기 및 프로그램이 수행될 수 있지만, NAND 플래시 메모리는 한 번에 하나의 블록에서 소거를 수행할 수 있다. 하나의 NAND 플래시 페이지를 재기입하기 위해, 복수의 플래시 페이지들을 포함하는 전체 소거 블록이 먼저 소거되어야 한다.

[0004] SSD와 같은 NAND 플래시 기반 스토리지 장치들은 제자리(in-placing) 갱신을 허용하지 않으므로, 가용한 자유 블록 카운트가 특정한 문턱에 도달할 때 후속 쓰기들을 위한 공간을 확보하기 위하여 가비지 컬렉션 동작이 수행된다. 가비지 컬렉션은 목표 소거 블록으로부터 유효 데이터를 읽고 유효 데이터를 다른 블록에 기입하는 것을 포함하며, 무효 데이터는 새로운 블록으로 전송되지 않는다. NAND 소거 블록을 소거하는 것은 상대적으로 많은 시간을 필요로 하며, 각 소거 블록은 3천번 내지 1만번 사이의 제한된 소거 횟수를 갖는다. 따라서, 가비지 컬렉션 오버헤드는 이 기술 분야에서 가장 큰 속도 제약 중 하나이며, 높은 데이터 입출력 레이턴시 및 낮은 입출력 성능을 유발한다.

[0005] 따라서, 핫 데이터 및 콜드 데이터를 다르게 취급하지 않으며 함께 저장하는 운영 체제(OS) 및 응용들은, 핫 데이터 및 콜드 데이터를 다르게 취급하는 운영 체제 및 응용들과 비교하여, 시간의 흐름에 따라 성능이 열화되고, 더 많은 소거 횟수가 필요함에 따라 NAND 셀들이 더 빨리 마모되므로 SSD 수명이 단축된다.

[0006] SSD 제조사들 및 스토리지 기술 위원회들은 다중 스트림이라 불리는 새로운 SSD 및 표준을 제안하였으며, 이에 따르면 호스트는 데이터를 스트림이라 불리는 클래스로 그룹화할 수 있다. 각 스트림은 SSD에 의해 독립적으로 저장(또는 세이브)된다. 스트림들의 데이터는 이론상 어떠한 이유로든 그룹될 수 있다. 이러한 이유 중 하나는 가비지 컬렉션 이슈를 줄이기 위하여 유사한 수명을 갖는 데이터를 함께 저장하는 것을 목적으로 한다.

[0007] 그러나, 이러한 새로운 인터페이스를 활용하기 위하여, 응용들(소스 코드를 포함) 및 운영 체제 내에서 많은 변화가 필요하다. 통상적인 컴퓨터는 수십 또는 수백개의 설치되고 실행되는 소프트웨어 응용들을 가질 수 있으므로, SSD를 더 효율적으로 활용하기 위하여 모든 응용들 특히 레거시 및 비공개 소스 응용들이 이러한 변화를 채택하도록 하는 것은 매우 어렵다. 또한, 다중 스트림 SSD는 현재 다중 스트림 SSD를 사용할 줄 아는 운영 체제 및 응용들에 의해서만 사용될 수 있다는 제한된 적용성을 갖는다.

[0008] 현재 필요한 것은 스토리지 장치의 향상된 데이터 특성 기반 데이터 배치이며, 더 상세하게는 컴퓨터 장치가 데이터 특성 기반 데이터 배치(예를 들어, 다중 스트림) SSD를 활용할 수 있게 하는 자동 프로세스이다.

### 발명의 내용

## 해결하려는 과제

[0009] 본 발명의 목적은 기존의 장치들 또는 응용들에서 데이터 특성 기반 데이터 배치를 활용할 수 있게 하는 장치 및 방법을 제공하는 데에 있다.

## 과제의 해결 수단

[0010] 본 발명은 컴퓨터 장치가 불휘발성 메모리 장치 내에서 데이터 특성 기반 데이터 배치를 활용하게 하는 인터페이스를 제공하는 방법에 관한 것이다. 본 발명의 방법은, 실행 중인 응용 및 상기 컴퓨터 장치에 연결된 하드웨어 장치 중 적어도 하나를 검출하는 소프트웨어 구성 요소를 상기 컴퓨터 장치에서 실행하는 단계, 상기 실행 중인 응용 및 상기 하드웨어 장치 중 적어도 하나를 검출하는 것에 응답하여, 상기 소프트웨어 구성 요소가 작업 흐름 저장소를 검색하여 상기 실행 중인 응용 및 상기 하드웨어 장치 중 적어도 하나와 연관된 미리 정해진 작업 흐름을 찾는 단계, 상기 소프트웨어 구성 요소가 상기 실행 중인 응용 및 상기 하드웨어 장치 중 적어도 하나의 활동을 상기 미리 정해진 작업 흐름과 비교하는 단계, 그리고 데이터 아이템들 및 할당된 데이터 특성 식별자들이 채널을 통해 상기 불휘발성 메모리 장치에 전송되어 저장되도록 상기 실행 중인 응용 또는 상기 하드웨어 장치에 의해 사용되는 상기 데이터 아이템들에 상기 데이터 특성 식별자들을 자동으로 할당하도록 상기 미리 정해진 작업 흐름을 이용하는 단계를 포함한다. 상기 미리 정해진 작업 흐름은 미리 정해진 데이터 특성 식별자들을 상기 실행 중인 응용 또는 상기 하드웨어 장치에 의해 상기 불휘발성 메모리에 기입되는 상기 데이터 아이템들의 상이한 타입들과 연관한다. 상기 불휘발성 메모리 장치는 상기 데이터 특성 식별자들을 읽고, 그리고 동일한 데이터 특성 식별자들을 갖는 데이터 아이템들이 동일한 블록에 저장되도록, 상기 데이터 아이템을 저장할 상기 불휘발성 메모리 장치의 블록을 식별한다.

[0011] 본 발명은 시스템에 관한 것이다. 본 발명의 시스템은 불휘발성 메모리 장치, 그리고 상기 불휘발성 메모리 장치에 연결된 컴퓨터 장치를 포함한다. 상기 컴퓨터 장치는 상기 컴퓨터 장치가 상기 불휘발성 메모리 장치 내에서 데이터 특성 기반 데이터 배치를 활용하게 하는 인터페이스로 기능하는 소프트웨어 구성 요소를 실행한다. 상기 소프트웨어 구성 요소는, 실행 중인 응용 및 상기 컴퓨터 장치에 연결된 하드웨어 장치 중 적어도 하나를 검출하는 단계, 상기 실행 중인 응용 및 상기 하드웨어 장치 중 하나를 검출하는 것에 응답하여, 작업 흐름 저장소를 검색하여 상기 실행 중인 응용 및 상기 하드웨어 장치 중 적어도 하나와 연관된 미리 정해진 작업 흐름을 찾는 단계, 상기 소프트웨어 구성 요소가 상기 실행 중인 응용 및 상기 하드웨어 장치 중 적어도 하나의 활동을 상기 미리 정해진 작업 흐름과 비교하는 단계, 그리고 데이터 아이템들 및 할당된 데이터 특성 식별자들이 채널을 통해 상기 불휘발성 메모리 장치에 전송되어 저장되도록 상기 응용 또는 상기 하드웨어 장치에 의해 사용되는 상기 데이터 아이템들에 상기 데이터 특성 식별자들을 자동으로 할당하도록 상기 미리 정해진 작업 흐름을 이용하는 단계를 수행하도록 구성된다. 상기 미리 정해진 작업 흐름은 상기 실행 중인 응용 또는 상기 하드웨어 장치에 의해 상기 불휘발성 메모리 장치에 기입되는 데이터 아이템들의 상이한 타입들에 미리 정해진 데이터 특성 식별자들을 연관한다. 상기 불휘발성 메모리 장치는 상기 데이터 특성 식별자들을 읽고, 그리고 동일한 데이터 특성 식별자들을 갖는 데이터 아이템들이 동일한 블록에 저장되도록, 상기 데이터 아이템을 저장할 상기 불휘발성 메모리 장치의 블록을 식별한다.

[0012] 본 발명은 불휘발성 메모리 장치에 관한 것이다. 본 발명의 불휘발성 메모리 장치는, 메모리 어레이, 그리고 상기 메모리 어레이 및 채널에 연결되는 스토리지 컨트롤러를 포함한다. 상기 스토리지 컨트롤러는, 운영 체제 및 실행 중인 응용 중 적어도 하나로부터 상기 메모리 채널을 통해 저장될 제1 시리즈의 데이터 아이템들을 수신하고, 상기 제1 시리즈의 데이터 아이템들 각각은 데이터 유사도를 가리키는 하나 또는 그보다 많은 데이터 특성들에 기반하여 상기 제1 시리즈의 데이터 아이템들과 연관되는 제1 데이터 특성 식별자를 포함하고, 하드웨어 장치 컨트롤러로부터 상기 메모리 채널을 통해 저장될 제2 시리즈의 데이터 아이템들을 수신하고, 상기 제2 시리즈의 데이터 아이템들 각각은 데이터 유사도를 가리키는 하나 또는 그보다 많은 데이터 특성들에 기반하여 상기 제2 시리즈의 데이터 아이템들과 연관되는 제2 데이터 특성 식별자를 포함하고, 상기 제1 및 제2 데이터 특성 식별자들을 읽고, 동일한 데이터 특성 식별자들을 갖는 데이터 아이템들이 동일한 블록에 저장되도록, 상기 제1 및 제2 시리즈의 데이터 아이템들을 저장할 블록들을 식별하고, 그리고 상기 제1 및 제2 시리즈의 데이터 아이템들을 상기 식별된 블록들에 저장하도록 구성된다.

[0013] 본 발명은 불휘발성 메모리 장치의 스토리지 컨트롤러에 의해 수행되는 상기 불휘발성 메모리 장치 내부의 데이터 특성 기반 데이터 배치를 위한 방법에 관한 것이다. 본 발명의 방법은, 운영 체제 및 실행 중인 응용 중 적어도 하나로부터 메모리 채널을 통해 저장될 제1 시리즈의 데이터 아이템들을 수신하는 단계, 상기 제1 시리즈



의 데이터 아이템들 각각은 데이터 유사도를 가리키는 하나 또는 그보다 많은 데이터 아이템들에 기반하여 연관되는 제1 데이터 특성 식별자를 포함하고, 하드웨어 장치 컨트롤러로부터 상기 메모리 채널을 통해 저장될 제2 시리즈의 데이터 아이템들을 수신하는 단계, 상기 제2 시리즈의 데이터 아이템들 각각은 데이터 유사도를 가리키는 하나 또는 그보다 많은 데이터 아이템들에 기반하여 연관되는 제2 데이터 특성 식별자를 포함하고, 상기 제1 및 제2 데이터 특성 식별자들을 읽고, 동일한 데이터 특성 식별자를 갖는 데이터 아이템들이 동일한 블록에 저장되도록 상기 제1 및 제2 시리즈의 데이터 아이템들을 저장할 상기 불휘발성 메모리 장치의 블록들을 식별하는 단계, 그리고 상기 제1 및 제2 시리즈의 데이터 아이템들을 상기 식별된 블록들에 저장하는 단계를 포함한다.

[0014] 본 발명은 컴퓨터 장치가 불휘발성 메모리 장치 내에서 데이터 특성 기반 데이터 배치를 활용하게 하는 인터페이스를 제공하는 방법에 관한 것이다. 본 발명의 방법은, 상기 컴퓨터 장치에서 소프트웨어 구성 요소를 실행하는 단계, 상기 소프트웨어 구성 요소가 상기 소프트웨어 구성 요소와 연관된 미리 정해진 작업 흐름을 찾도록 작업 흐름 저장소에 요청하는 단계, 상기 미리 정해진 작업 흐름은 상기 불휘발성 메모리 장치에 기입될 데이터 아이템들의 상이한 특성들에 미리 정해진 데이터 특성 식별자들을 연관하고, 상기 소프트웨어 구성 요소의 동작 동안에, 상기 소프트웨어 구성 요소에 의해 사용되는 데이터 아이템들과 연관되는 하나 또는 그보다 많은 데이터 특성 식별자들을 위해 상기 미리 정해진 작업 흐름에 하나 또는 그보다 많은 요청들을 전송하는 단계, 그리고 상기 소프트웨어 구성 요소가 상기 하나 또는 그보다 많은 데이터 특성 식별자들을 상기 미리 정해진 작업 흐름으로부터 수신하는 것에 응답하여, 상기 소프트웨어 구성 요소에 의해 사용되는 상기 데이터 아이템들에 상기 데이터 특성 식별자들을 자동으로 할당하고 그리고 상기 데이터 아이템들 및 상기 할당된 데이터 특성 식별자들을 저장하기 위해 상기 불휘발성 메모리 장치에 전송하는 단계를 포함한다. 상기 불휘발성 메모리 장치는 상기 데이터 특성 식별자들을 읽고, 동일한 데이터 특성 식별자들을 갖는 데이터 아이템들이 동일한 블록에 저장되도록, 상기 데이터 아이템들이 저장될 상기 불휘발성 메모리 장치의 블록들을 식별한다.

### 발명의 효과

[0015] 본 발명에 따르면, 기존의 장치, 응용 또는 운영 체제를 일부 수정함으로써 데이터 특성 기반 데이터 배치가 활용될 수 있다. 따라서, 기존의 장치, 응용 또는 운영 체제의 데이터 관리 효율성이 향상된다.

### 도면의 간단한 설명

[0016] 본 발명의 특성 및 활용이 후술되는 실시 예들의 설명 및 첨부된 도면들을 참조하여 더 명백하게 이해될 것이다.

도 1은 작업 흐름 모니터 인터페이스에 의해 제공되는 불휘발성 메모리 장치의 데이터 특성 기반 데이터 배치를 위한 시스템의 실시 예를 보여주는 블록도이다.

도 2는 본 발명의 실시 예에 따라 컴퓨터 장치가 불휘발성 메모리 장치 내의 데이터 특성 기반 데이터 배치를 활용할 수 있게 하기 위한 작업 흐름 모니터 인터페이스를 제공하기 위한 호스트 시스템에서 수행되는 프로세스를 보여주는 순서도이다.

도 3은 데이터 특성 기반 데이터 배치를 구비한 SSD와 호환되는 다양한 컴퓨팅 장치들로 구현된 실시 예들의 작업 흐름 모니터 인터페이스를 보여주는 다이어그램이다.

도 4는 본 발명의 실시 예에 따라 불휘발성 메모리 장치의 스토리지 컨트롤러에 의해 수행되는 데이터 특성 기반 데이터 배치를 위한 프로세스를 보여주는 순서도이다.

도 5는 상이한 수명 및 상이한 특성들을 갖는 데이터를 저장하는 데이터 베이스 응용에 의해 수행되는 스토리지 동작들에 대한 작업 흐름 모니터 인터페이스의 동작의 예를 보여주는 블록도이다.

도 6은 작업 흐름 모니터 인터페이스를 더 상세하게 보여주는 블록도이다.

도 7은 데이터 식별자 할당 방법의 다른 예를 보여주는 블록도이다.

### 발명을 실시하기 위한 구체적인 내용

[0017] 첨부된 도면들을 참조하여 본 발명의 기술적 사상의 실시 예들이 상세하게 설명된다. 유사한 참조 번호들은 유사한 구성 요소들을 참조할 수 있다.

- [0018] 본 발명의 장점들 및 특징들, 그리고 그것을 달성하는 방법들은 상세한 설명 및 첨부된 도면들을 참조하여 더 용이하게 이해될 수 있다. 그러나, 본 발명의 기술적 사상은 다양한 다른 형태들로 구현될 수 있으며, 여기에 개시되는 실시 예들로 한정되지 않는 것으로 이해되어야 한다. 실시 예들은 이하의 기재가 철저하고 완전하며, 본 발명의 기술적 사상을 이 분야에 통상적인 기술을 가진 자들에게 전달할 수 있도록 제공되는 것에 불과하며, 본 발명의 기술적 사상은 첨부된 청구범위에 의해서만 정의된다. 도면들에서, 계층들 및 영역들의 두께는 명확성을 위해 강조될 수 있다.
- [0019] 별도로 언급되지 않으면, "포함하다", "구비하다", "갖는다"의 용어들은 개방적인 용어들(즉, 한정되지 않는)것으로 이해되어야 한다.
- [0020] 여기에 사용되는 "알고리즘" 또는 "모듈"의 용어는 특정한 작업들을 수행하는 FPGA (field programmable gate array) 또는 ASIC (an application specific integrated circuit)와 같은 소프트웨어 또는 하드웨어 구성 요소일 수 있으며, 한정되지 않는다. 알고리즘 또는 모듈은 어드레스로 식별 가능한 스토리지 매체에 거주하도록 구성될 수 있으며, 하나 또는 그보다 많은 프로세스들을 실행하도록 구성될 수 있다. 따라서, 알고리즘 또는 모듈은, 예시적으로, 소프트웨어 구성 요소, 객체 지향 소프트웨어 구성 요소, 클래스 구성 요소, 그리고 작업 구성 요소와 같은 구성 요소(component), 프로세스들, 기능들, 속성들(attributes), 프로시저들, 서브루틴들, 프로그램 코드의 세그먼트들, 펌웨어, 마이크로코드, 회로망, 데이터, 데이터베이스, 데이터 구조, 테이블들, 어레이들, 그리고 변수들을 포함할 수 있다. 구성 요소들 및 구성 요소들 또는 모듈들에 제공되는 기능들은 더 적은 구성 요소들 또는 모듈들로 결합되거나, 또는 추가적인 구성 요소들 및 구성 요소들 또는 모듈들로 더 분리될 수 있다.
- [0021] 별도로 정의되지 않으면, 여기에서 사용되는 모든 기술적 과학적 용어들은 이 발명이 속하는 분야에서 통상적인 기술을 가진 자들에 의해 공통적으로 이해되는 것과 동일한 의미를 갖는다. 어느 또는 모든 예들 또는 예시적인 용어들의 사용은 본 발명을 더 명확하게 조명하기 위해 의도된 것이며, 별도로 명시되지 않는 한 본 발명의 범위를 한정하는 용도로 사용되지 않는다. 또한, 별도로 정의되지 않으면, 일반적으로 사용되는 사전들에서 정의된 모든 용어들은 과도하게 해석되지 않아야 한다.
- [0022] 예시적으로, 본 발명의 실시 예들은, 응용들 및 장치들이 다중 스트리밍을 활용하도록 프로그램되지 않더라도, 컴퓨터 장치가 불휘발성 메모리 장치 내에서 데이터 특성 기반 데이터 배치(예를 들어, 다중 스트리밍)을 활용하는 것을 가능하게 하기 위한 인터페이스를 제공한다. 구체적으로, 본 발명의 실시 예들은, 컴퓨터 장치의 활동을 모니터링하고 그리고 실행중인 응용들 또는 연결된 장치들의 작업 흐름의 지식을 이용하여 미리 정해진 데이터 특성 식별자들을 불휘발성 메모리 장치에 세이브(또는 기입)되는 데이터 아이템들에 할당하는 응용 모델 기반 스트림 식별자 프로세스를 제공한다. 다중 스트리밍과 동일하게, 수신자 메모리 장치는 동일한 데이터 특성 식별자를 갖는 모든 데이터가 동시에(거의 동시에) 무효화될 것으로 가정하고, 이러한 데이터를 공통 소거 블록들에 배치(또는 기입)할 수 있다.
- [0023] 도 1은 작업 흐름 모니터링 인터페이스에 의해 제공되는 불휘발성 메모리 장치의 데이터 특성 기반 데이터 배치를 위한 실시 예를 보여주는 블록도이다. 여기에 설명되는 실시 예들은 가비지 컬렉션을 필요로 하는 어느 불휘발성 메모리 장치에도 적용될 수 있으며, 불휘발성 메모리 장치가 솔리드 스테이트 드라이브(SSD)를 포함하는 실시 예를 참조하여 설명된다.
- [0024] 시스템은 채널(14)을 통해 SSD (12)에 결합된 호스트 시스템(10)을 포함한다. 여기에서 사용되는 것과 같이, 채널은 불휘발성 메모리 장치(예를 들어, SSD)를 컴퓨터 또는 SSD를 사용하는 다른 하드웨어에 연결하는 임의의 물리 인터페이스를 나타낼 수 있다. 인터페이스는 SATA, SCSI, PCIe, USB, iSCSI, FC, 이더넷 등과 같은 유선 인터페이스, 그리고 WiFi와 같은 무선 인터페이스를 포함할 수 있다. 잘 알려진 바와 같이, SSD는 데이터를 저장하는 이동식 부품을 갖지 않으며, 데이터를 유지하는 데에 일정한 전력을 필요로 하지 않는다. 본 발명과 연관된 호스트 시스템(10)의 구성 요소들은, 운영 체제(OS, 20) 및 파일 시스템(21)을 포함하는 메모리(18)로부터의 컴퓨터 명령들을 실행하는 프로세서(16)를 포함한다. 호스트 시스템(10)은 채널(14)과 인터페이싱하기 위한 메모리 컨트롤러와 같은 미도시된 다른 구성 요소들을 더 포함할 수 있다. 호스트 시스템(10) 및 SSD (12)는 채널(14)을 통해 커맨드들 및 데이터 아이템들(28)을 통신할 수 있다. 예시적으로, 호스트 시스템(10)은 임의의 타입의 운영 체제를 실행하는 통상적인 컴퓨터 또는 서버일 수 있다. 운영 체제의 타입들은 단일 및 다중 사용자, 분산, 템플릿(templated), 임베디드, 실시간 및 라이브러리를 포함할 수 있다. 다른 예로서, 호스트 시스템(10)은 장치 제어기와 같은 스탠드얼론 구성 요소일 수 있으며, 이 경우에 운영 체제는 경량 운영 체제(또는 그들의 일부) 또는 펌웨어일 수 있다.

- [0025] SSD (12)는 스토리지 컨트롤러(22) 및 호스트 시스템(10)으로부터의 데이터를 저장하는 불휘발성 메모리 어레이(24)를 포함한다. 스토리지 컨트롤러(22)는 불휘발성 메모리 어레이(24)에 저장된 데이터를 관리하고, 통신 프로토콜을 통해 채널(14)을 거쳐 호스트 시스템(10)과 통신한다. 불휘발성 메모리 어레이(24)는 플래시 메모리, 강유전체 랜덤 액세스 메모리(F-RAM), 자기저항 랜덤 액세스 메모리(MRAM), 상 변화 메모리(PCM), 노래기(millipede) 메모리 등과 같은 불휘발성 랜덤 액세스 메모리(NVRAM)를 포함할 수 있다. SSD (12) 및 채널(14) 모두는 듀얼 채널 구조와 같은 다중 채널 메모리 구조를 지원하며, 단일, 2배속 또는 4배속 데이터 전송을 지원할 수 있다.
- [0026] 본 발명의 실시 예들에 따르면, SSD (12)에서 가비지 컬렉션 오버헤드를 감소시키기 위하여, SSD (120)에 향상된 데이터 특성 기반 데이터 배치가 제공된다. 이는, 응용들 및 하드웨어 구성 요소들 모두가 상이한 수명들을 갖는 데이터 아이템들을 SSD (12)에 개별적으로 저장하는 것을 가능하게 하는 작업 흐름 모니터 인터페이스(26)를 제공함으로써 달성된다. 또한, 작업 흐름 모니터 인터페이스(26)의 사용은 호스트 시스템(10)에서 실행되는 사용자 응용들의 변화를 요구하지 않는다.
- [0027] 예시적으로, 작업 흐름 모니터 인터페이스(26)는 프로세서(16)에 의해 실행되며 호스트 시스템(10)이 응용들 또는 연결된 하드웨어 장치들을 실행하는 것을 지속적으로 모니터링하는 적어도 하나의 소프트웨어 구성 요소를 포함한다. 작업 흐름 모니터 인터페이스(26)는 응용들 및 연결된 장치들에 대한 미리 정해진 작업 흐름 패턴들의 지식을 구비하며, 이러한 작업 흐름 패턴에 기반하여 응용들 및 연결된 장치들에 의해 기입되는 데이터 아이템들에 데이터 특성 식별자들(예를 들어, 스트림 ID)을 할당한다.
- [0028] 본 발명의 실시 예들의 일 측면에 따르면, 작업 흐름 모니터 인터페이스(26)는 다양한 응용들 및 장치들에 대한 알려진 작업 흐름들의 라이브러리로 작용하는 작업 흐름 저장소(27)를 포함할 수 있다. 작업 흐름 저장소(27) 내의 작업 흐름들 각각은 작업 흐름의 다양한 단계들 동안에 주어진 특성 식별자(예를 들어, 스트림 식별자)를 갖고 생성되는 하나 또는 그보다 많은 스트림들(즉, 하나 또는 그보다 많은 개별 데이터 쓰기들)을 포함할 수 있다. 예시적으로, 작업 흐름 저장소(27)는 작업 흐름 모니터 인터페이스(26)로부터 개별적으로 분리된 구성 요소일 수 있다. 그러나, 다른 예로서, 작업 흐름 저장소(27)는 작업 흐름 모니터 인터페이스(26)와 집적될 수 있다. 예시적으로, 데이터 특성 식별자(30)의 실제 할당은 운영 체제(20)의 파일 시스템(21) 내의 소프트웨어 후크(hook)에 의해 수행될 수 있다.
- [0029] 도 2는 본 발명의 실시 예에 따라 컴퓨터 장치가 불휘발성 메모리 장치 내의 데이터 특성 기반 데이터 배치를 활용할 수 있게 하기 위한 작업 흐름 모니터 인터페이스를 제공하기 위한 호스트 시스템에서 수행되는 프로세스를 보여주는 순서도이다. 프로세스는 실행 중인 응용 및 컴퓨팅 장치와 연결 중인 하드웨어 장치 중 적어도 하나를 검출하는 소프트웨어 구성 요소(예를 들어, 작업 흐름 모니터 인터페이스(26))를 컴퓨팅 장치(10)에서 실행하는 것(200)을 포함할 수 있다. 예시적으로, 실행 중인 응용들을 검출하는 것은 프로세스들 또는 스레드들을 검출하는 것을 포함할 수 있다. 하드웨어 장치 연결의 예는 호스트 장치(10)에 연결되고 통신하는 카메라를 포함할 수 있다.
- [0030] 예시적으로, 실행 중인 응용 또는 하드웨어 장치를 검출하는 것은 전체적으로 소프트웨어 구성 요소가 데이터 특성 기반 데이터 배치를 제공(예를 들어, 응용 또는 하드웨어 장치가 다중 스트리밍 능력에 호환되게 설계되지 않더라도 다중 스트리밍 능력을 제공)하는 새로운 데이터 배치 클라이언트를 검출하는 것으로 보여질 수 있다.
- [0031] 실행 중인 응용 또는 연결 중인 하드웨어 장치 연결을 검출하는 것에 응답하여, 소프트웨어 구성 요소는 작업 흐름 저장소(27)를 검색하여 실행 중인 응용 또는 하드웨어 장치와 연관된 미리 정해진 작업 흐름을 찾을 수 있다. 미리 정해진 작업 흐름은 실행 중인 응용 또는 하드웨어 장치에 의해 불휘발성 메모리 장치에 기입되는 상이한 타입의 데이터 아이템들에 데이터 특성 식별자들을 연관시킬 수 있다(202).
- [0032] 예시적으로, 작업 흐름 저장소(27)는 작업 흐름 모니터 인터페이스(26)와 집적되거나 또는 개별 구성 요소로 구현될 수 있다. 예시적으로, 작업 흐름 저장소(27)는 컴퓨팅 장치(10) 내부에 위치할 수 있으며, 다른 예로서, 작업 흐름 저장소(27)는 컴퓨팅 장치(10)로부터 떨어져 위치할 수 있다. 어느 경우에도, 실시 예로서, 작업 흐름 모니터 인터페이스(26)는 새로운 응용들 및 하드웨어 장치들을 수용하기 위하여 작업 흐름 저장소(27)의 새로운 또는 갱신 작업 흐름들을 수신할 수 있다.
- [0033] 소프트웨어 구성 요소는 실행 중인 응용 또는 하드웨어 장치의 활동들을 미리 정해진 작업 흐름과 비교할 수 있다(204). 예시적으로, 활동들을 비교하는 것은, 응용 또는 하드웨어 장치에 의해 수행되는 하나 또는 그보다 많은 현재 프로세스들 또는 스레드들을 조사(examine)하고, 현재 프로세스들 또는 스레드들을 미리 정해진 작업 흐름

내의 프로세스들 또는 스텝들의 유사한 시퀀스와 맞추는(match) 것을 포함할 수 있다. 상술된 바와 같이, 작업 흐름은 작업 흐름의 다양한 단계들 동안 프로세스들 또는 스텝들의 시퀀스 동안에 수행되는 하나 또는 그보다 많은 개별 데이터 쓰기들을 주어진 데이터 특성 식별자와 연관시킨다.

[0034] 소프트웨어 구성 요소는 미리 정해진 작업 흐름을 이용하여, 데이터 특성 식별자들을 실행 중인 응용 또는 하드웨어 장치에 의해 사용되는 데이터 아이템들에 자동으로 할당하며, 데이터 아이템들 및 할당된 데이터 특성 식별자들은 저장을 위해 불휘발성 메모리 어레이(24)로 전송된다(206).

[0035] 본 발명의 일 실시 예에 따르면, 작업 흐름 모니터 인터페이스(26)는 작업 흐름 저장소(27)를 이용하여, 데이터 수명, 데이터 타입, 데이터 사이즈 및 물리적 또는 논리적 데이터 소스와 같은 데이터 유사도를 가리키는 하나 또는 그보다 많은 데이터 특성들에 기반하여, 데이터 아이템들(28) 각각에 데이터 특성 식별자들(30)을 할당 또는 연관시킬 수 있다. 이러한 방법으로, 동일한 또는 유사한 데이터 특성들을 갖는 데이터 아이템들(28)은 동일한 데이터 특성 식별자 값을 할당받을 수 있다.

[0036] 실시 예로서 작업 흐름 모니터 인터페이스(26)가 운영 체제 레벨에서 제공되므로, 현존하는 응용들에서 본 발명의 실시 예에 따른 데이터 특성 기반 데이터 배치 프로세스에 부합하기 위하여 어떠한 변화도 요구되지 않는다. 따라서, 작업 흐름 모니터 인터페이스(26)는 프로세서 및 운영 체제를 구비한 어떤 타입의 컴퓨팅 장치에도 구현될 수 있으며, 기존 다중 스트림의 사용을 응용들 및 운영 체제들 너머로 확장할 수 있다.

[0037] 도 3은 데이터 특성 기반 데이터 배치(예를 들어, 다중 스트리밍)를 구비한 SSD (12)와 호환되는 다양한 컴퓨팅 장치들(300A~300D)로 구현된 실시 예들의 작업 흐름 모니터 인터페이스를 보여주는 다이어그램이다. 컴퓨팅 장치들(300A, 300B)은, 각각 작업 흐름 모니터 인터페이스들(26A, 26B)이 운영 체제들(20A, 20B)에 제공되거나 또는 옵션으로 스탠드얼론 응용으로 제공되는 퍼스널 컴퓨터 또는 서버 또는 스토리지 하부시스템들과 같은 호스트 장치들을 나타낼 수 있다. 작업 흐름 모니터 인터페이스(26A)는 응용(304A)에 의해 수행되는 데이터 아이템 동작을 모니터링하고, 작업 흐름 저장소로부터의 일치하는 작업 흐름에 기반하여 응용(304A)으로부터의 데이터 아이템들 각각에 데이터 특성 식별자들(30A)을 자동으로 할당할 수 있다. 마찬가지로, 작업 흐름 모니터 인터페이스(26B)는 운영 체제(20B)의 블록 레이어(23B)에 의해 수행되는 데이터 아이템 동작들을 모니터링하고, 작업 흐름 저장소로부터의 일치하는 작업 흐름에 기반하여 블록 레이어(23B)에 의해 수행되는 데이터 아이템들 각각에 데이터 특성 식별자들(30B)을 자동으로 할당할 수 있다.

[0038] 컴퓨팅 장치들(300C, 300D)은 불휘발성 메모리 장치를 활용하는 임의의 하드웨어 장치를 나타내며, 카메라, 마이크, 온도계, GPS, 시계, 센서, 스위치, 라우터, RAID 시스템, 또는 호스트 버스 어댑터(HBA) 시스템을 포함할 수 있다. 일부 타입의 하드웨어 장치들에서, 작업 흐름 모니터 인터페이스(26C)는 하드웨어 장치 제어기(306C) 내에 제공될 수 있다. 작업 흐름 모니터 인터페이스(26C)는 알려진 작업 흐름에 기반하여 장치 제어기(306C)로부터의 데이터 아이템 동작들(예를 들어, 쓰기)에 데이터 특성 식별자들(30C)을 자동으로 할당할 수 있다. 카메라와 같은 다른 타입의 하드웨어 장치들(300D)에서, 작업 흐름 모니터 인터페이스(26D)는 운영 체제의 일부로 구현되거나 또는 프로세서에서 별도의 응용으로 구동될 수 있다. 작업 흐름 모니터 인터페이스(26D)는 프로세서(306D)에 의해 수행되는 데이터 아이템 동작들을 모니터링하고, 알려진 작업 흐름 또는 작업 흐름 저장소로부터의 일치하는 작업 흐름에 기반하여 프로세서(306D)로부터의 데이터 아이템들 각각에 데이터 특성 식별자들(30D)을 자동으로 할당할 수 있다.

[0039] 도 4는 본 발명의 실시 예에 따라 불휘발성 메모리 장치의 스토리지 컨트롤러에 의해 수행되는 데이터 특성 기반 데이터 배치를 위한 프로세스를 보여주는 순서도이다. 도 3 및 도 4를 참조하면, 프로세스는 운영 체제 또는 실행 중인 응용(예를 들어, 운영 체제(20B 또는 응용들(304A)) 중 적어도 하나로부터 저장될 데이터 아이템들의 제1 시리즈를 채널을 통해 수신하는 것을 포함할 수 있다. 데이터 아이템들 각각은, 데이터 유사도를 가리키는 하나 또는 그보다 많은 데이터 특성들(데이터 수명, 데이터 타입, 데이터 사이즈, 논리 블록 어드레스(LBA) 범위, LBA 액세스 패턴, 그리고 물리 데이터 소스를 포함)에 기반하여 데이터 아이템들과 연관된 제1 데이터 특성 식별자를 포함할 수 있다(400). 데이터 수명, 데이터 타입, 데이터 사이즈, 논리 블록 어드레스(LBA) 범위, LBA 액세스 패턴, 그리고 물리 데이터 소스와 같은 다양한 데이터 특성들은, 여기에 나열되지 않은 다른 특성들 또는 현재 알려지지 않은 다른 특성들을 포함하여 액세스 유사도에 대해 고려될 수 있다.

[0040] 예시적으로, 데이터 타입은 원 응용을 넘어선 특성을 포함할 수 있다. 예시적으로, 운영 체제 또는 응용으로부터 수신되는 데이터 아이템들은, 작업 흐름 모니터 인터페이스(26)에 더하여 다른 프로세스에 의해 데이터 아이템들과 연관된 데이터 특성 식별자들을 포함할 수 있다.



- [0041] 스토리지 컨트롤러(22)는 하드웨어 장치(예를 들어, 제어기(306C) 또는 프로세서(306D))로부터 저장될 데이터 아이템들의 다른 시리즈를 채널을 통해 수신할 수 있다. 데이터 아이템들 각각은 데이터 유사도를 가리키는 하나 또는 그보다 많은 데이터 특성들에 기반하여 데이터 아이템들과 연관된 제2 데이터 특성 식별자를 포함할 수 있다(402).
- [0042] 스토리지 컨트롤러(22)는 데이터 특성 식별자들을 읽고, 동일한 데이터 특성 식별자들을 갖는 데이터 아이템들이 블록들의 동일한 세트에 저장되도록 대응하는 데이터 아이템들이 저장될 메모리 장치의 블록들을 식별하고 (404), 그리고 데이터 아이템들을 식별된 블록들에 저장할 수 있다(406).
- [0043] 예를 들어, 도 3에서, SSD (12)의 스토리지 컨트롤러(22)는 데이터 특성 식별자들(30A, 30B)에 각각 할당된 컴퓨팅 장치들(300A, 300B)로부터 데이터 아이템들을 수신할 수 있다. 이후에, 스토리지 컨트롤러(22)는 데이터 특성 식별자(30A)와 연관된 데이터 아이템들을 동일한 소거 블록(308A) 내에 저장하고, 데이터 특성 식별자들(30B)과 연관된 데이터 아이템들을 동일한 소거 블록(308B) 내에 저장할 수 있다. 마찬가지로, SSD (12)의 스토리지 컨트롤러(22)는 데이터 특성 식별자들(30C, 30D)에 각각 할당된 컴퓨팅 장치들(300C, 300D)의 장치 제어기들(306C, 306D)로부터 데이터 아이템들을 수신한다. 이후에, 스토리지 컨트롤러(22)는 데이터 특성 식별자(30C)와 연관된 데이터 아이템들을 동일한 소거 블록(308C)에 저장하고, 데이터 특성 식별자(30D)와 연관된 데이터 아이템들을 동일한 소거 블록(308D)에 저장한다. 예시적으로, 데이터 식별자들이 소스들 사이에 실제적 유사도가 있음을 가리키면, 소거 블록들(308C, 308B)은 상호 배타적일 필요는 없다.
- [0044] 본 발명의 실시 예의 작업 흐름 모니터 인터페이스(26)에 따르면, 데이터 특성 식별자(30)는 저장소에 대한 임의의 타입의 입력 장치로부터 출력되는 데이터 아이템들(28)에 할당될 수 있다. 예를 들어, 작업 흐름 모니터 인터페이스(26)는 이미지를 주기적으로(예를 들어, 매 초마다 한 번씩) 이미지를 획득하는 디지털 보안 카메라 내에 구현되는 것으로 가정한다. 작업 흐름 모니터 인터페이스(26)는 데이터 유사도를 가리키는 데이터 특성들, 예를 들어 이미지의 캡처율, 이미지 파일 사이즈, 그리고 이미지의 근원(예를 들어, 장치 식별자 또는 GPS 위치)와 같은 데이터 특성들에 기반하여 각 이미지 파일에 데이터 특성 식별자(30)를 할당할 수 있다. 이러한 데이터 특성들은 통상적인 다중 스트리밍에 의해 사용되는 메타 데이터와 같이 구체적인 응용 또는 파일 시스템과 연관될 필요는 없다.
- [0045] 도 5는 상이한 수명들 및 상이한 특성들을 갖는 데이터를 저장하는 데이터 베이스 응용에 의해 수행되는 스토리지 동작들에 대한 작업 흐름 모니터 인터페이스(26)의 동작의 예를 보여주는 블록도이다. 예시적으로, 데이터 베이스는 고속 저장을 위한 임베디드형 끊임없는 키-값 저장인 RocksDB일 수 있다. RocksDB는 다수의 CPU 코어들의 장점을 이용하고 그리고 SSD와 같이 입출력 부하에 대해 고속 저장의 효율적인 사용에 최적화된 LevelDB의 갈래일 수 있다. 동작 시에, 키-값 쌍들의 소팅(sorting)된 세트를 갖는 데이터 파일(예를 들어, SST 파일)이 제1 레벨 파일로 순차적으로 기입된다. 쓰기 동작 시에, 파일들을 수정하는 대신에, 일정 시간 동안 변경들이 메모리에서 수집된다. 이후에, 변경들이 제2 레벨 파일로 기입된다. 더 많은 변경들에 대해, 또 다른 레벨이 추가될 수 있다.
- [0046] 작업 흐름 모니터 인터페이스(26)가 RocksDB가 시작함을 검출한 때에, 작업 흐름 모니터 인터페이스(26)는 RocksDB 응용과 연관된 RocksDB 작업 흐름(27')을 획득 및 이용하여 차별적인 데이터 특성 식별자들을 RocksDB에 의해 취급되는 데이터에 할당할 수 있다. RocksDB 작업 흐름(27')은 LSM 트리 작업 흐름과 같은 트리 작업 흐름을 포함할 수 있다. RocksDB는 LSM 트리 작업 흐름의 구현의 일 예이며, 다른 예들은 BigTable, Apache HBase, LevelDB, WiredTiger, Apache Cassandra, Riak, SQLite4 및 Cockroachdb를 포함할 수 있다.
- [0047] 데이터 베이스(500) 및 작업 흐름 모니터 인터페이스(26) 및 RocksDB 작업 흐름(27')이 호스트 시스템의 메모리(18) 내에 도시되어 있다. 데이터 베이스(500)는 메모리(18) 내에 저장된 메모리 테이블(502, Memtable)을 사용할 수 있다. SSD (12) 내에서, 데이터 베이스(500)는 이전 쓰기 로그(504) 및 상이한 레벨들(예를 들어, 레벨 0, 레벨 1, 레벨 m)에 저장되는 다양한 SST 파일들(506)(예를 들어, SST 파일 1, SST 파일 2, SST 파일 n)을 사용할 수 있다.
- [0048] 데이터 베이스(500)가 쓰기 동작(510)을 수행할 때, 데이터 베이스(500)는 데이터를 메모리 테이블(502)에 구조체로 저장할 수 있다. 또한, 데이터 베이스(500)는 쓰기를 이전 쓰기 로그(504)에 추가할 수 있다. 이전 쓰기 로그(504)는 모든 쓰기를 수신하며, 전원이 차단된 후에도 영구적으로 보존될 수 있다. 작업 흐름 모니터 인터페이스(26)는 쓰기 동작(510)을 모니터링하고, RocksDB 작업 흐름(27')에 기반하여 데이터 특성 ID#1을 로그 파일들에 할당할 수 있다.

- [0049] 메모리 테이블(502)은 메모리 테이블(502)의 내용물이 설정(또는 변경) 가능한 문턱에 도달할 때까지 쓰기들을 저장한다. 문턱에 도달한 때에, 플러시 동작(512)이 수행되어 쓰기들이 SSD (12) 내의 SST 파일(506)에 기입된다. 이전 쓰기 로그(504) 내의 데이터는 메모리 테이블(502) 내의 대응하는 데이터가 SST 파일(506)로 플러시된 후에 제거될 수 있다. SST 파일(506)은 통상적으로 메모리 테이블(502)이 플러시된 후에는 다시 기입되지 않는다. 작업 흐름 모니터 인터페이스(26)는 플러시 동작(512)을 모니터링하고, RocksDB 작업 흐름(27')에 기반하여 데이터 특성 ID#2를 SST 파일(506)의 레벨0에 할당할 수 있다.
- [0050] 삽입 또는 갱신이 발생함에 따라, 행들을 겹쳐쓰는 대신에, 데이터 베이스(500)는 삽입된 또는 갱신된 데이터의 새로운 타임스탬프가 찍힌 버전을 또 다른 SST 파일(506)에 기입할 수 있다. 데이터 베이스(500)는 컴팩션을 이용하여 SSD (12)의 SST 파일들(506)의 축적을 관리할 수 있다. 레벨 0에 저장된 데이터가 데이터 문턱에 도달하면, 데이터 베이스(500)는 갱신 동안에 생성되는 삭제된 데이터, 열들 및 행들을 제거하고 그리고 SST 파일들의 레벨 0의 유효한 데이터를 레벨 1에 저장되는 새로운 SST 파일들로 통합함으로써 데이터 베이스 파일을 압축하도록 컴팩션 동작(514)을 수행할 수 있다. 데이터 베이스(500)는 예전의 다중 SST 파일들 대신에 새로운 통합된 SST 파일들을 사용하며, 따라서 읽기 요청들이 컴팩션 이전보다 더 효율적으로 처리된다. 예전의 SST 파일들은 해당 파일들을 사용하는 진행 중인 읽기들이 종료되자마자 삭제된다. 작업 흐름 모니터 인터페이스(26)는 컴팩션 동작(514)을 모니터링하고, RocksDB 작업 흐름(27')에 기반하여 레벨 1 SST 파일들(506)에 데이터 특성 ID#3을 할당할 수 있다.
- [0051] 레벨 1에 저장된 데이터가 데이터 문턱에 도달하면, 데이터 베이스(500)는 사용되지 않는 데이터를 제거하고 레벨 1 SST 파일들의 유효 데이터를 레벨 2에 저장되는 새로운 SST 파일들로 통합하는 컴팩션 동작(516)을 수행할 수 있다. 작업 흐름 모니터 인터페이스(26)는 컴팩션 동작(516)을 모니터링하고, RocksDB 작업 흐름(27')에 기반하여 레벨 2 SST 파일들(506)에 데이터 특성 ID#4를 할당할 수 있다. 본 발명의 실시 예들에 따르면, 하나의 응용은 단일 타입의 파일에 대해 다중 스트림들을 가질 수 있다.
- [0052] 본 발명의 실시 예들에 따르면, 데이터 특성 식별자들은 예측된 수명 그룹들(이 예에서, SST 레벨들)에 기반하여 데이터에 할당되며, 해당 데이터는 예측된 수명들에 기반하여 SSD에 저장될 수 있다. 데이터가 언제 무효화 또는 삭제될 지 알려지지 않지만, 작업 흐름 모니터 인터페이스(26)는 동일한 데이터 특성 식별자들을 갖는 데이터가 동시 또는 거의 동시에 무효화될 것으로 간주한다.
- [0053] 도 6은 작업 흐름 모니터 인터페이스의 데이터 식별자 할당 방법들을 더 상세하게 보여주는 블록도이다. 도 1 및 도 3과 마찬가지로, 작업 흐름 모니터 인터페이스(26)는 호스트 시스템(10)의 운영 체제(20) 레벨에 제공되는 것으로 도시된다. 작업 흐름 모니터 인터페이스(26)는 응용들(300), 파일 시스템(21), 블록 계층 및 장치 드라이버들(23), 그리고 연결된 장치들(302)의 동작들을 모니터링하고, 데이터 특성 식별자들(30)을 대응하는 작업 흐름(27')에 기반하여 데이터 아이템들에 할당할 수 있다. SSD (12)는 데이터 특성 식별자들(30)을 읽고, 동일한 데이터 특성 식별자들(30)을 갖는 데이터 아이템들이 동일한 블록에 저장되도록 대응하는 데이터 아이템들을 저장하는 데에 사용되는 SSD (12)의 블록들을 식별할 수 있다.
- [0054] 본 발명의 실시 예들에 따르면, 작업 흐름 모니터 인터페이스(26)는 배경으로 지속적으로 실행되어 응용들(300) 및 연결된 장치들(302)의 시작을 모니터링할 수 있다. 예시적으로, 작업 흐름 모니터 인터페이스(26)는 SSD (12)에 대한 데이터 아이템 수정들의 데이터 베이스(미도시)를 유지할 수 있다. 작업 흐름 모니터 인터페이스(26)는 응용들 또는 장치에 의해 수행되는 파일 갱신 동작들을 모니터링하고, 작업 흐름(27')에 따라 파일 갱신 동작들 각각에 데이터 특성 식별자(30)를 자동으로 할당할 수 있다.
- [0055] 작업 흐름 모니터 인터페이스(26)는 파일 시스템에서 시스템 호출 후크들(hooks)을 활용하는 구성 요소를 포함하며, 응용들(300), 운영 체제(20) 또는 연결된 장치(302)에 의해 파일들에 행해지는 파일 갱신 동작들(생성, 쓰기, 갱신, 삭제 등)을 모니터링할 수 있다. 윈도우즈 및 리눅스와 같은 대부분의 운영 체제들은 시스템 프로그램 목적을 위한 파일 시스템 후크들을 제공한다. 작업 흐름 모니터 인터페이스(26)는 작업 흐름(27')에서 파일 갱신 동작(또는 일련의 갱신 동작들)을 참조하여 해당 동작에 할당된 미리 정해진 데이터 특성 식별자를 찾을 수 있다.
- [0056] 작업 흐름 모니터 인터페이스(26)는 미리 정해진 데이터 특성 식별자(30)를 갱신 동작에 할당할 수 있다. 예시적으로, 작업 흐름(27')은 파일의 타입 또는 데이터의 다른 특성에 따라, 파일 생성 또는 새로운 파일 쓰기 데이터 특성 식별자들(30)을 할당할 수 있다. 끝으로, 작업 흐름 모니터 인터페이스(26)는 실제 파일 쓰기를 기저의 파일 시스템(21)에 전달할 수 있다.

- [0057] 데이터 특성 식별자 할당의 예로서, 사용자가 사진 응용을 편집하여 "foo.jpg"라 명명된 JPEG 파일 타입의 사진 파일을 편집중이고, 그리고 사용자의 컴퓨터의 SSD (12)는 네 개의 데이터 특성 식별자들 또는 스트림 식별자들을 처리하도록 구성된 경우가 가정된다. 사용자가 사진을 SSD (12)에 저장할 때, 작업 흐름 모니터 인터페이스 (26)는 사진 응용과 연관된 작업 흐름(27')에서 파일 저장 요청을 모니터 및 참조한다. 작업 흐름(27')은 네 가지 데이터 특성 식별자들 중 저장 요청에 할당될 데이터 특성 식별자(예를 들어, 1)를 식별할 수 있다.
- [0058] 작업 흐름 모니터 인터페이스(26)는 작업 흐름(27')에서 특정된 1의 데이터 특성 식별자를 할당한다. 데이터 특성 식별자는 파일 데이터와 함께 파일 시스템(21) 및 블록 계층(23)을 통해 SSD (12)로 전송되며, 1의 데이터 특성 식별자가 할당된 다른 데이터와 함께 블록에 저장된다.
- [0059] 다른 예로서, 작업 흐름 모니터 인터페이스(26) 및 작업 흐름 저장소(27')는 운영 체제의 블록 계층(23) 또는 SSD (12) 내부에 구현될 수 있다.
- [0060] 도 7은 데이터 식별자 할당 방법의 다른 예를 보여주는 블록도이다. 데이터 특성 식별자들을 할당하는 별도의 작업 흐름 모니터 인터페이스(26, 도 6)를 제공하는 대신, 도 7의 실시 예는 알려진 작업 흐름들 및 데이터 배치 설정들을 저장하는 작업 흐름 저장소(27')가 응용들(300), 운영 체제(20)의 다양한 계층들(예를 들어, 파일 시스템(21), 블록 계층(25) 및 장치 드라이버(29) 및 연결된 장치(302)에 실장된 소프트웨어와 같은 소프트웨어 구성 요소를 실행하는 것을 직접적으로 평가 가능하게 할 수 있다. 작업 흐름 저장소(31)는 소프트웨어 구성 요소에 의해 수행되는 활동 또는 기능 동안에 불휘발성 메모리 장치에 기입되는 데이터 아이템들의 상이한 타입들에 미리 정의된 데이터 특성 식별자들을 연관시키는 하나 또는 그보다 많은 미리 정해진 작업 흐름들(33)을 저장할 수 있다.
- [0061] 다양한 소프트웨어 구성 요소들은 작업 흐름 저장소(31)에 직접 요청하여 어떠한 데이터 특성 식별자(30)를 사용할 지 획득할 수 있다. 따라서, 예를 들어, 소프트웨어 구성 요소는 작업 흐름 저장소(31)에 대해 자기 자신을 식별하고, 다양한 기능들을 위해 사용할 데이터 특성 식별자들을 획득하고, 그리고 데이터 특성 식별자들을 데이터 아이템들에 할당할 수 있다.
- [0062] 기존의 구성 요소의 동작들을 모니터하고 데이터 특성 식별자들을 자동으로 인가하는 작업 흐름 모니터의 실시 예와 반대로, 도 7의 실시 예는 작업 흐름 저장소(31)와의 요청 및 응답 메커니즘에 의존하며, 기존의 구성 요소들이 다중 스트리밍 능력을 부가할 수 있다. 작업 흐름 모니터의 실시 예는 기존의 소프트웨어 구성 요소들에서 매우 적은 재프로그래밍을 필요로 하지만, 작업 흐름 모니터 소프트웨어를 생성 및 실행하는 데에 추가적인 작업을 필요로 한다. 도 7의 실시 예는 기존의 소프트웨어 구성 요소에서 일부 재프로그래밍을 필요로 하지만, 작업 흐름 모니터를 생성, 실행 및 유지할 필요가 없다.
- [0063] 본 실시 예의 구체적인 예에서, 소프트웨어 구성 요소는 작업 흐름 저장소(31)에 요청하여 소프트웨어 구성 요소와 연관된 미리 정해진 작업 흐름(33)을 찾는다. 예시적으로, 요청은 소프트웨어 구성 요소의 식별자를 포함할 수 있다. 작업 흐름 저장소(31)로부터의 응답은 소프트웨어 구성 요소에 할당된 작업 흐름(33)의 정체(identity)를 포함할 수 있다.
- [0064] 동작 중에, 소프트웨어 구성 요소는 하나 또는 그보다 많은 요청들을 식별된 작업 흐름(33)으로 전송하여 하나 또는 그보다 많은 데이터 특성 식별자들을 획득하고 소프트웨어 구성 요소에 의해 사용되는 데이터 아이템들에 연관시킬 수 있다. 예시적으로, 요청은 소프트웨어 구성 요소에 의해 수행되는 현재 동작 및 수행되는 현재 동작의 하나 또는 그보다 많은 특성들(예를 들어, 데이터 타입)을 포함할 수 있다. 예시적으로, 소프트웨어 구성 요소는 소프트웨어 구성 요소의 식별자(예를 들어, 이름 또는 식별자 형태)를 이용하여 작업 흐름 저장소(31)에 소프트웨어 구성 요소의 식별을 요청하고, 다음으로 소프트웨어 구성 요소와 연관된 데이터 특성 식별자들을 식별된 작업 흐름(33)에 요청할 수 있다. 다른 예로서, 소프트웨어 구성 요소는 소프트웨어 구성 요소의 정체(또는 식별자) 및 데이터 특성 식별자들의 요청을 포함하는 단일 요청을 할 수 있다.
- [0065] 응용으로서 이름 또는 식별자를 사용하여 작업 흐름 저장소(31)에 요청하기 보다는, 소프트웨어 구성 요소가 운영 체제를 포함하는 예에서 운영 체제의 각 레벨은 하나 또는 그보다 많은 데이터 특성 식별자들을 획득하기 위하여 대응하는 운영 체제 레벨과 연관된 작업 흐름 저장소(31)로의 입력을 제공할 수 있다. 예를 들어, 파일 시스템 레벨에서, 운영 체제는 데이터 특성 식별자들을 요청할 때에 파일 타입, 사이트 등을 이용할 수 있다. 블록 레벨에서, 운영 체제는 LBA 범위 등을 이용할 수 있다. 예시적으로, 운영 체제는 부착된 장치들로부터의 장치 식별자들을 이용하여 작업 흐름 저장소(31)로부터 데이터 특성 식별자들을 획득할 수 있다.
- [0066] 소프트웨어 구성 요소로부터의 요청에 응답하여, 작업 흐름(33)은 해당 동작 및 데이터 아이템의 타입에 대해

작업 흐름(33)에서 특정된 하나 또는 그보다 많은 데이터 특성 식별자들을 소프트웨어 구성 요소에 반환할 수 있다.

- [0067] 작업 흐름(33)으로부터 데이터 특성 식별자들을 수신함에 따라, 소프트웨어 구성 요소는 데이터 특성 식별자를 소프트웨어 구성 요소에 의해 사용되는 데이터 아이템들에 자동으로 할당하고, 데이터 아이템들 및 할당된 데이터 특성 식별자들을 불휘발성 메모리 장치(예를 들어, SSD (12))에 저장할 수 있다.
- [0068] SSD (12)는 데이터 특성 식별자들(30)을 읽고, 동일한 데이터 특성 식별자들(30)을 가진 데이터 아이템들이 동일 블록에 저장되도록, 대응하는 데이터 아이템들을 저장하는 데에 사용될 SSD (12)의 블록을 식별할 수 있다.
- [0069] 데이터 특성 식별자 할당의 예로서, 사용자가 사진 응용을 편집하여 "foo.jpg"라 명명된 JPEG 파일 타입의 사진 파일을 편집중이고, 그리고 사용자의 컴퓨터의 SSD (12)는 네 개의 데이터 특성 식별자들 또는 스트림 식별자들을 처리하도록 구성된 경우가 가정된다. 사용자가 사진을 SSD (12)에 저장할 때, 사진 응용은 파일 저장 동작에 할당될 데이터 특성 식별자를 위해 사진 응용과 연관된 작업 흐름(33)에 요청을 전달할 수 있다. 작업 흐름(33)은 네 가지 데이터 특성 식별자들 저장 요청에 할당될 데이터 특성 식별자(예를 들어, 1)를 특정할 수 있다.
- [0070] 사진 응용은 작업 흐름(33)에서 특정된 1의 데이터 특성 식별자를 할당한다. 데이터 특성 식별자는 파일 데이터와 함께 파일 시스템(21) 및 블록 계층(23)을 통해 SSD (12)로 전송되며, 1의 데이터 특성 식별자가 할당된 다른 데이터를 저장하는 블록에 저장된다.
- [0071] 다른 예로서, 작업 흐름 저장소(31)는 데이터 베이스 응용으로 구현될 수 있다.
- [0072] 예시적으로, 작업 흐름 모니터 인터페이스(26)는 소프트웨어 구성 요소로 구현될 수 있다. 다른 예로서, 작업 흐름 모니터 인터페이스(26)는 하드웨어 및 소프트웨어의 조합으로 구현될 수 있다. 작업 흐름 모니터 인터페이스(26)가 단일 구성 요소로 도시되어 있지만, 작업 흐름 모니터 인터페이스(26)의 기능은 더 많은 수의 모듈들 또는 구성 요소들을 이용하여 구현될 수 있다.
- [0073] 본 발명의 실시 예들에 따른 작업 흐름 모니터 인터페이스(26)는 클라이언트로부터 기업까지, 단일 스탠드얼론 기계(데스크톱, 랩톱, 워크스테이션, 서버 등)의 디스크, 스토리지 어레이, 소프트웨어 정의 스토리지(SDS), 응용 특정 스토리지, 가상 기계(VM), 가상 데스크톱 인프라스트럭처(VDI), 콘텐츠 분산 네트워크(CDN) 등과 같은 폭넓은 저장 시장에 적용될 수 있다.
- [0074] 예시적으로, SSD (12)의 NVM 어레이(24)는 복수의 플래시 메모리들과 같은 복수의 불휘발성 메모리 칩들로 형성될 수 있다. 다른 예로서, NVM 어레이(24)는 플래시 메모리 칩들 대신에 상이한 타입의 불휘발성 메모리 칩들(예를 들어, PRAM, FRAM, MRAM 등)로 형성될 수 있다. 다른 예로서, NVM 어레이(24)는 DRAM 또는 SRAM과 같은 휘발성 메모리들로 형성될 수 있으며, 둘 또는 그보다 많은 타입의 메모리들이 혼합된 하이브리드 타입을 가질 수 있다.
- [0075] 스토리지 장치의 데이터 특성 기반 데이터 배치를 위한 방법들 및 시스템들, 그리고 컴퓨터 시스템이 데이터 배치 방법을 활용할 수 있게 하는 작업 흐름 모니터 인터페이스가 설명되었다. 본 발명은 도시된 실시 예들에 따라 설명되었으며, 실시 예들에 다양한 변경들이 가능하고, 이러한 변경은 본 발명의 기술적 사상의 범위에 속한다. 예를 들어, 예시적인 실시 예들은 하드웨어, 소프트웨어, 프로그램 명령들을 저장하는 컴퓨터로 독출 가능한 매체 또는 이들의 조합으로 구현될 수 있다. 본 발명에 따라 기입된 소프트웨어는 메모리, 하드 디스크, 또는 CD/DVD-ROM과 같은 컴퓨터로 독출 가능한 매체의 형태로 저장될 수 있으며, 프로세서에 의해 실행될 수 있다. 따라서, 다양한 변경들이 이 분야에 통상적인 기술을 가진 자에 의해 첨부된 청구 범위의 기술적 사상으로부터 괴리되지 않은 범위에서 수행될 수 있다.

## 부호의 설명

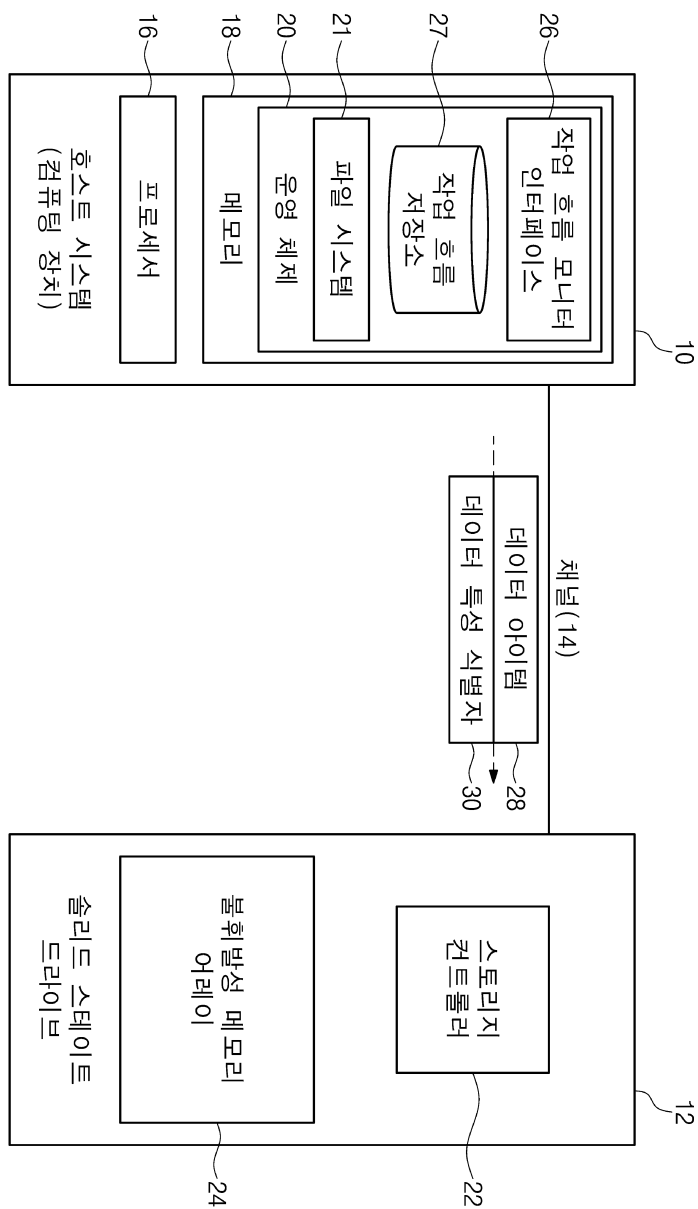
- [0076] 10; 호스트 시스템
- 12; 솔리드 스테이트 드라이브
- 14; 채널
- 16; 프로세서
- 18; 메모리



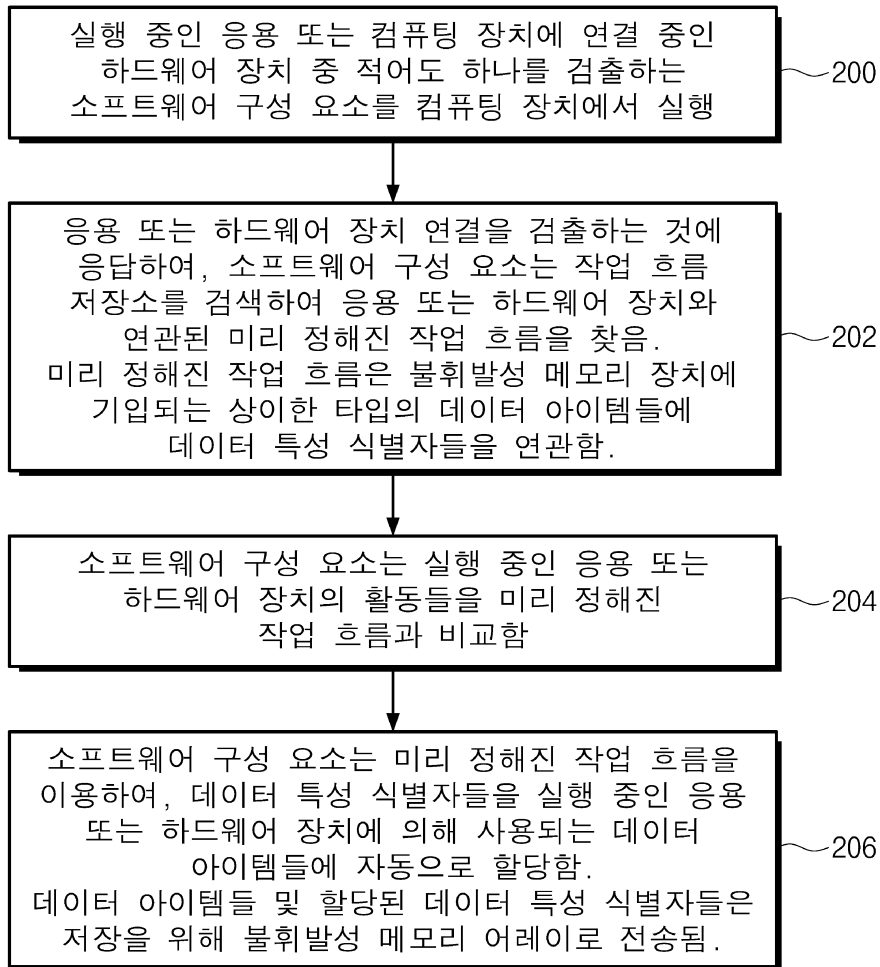
- 20; 운영체제
- 21; 파일 시스템
- 22; 스토리지 컨트롤러
- 24; 불휘발성 메모리 어레이
- 26; 작업 흐름 모니터 인터페이스
- 27; 작업 흐름 저장소
- 28; 데이터 아이템
- 29' 데이터 특성 식별자

도면

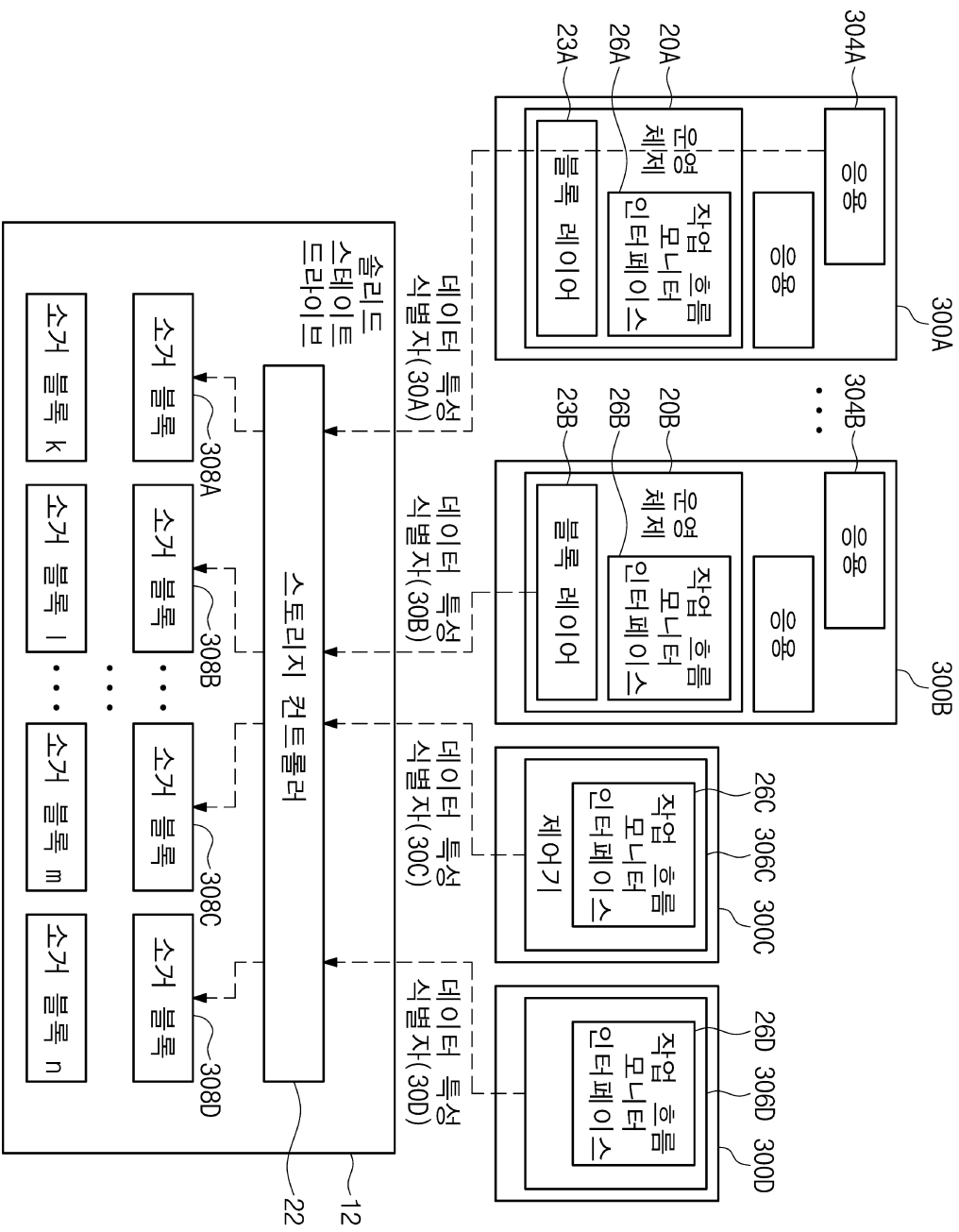
도면1



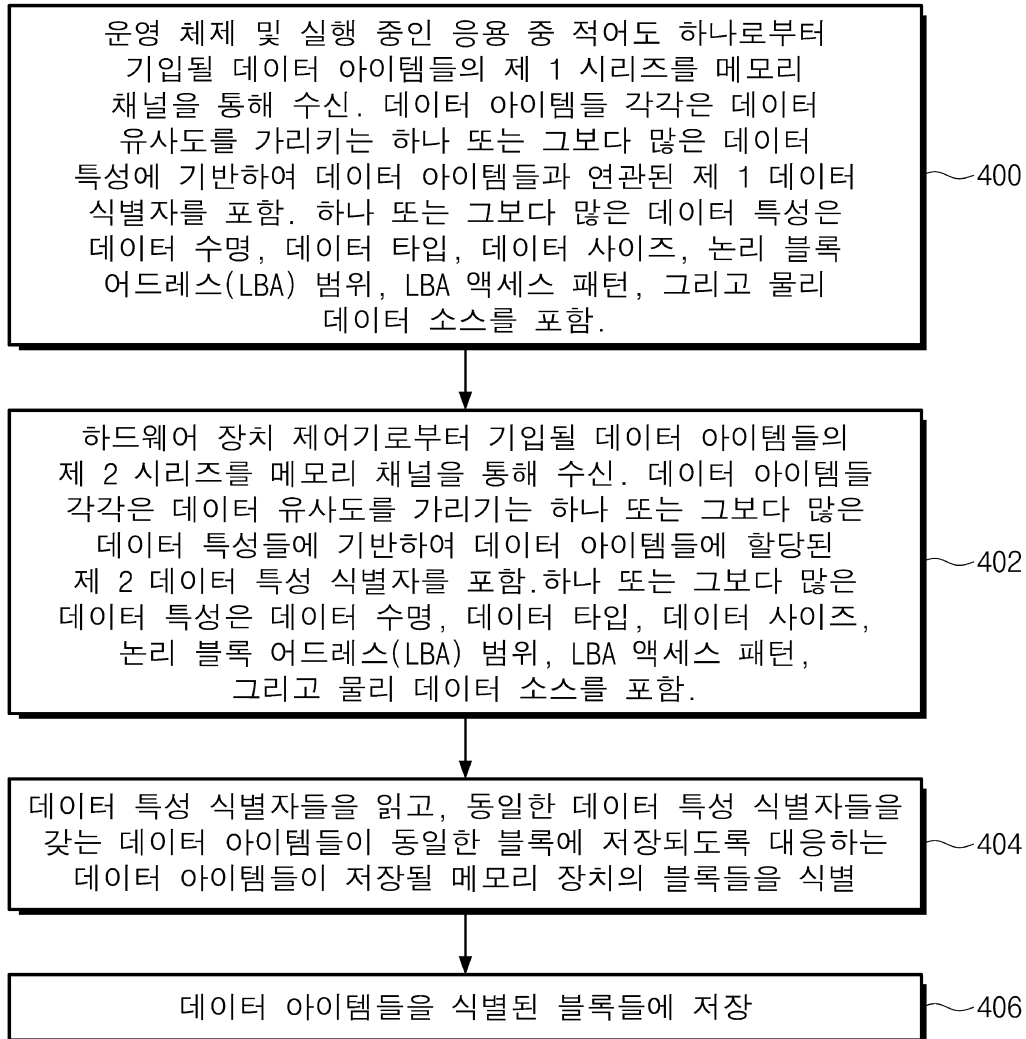
도면2



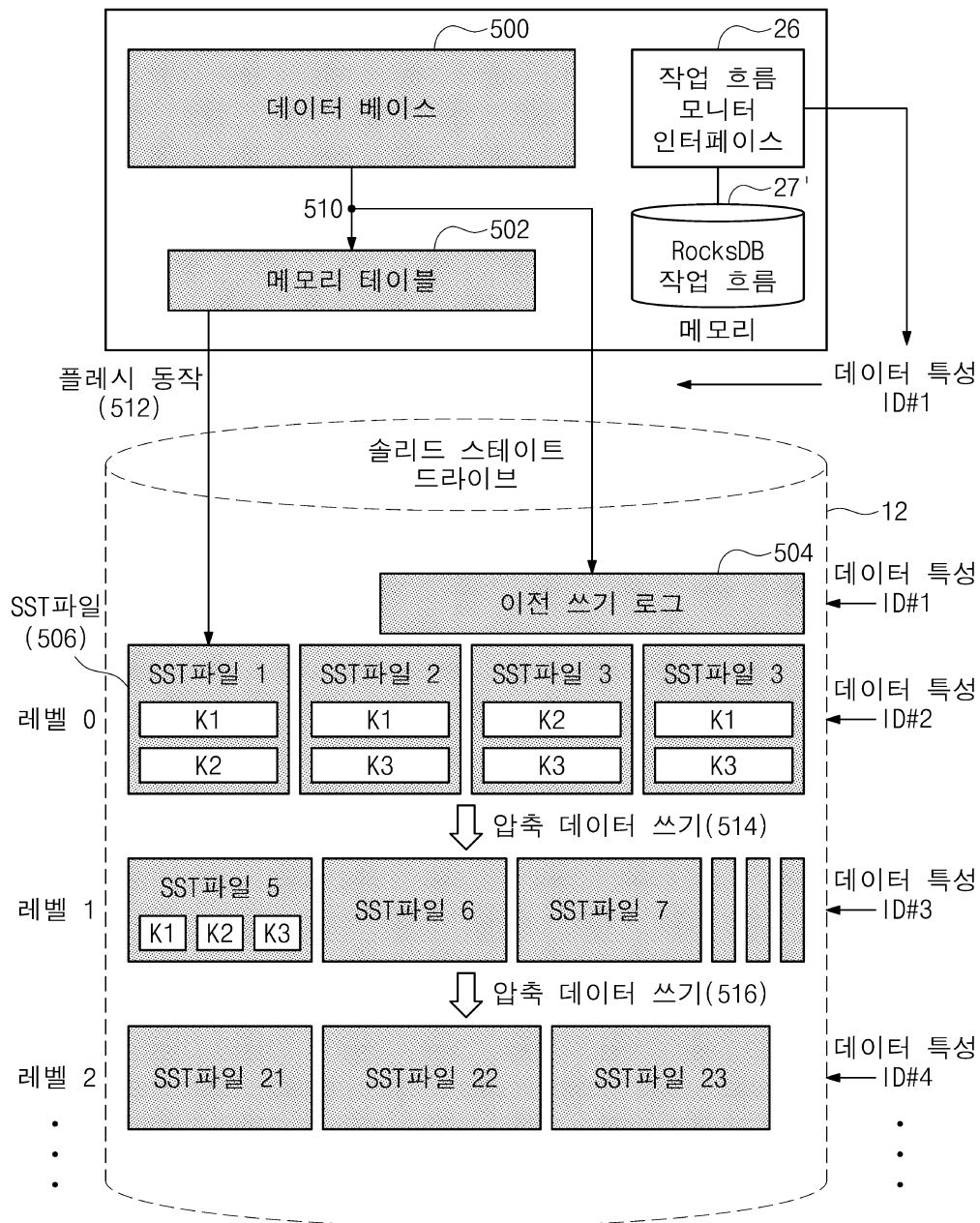
도면3



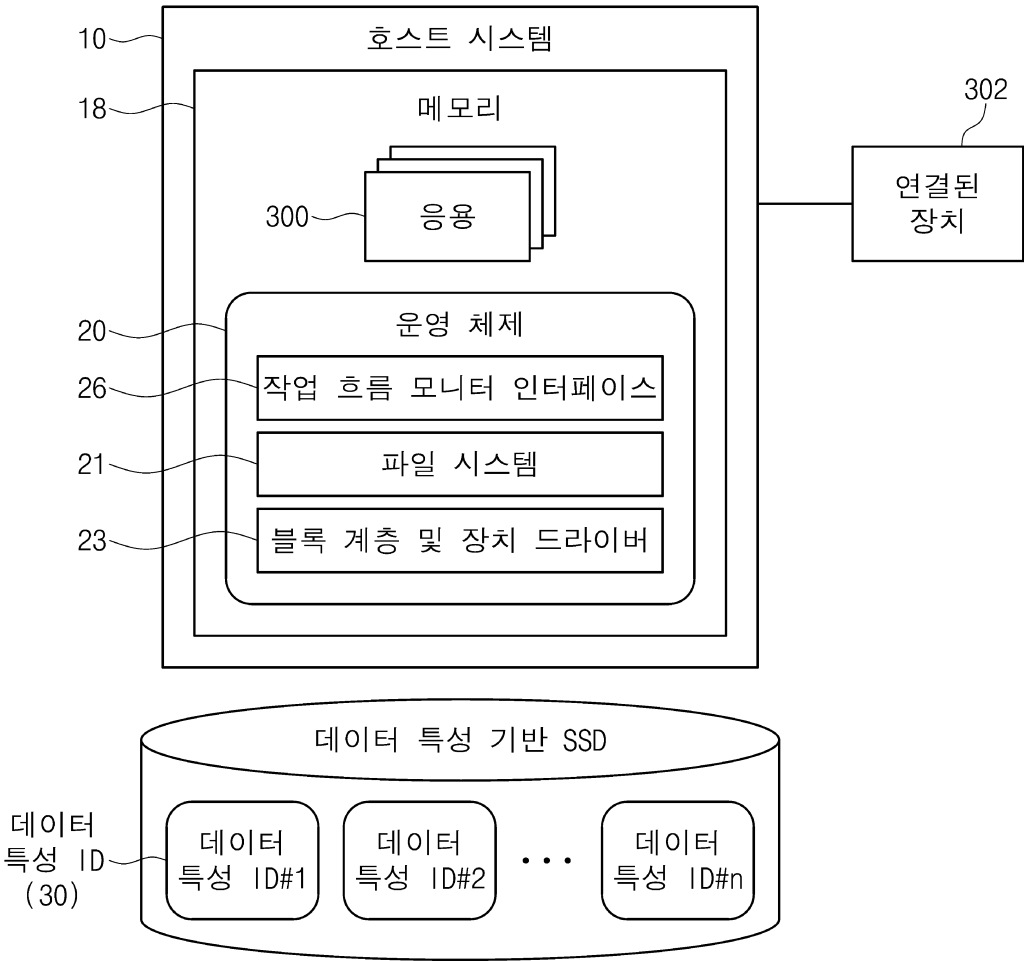
도면4



도면5



도면6



도면7

