



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2020-0098971
(43) 공개일자 2020년08월21일

(51) 국제특허분류(Int. Cl.)
G06F 12/06 (2006.01) G06F 12/02 (2018.01)
G06F 3/06 (2006.01)
(52) CPC특허분류
G06F 12/0638 (2013.01)
G06F 12/0246 (2013.01)
(21) 출원번호 10-2019-0016770
(22) 출원일자 2019년02월13일
심사청구일자 2019년02월13일

(71) 출원인
울산과학기술원
울산광역시 울주군 언양읍 유니스트길 50
(72) 발명자
최영리
울산광역시 울주군 언양읍 유니스트길 50 울산과
학기술대학교
KAIYRAKHMETOLZHAS
울산광역시 울주군 언양읍 유니스트길 50 울산과
학기술대학교
(74) 대리인
제일특허법인(유)

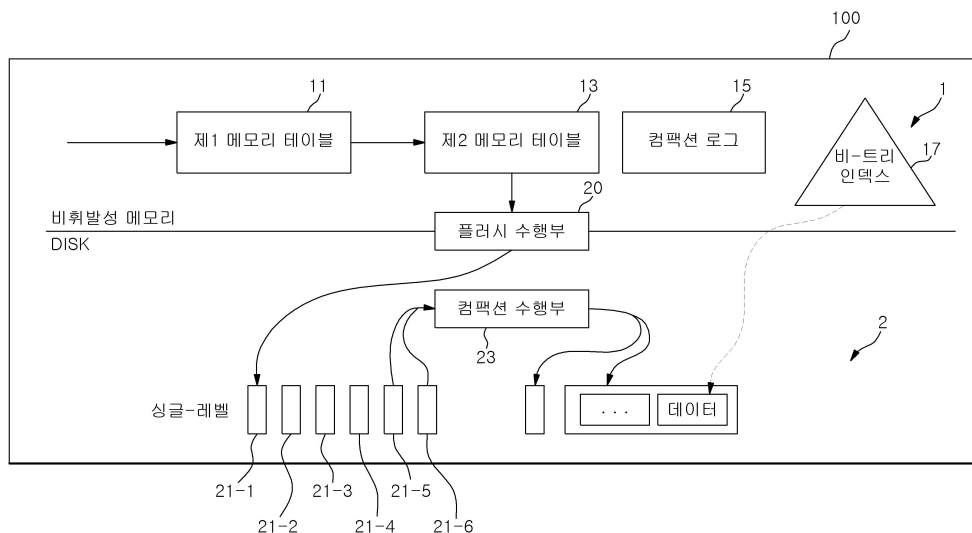
전체 청구항 수 : 총 14 항

(54) 발명의 명칭 싱글-레벨 기반의 데이터 저장 장치 및 방법

(57) 요약

본 발명의 일 실시예에 따른 싱글-레벨 기반의 데이터 저장 방법은, 비휘발성 메모리에 포함된 제1 메모리 테이블에 데이터를 저장하는 단계와, 상기 제1 메모리 테이블에 저장된 데이터의 용량이 소정 값 이상인 경우, 상기 제1 메모리 테이블에 저장된 적어도 일부의 데이터를 상기 비휘발성 메모리에 포함된 제2 메모리 테이블에 저장하는 단계와, 상기 제2 메모리 테이블에 저장된 데이터를 식별하여 플러시(flush) 동작을 수행하는 단계와, 상기 플러시 동작에 기초하여 상기 제2 메모리 테이블에 저장된 데이터를 상기 비휘발성 메모리와는 구분되는 디스크(disk)에 싱글-레벨로 저장하는 단계를 포함한다.

대표도



(52) CPC특허분류

G06F 3/0646 (2013.01)

G06F 3/068 (2013.01)

G06F 2212/7202 (2013.01)

G06F 2212/7204 (2013.01)

이 발명을 지원한 국가연구개발사업

과제고유번호 1711075423

부처명 과학기술정보통신부

연구관리전문기관 한국연구재단

연구사업명 개인기초연구(과기정통부)(R&D)

연구과제명 비휘발성 메모리 최적 활용을 위한 클라우드 컴퓨팅 기술 연구

기 여 율 1/2

주관기관 울산과학기술원

연구기간 2018.03.01 ~ 2019.02.28

이 발명을 지원한 국가연구개발사업

과제고유번호 1711073161

부처명 과학기술정보통신부

연구관리전문기관 정보통신기술진흥센터

연구사업명 SW컴퓨팅산업원천기술개발(R&D)

연구과제명 빅데이터 처리 고도화 핵심 기술개발 사업 총괄 및 고성능 컴퓨팅 기술을 활용한 성능 가속화 기술 개발

기 여 율 1/2

주관기관 울산과학기술원

연구기간 2015.06.01 ~ 2019.05.31

명세서

청구범위

청구항 1

비휘발성 메모리에 포함된 제1 메모리 테이블에 데이터를 저장하는 단계와,

상기 제1 메모리 테이블에 저장된 데이터의 용량이 소정 값 이상인 경우, 상기 제1 메모리 테이블에 저장된 적어도 일부의 데이터를 상기 비휘발성 메모리에 포함된 제2 메모리 테이블에 저장하는 단계와,

상기 제2 메모리 테이블에 저장된 데이터를 식별하여 플러시(flush) 동작을 수행하는 단계와,

상기 플러시 동작에 기초하여 상기 제2 메모리 테이블에 저장된 데이터를 상기 비휘발성 메모리와는 구분되는 디스크(disk)에 싱글-레벨로 저장하는 단계를 포함하는

싱글-레벨 기반의 데이터 저장 방법.

청구항 2

제1항에 있어서,

상기 싱글-레벨로 저장하는 단계는,

비-트리(B-tree) 데이터 구조에 기초하여 데이터가 검색되도록, 상기 플러시 동작에 기초하여 상기 제2 메모리 테이블에 저장된 데이터를 싱글-레벨로 저장하는 단계를 포함하는

싱글-레벨 기반의 데이터 저장 방법.

청구항 3

제2항에 있어서,

상기 싱글-레벨로 저장된 데이터의 적어도 일부에 대해 컴팩션(compaction)을 수행하는 단계를 포함하는

싱글-레벨 기반의 데이터 저장 방법.

청구항 4

제3항에 있어서,

상기 컴팩션을 수행하는 단계는,

상기 싱글-레벨로 저장된 데이터 중 업데이트 전의 데이터를 삭제하고, 상기 싱글-레벨로 저장된 나머지 데이터를 상기 나머지 데이터에 포함되는 키 값(key value)에 기초하여 정렬하는 단계를 포함하는

싱글-레벨 기반의 데이터 저장 방법.

청구항 5

제3항에 있어서,

상기 컴팩션을 수행하는 단계는,

상기 싱글-레벨로 저장된 데이터 중 적어도 일부에서 저장된 시점이 업데이트 시점 이전인 데이터와 업데이트 시점 이후의 데이터의 비율을 식별하는 단계와,

상기 식별된 비율이 소정 값 보다 낮은 경우, 상기 업데이트 시점 이전인 데이터의 적어도 일부를 컴팩션 후보(compaction candidate)로 선택하는 단계를 포함하고,

상기 선택된 컴팩션 후보 중 적어도 일부를 삭제하는 단계를 포함하는

싱글-레벨 기반의 데이터 저장 방법.

청구항 6

제3항에 있어서,

상기 컴팩션을 수행하는 단계는,

상기 비-트리의 각 노드 중 소정 개수의 노드를 스캔하는 단계와,

상기 싱글-레벨로 저장된 데이터 중 상기 스캔된 노드와 관련된 데이터가 소정 개수 이상의 데이터 테이블 파일에 분산되어 저장된 경우, 상기 데이터 테이블 파일을 상기 컴팩션 후보로 선택하는 단계를 포함하는

싱글-레벨 기반의 데이터 저장 방법.

청구항 7

제5항에 있어서,

상기 컴팩션을 수행하는 단계는,

상기 싱글-레벨로 저장된 데이터를 읽기(read) 위해 요구되는 데이터 테이블 파일의 수를 산출하는 단계와,

상기 산출된 데이터 테이블 파일의 수가 가장 큰 데이터와 관련된 데이터 테이블 파일을 상기 컴팩션 후보로 선택하는 단계를 포함하는

싱글-레벨 기반의 데이터 저장 방법.

청구항 8

입력되는 데이터를 저장하는 제1 메모리 테이블과,

상기 제1 메모리 테이블에 저장된 데이터의 용량이 소정 값 이상인 경우, 상기 제1 메모리 테이블에 저장된 적어도 일부의 데이터를 저장하는 제2 메모리 테이블과,

상기 제2 메모리 테이블에 저장된 데이터를 식별하여 플러시(flush) 동작을 수행하는 플러시 수행부와,

상기 플러시 동작에 기초하여 상기 제2 메모리 테이블에 저장된 데이터를 상기 비휘발성 메모리와는 구분되는 디스크(disk)에 싱글-레벨로 저장하는 싱글-레벨 저장부를 포함하고,

상기 제1 메모리 테이블 및 상기 제2 메모리 테이블은 비휘발성 메모리에 저장되는

싱글-레벨 기반의 데이터 저장 장치.

청구항 9

제8항에 있어서,

상기 싱글-레벨 저장부는,

비-트리(B-tree) 데이터 구조에 기초하여 데이터가 검색되도록, 상기 플러시 동작에 기초하여 상기 제2 메모리 테이블에 저장된 데이터를 싱글-레벨로 저장하는

싱글-레벨 기반의 데이터 저장 장치.

청구항 10

제9항에 있어서,

상기 싱글-레벨로 저장된 데이터의 적어도 일부에 대해 컴팩션(compaction)을 수행하는 컴팩션 수행부를 포함하는

싱글-레벨 기반의 데이터 저장 장치.

청구항 11

제10항에 있어서,

상기 컴팩션 수행부는,

상기 싱글-레벨로 저장된 데이터 중 업데이트 전의 데이터를 삭제하고, 상기 싱글-레벨로 저장된 나머지 데이터를 상기 나머지 데이터에 포함되는 키 값(key value)에 기초하여 정렬하는

싱글-레벨 기반의 데이터 저장 장치.

청구항 12

제10항에 있어서,

상기 컴팩션 수행부는,

상기 싱글-레벨로 저장된 데이터 중 적어도 일부에서 저장된 시점이 업데이트 시점 이전인 데이터와 업데이트 시점 이후의 데이터의 비율을 식별하는 단계와,

상기 식별된 비율이 소정 값 보다 낮은 경우, 상기 소정 시점 이전인 데이터의 적어도 일부를 컴팩션 후보(compaction candidate)로 선택하는 단계를 포함하고,

상기 선택된 컴팩션 후보 중 적어도 일부를 삭제하는

싱글-레벨 기반의 데이터 저장 장치.

청구항 13

제10항에 있어서,

상기 컴팩션 수행부는,

상기 비-트리의 각 노드 중 소정 개수의 노드를 스캔하고,

상기 싱글-레벨로 저장된 데이터 중 상기 스캔된 노드와 관련된 데이터가 소정 개수 이상의 데이터 테이블 파일에 분산되어 저장된 경우, 상기 데이터 테이블 파일을 상기 컴팩션 후보로 선택하는

싱글-레벨 기반의 데이터 저장 장치.

청구항 14

제10항에 있어서,

상기 컴팩션 수행부는,

상기 싱글-레벨로 저장된 데이터를 읽기(read) 위해 요구되는 데이터 테이블 파일의 수를 산출하고,

상기 산출된 데이터 테이블 파일의 수가 가장 큰 데이터와 관련된 데이터 테이블 파일을 상기 컴팩션 후보로 선택하는

싱글-레벨 기반의 데이터 저장 장치.

발명의 설명

기술 분야

[0001] 본 발명은 비휘발성 메모리를 이용하며 싱글-레벨로 데이터를 저장하는 데이터 저장 장치 및 방법에 관한 것이다.

배경 기술

[0002] 최근에는 빅 데이터 수요 및 데이터 분석 요구에 따라, 데이터베이스에서 원하는 정보를 찾기 위해 데이터가 저장된 파일의 위치를 나타내는 키(key) 값에 기초하여 정렬되는 인덱스 기능을 가지는 시스템이 요구되고 있다. 인덱스 기능은 포함되는 데이터가 쉽게 검색될 수 있도록 비-트리, LSM 트리 등과 같은 효율적인 구조를 가진다.

[0003] 비-트리(B-tree)를 이용한 인덱스 기능은 데이터를 키(key)에 따라 정렬하여 유지하고, 추가적인 데이터의

삽입, 삭제 시 정렬된 데이터를 갱신하여 다시 기록한다. 이 방식은 추가적인 데이터 삽입 또는 삭제 시, 대상 데이터 영역의 내용을 수정 후 다시 쓰기를(re-write)한다. 이러한 데이터의 다시 쓰기는 무작위 쓰기(random write) 형태로 발생하게 되는데, 이러한 경우, SSD 최대 성능 발휘를 어렵게 만들며, 가비지 컬렉션(garbage collection) 부하를 증가시킨다.

- [0004] LSM 트리(log structured merge tree)를 이용한 인덱스 기능은 데이터를 키에 따라 정렬하여 유지하되, 추가적인 데이터의 삽입, 삭제시, 새롭게 정렬된 데이터로 독립적으로 기록한다. 이 방식은 추가적인 데이터 삽입 또는 삭제시, 추가적인 연산에 대한 데이터를 따로 정렬하여 저장한다. 이 때, 순차 쓰기(sequential write)로 I/O(input/output)가 발생하므로 최대 쓰기 성능이 발휘된다. 다만, 독립적으로 정렬된 데이터가 늘어남에 따라, 데이터 조회 시 각각 분리된 데이터를 모두 접근해야 하므로 성능이 떨어지는 문제가 있다.

선행기술문헌

특허문헌

- [0005] (특허문헌 0001) 한국공개특허 제10-2014-0070834호 (2014년 06월 11일 공개)

발명의 내용

해결하려는 과제

- [0006] 본 발명이 해결하고자 하는 과제는, 데이터의 읽기 및 쓰기에 대해 보다 높은 성능 및 효율을 제공하는 데이터 저장 방법 및 장치를 제공하는 것이다.

- [0007] 다만, 본 발명이 해결하고자 하는 과제는 이상에서 언급한 바로 제한되지 않으며, 언급되지는 않았으나 아래의 기재로부터 본 발명이 속하는 기술분야에서 통상의 지식을 가진 자에게 명확하게 이해될 수 있는 목적을 포함할 수 있다.

과제의 해결 수단

- [0008] 본 발명의 일 실시예에 따른 싱글-레벨 기반의 데이터 저장 방법은, 비휘발성 메모리에 포함된 제1 메모리 테이블에 데이터를 저장하는 단계와 상기 제1 메모리 테이블에 저장된 데이터의 용량이 소정 값 이상인 경우, 상기 제1 메모리 테이블에 저장된 적어도 일부의 데이터를 상기 비휘발성 메모리에 포함된 제2 메모리 테이블에 저장하는 단계와, 상기 제2 메모리 테이블에 저장된 데이터를 식별하여 플러시(flush) 동작을 수행하는 단계와, 상기 플러시 동작에 기초하여 상기 제2 메모리 테이블에 저장된 데이터를 상기 비휘발성 메모리와는 구분되는 디스크(disk)에 싱글-레벨로 저장하는 단계를 포함한다.

- [0009] 또한, 상기 싱글-레벨로 저장하는 단계는, 비-트리(B-tree) 데이터 구조에 기초하여 데이터가 검색되도록, 상기 플러시 동작에 기초하여 상기 제2 메모리 테이블에 저장된 데이터를 싱글-레벨로 저장하는 단계를 포함할 수 있다.

- [0010] 또한, 상기 싱글-레벨로 저장된 데이터의 적어도 일부에 대해 컴팩션(compaction)을 수행하는 단계를 포함할 수 있다.

- [0011] 또한, 상기 컴팩션을 수행하는 단계는, 상기 싱글-레벨로 저장된 데이터 중 업데이트 전의 데이터를 삭제하고, 상기 싱글-레벨로 저장된 나머지 데이터를 상기 나머지 데이터에 포함되는 키 값(key value)에 기초하여 정렬하는 단계를 포함할 수 있다.

- [0012] 또한, 상기 컴팩션을 수행하는 단계는, 상기 싱글-레벨로 저장된 데이터 중 적어도 일부에서 저장된 시점이 업데이트 시점 이전인 데이터와 업데이트 시점 이후의 데이터의 비율을 식별하는 단계와, 상기 식별된 비율이 소정 값 보다 낮은 경우, 상기 업데이트 시점 이전인 데이터의 적어도 일부를 컴팩션 후보(compaction candidate)로 선택하는 단계를 포함하고, 상기 선택된 컴팩션 후보 중 적어도 일부를 삭제하는 단계를 포함할 수 있다.

- [0013] 또한, 상기 컴팩션을 수행하는 단계는, 상기 비-트리의 각 노드 중 소정 개수의 노드를 스캔하는 단계와, 상기 싱글-레벨로 저장된 데이터 중 상기 스캔된 노드와 관련된 데이터가 소정 개수 이상의 데이터 테이블 파일에 분산되어 저장된 경우, 상기 데이터 테이블 파일을 상기 컴팩션 후보로 선택하는 단계를 포함할 수 있다.

- [0014] 또한, 상기 컴팩션을 수행하는 단계는, 상기 싱글-레벨로 저장된 데이터를 읽기(read) 위해 요구되는 데이터 테이블 파일의 수를 산출하는 단계와, 상기 산출된 데이터 테이블 파일의 수가 가장 큰 데이터와 관련된 데이터 테이블 파일을 상기 컴팩션 후보로 선택하는 단계를 포함할 수 있다.
- [0015] 본 발명의 일 실시예에 따른 싱글-레벨 기반의 데이터 저장 장치는, 입력되는 데이터를 저장하는 제1 메모리 테이블과, 상기 제1 메모리 테이블에 저장된 데이터의 용량이 소정 값 이상인 경우, 상기 제1 메모리 테이블에 저장된 적어도 일부의 데이터를 저장하는 제2 메모리 테이블과, 상기 제2 메모리 테이블에 저장된 데이터의 상태를 식별하여 플러시(flush) 동작을 수행하는 플러시 수행부와, 상기 플러시 동작에 기초하여 상기 제2 메모리 테이블에 저장된 데이터 중 적어도 일부를 상기 비휘발성 메모리와는 구분되는 디스크(disk)에 싱글-레벨로 저장하는 싱글-레벨 저장부를 포함하고, 상기 제1 메모리 테이블 및 상기 제2 메모리 테이블은 비휘발성 메모리에 저장된다.
- [0016] 또한, 상기 싱글-레벨 저장부는, 비-트리(B-tree) 데이터 구조에 기초하여 데이터가 검색되도록, 상기 플러시 동작에 기초하여 상기 제2 메모리 테이블에 저장된 데이터를 싱글-레벨로 저장할 수 있다.
- [0017] 또한, 상기 싱글-레벨로 저장된 데이터의 적어도 일부에 대해 컴팩션(compaction)을 수행하는 컴팩션 수행부를 포함할 수 있다.
- [0018] 또한, 상기 컴팩션 수행부는, 상기 싱글-레벨로 저장된 데이터 중 상기 싱글-레벨로 저장된 데이터 중 업데이트 전의 데이터를 삭제하고, 상기 싱글-레벨로 저장된 나머지 데이터를 상기 나머지 데이터에 포함되는 키 값(key value)에 기초하여 정렬할 수 있다.
- [0019] 또한, 상기 컴팩션 수행부는, 상기 싱글-레벨로 저장된 데이터 중 적어도 일부에서 저장된 시점이 업데이트 시점 이전인 데이터와 업데이트 시점 이후의 데이터의 비율을 식별하는 단계와, 상기 식별된 비율이 소정 값 보다 낮은 경우, 상기 소정 시점 이전인 데이터의 적어도 일부를 컴팩션 후보(compaction candidate)로 선택하는 단계를 포함하고, 상기 선택된 컴팩션 후보 중 적어도 일부를 삭제할 수 있다.
- [0020] 또한, 상기 컴팩션 수행부는, 상기 비-트리의 각 노드 중 소정 개수의 노드를 스캔하고, 상기 싱글-레벨로 저장된 데이터 중 상기 스캔된 노드와 관련된 데이터가 소정 개수 이상의 데이터 테이블 파일에 분산되어 저장된 경우, 상기 데이터 테이블 파일을 상기 컴팩션 후보로 선택할 수 있다.
- [0021] 또한, 상기 컴팩션 수행부는, 상기 싱글-레벨로 저장된 데이터를 읽기(read) 위해 요구되는 데이터 테이블 파일의 수를 산출하고, 상기 산출된 데이터 테이블 파일의 수가 가장 큰 데이터와 관련된 데이터 테이블 파일을 상기 컴팩션 후보로 선택할 수 있다.

발명의 효과

- [0022] 본 발명의 실시예에 따른 싱글-레벨 기반의 데이터 저장 장치 및 방법은 데이터의 읽기 및 쓰기에 대해 보다 높은 성능 및 효율을 제공할 수 있다.
- [0023] 다만, 본 발명에서 얻을 수 있는 효과는 이상에서 언급한 효과들로 제한되지 않으며, 언급하지 않은 또 다른 효과들은 아래의 기재로부터 본 개시가 속하는 기술 분야에서 통상의 지식을 가진 자에게 명확하게 이해될 수 있을 것이다.

도면의 간단한 설명

- [0024] 도 1은 본 발명의 일 실시예에 따른 싱글-레벨 데이터 저장 방법의 개념도를 도시한다.
- 도 2는 본 발명의 일 실시예에 따른 싱글-레벨 데이터 저장 장치의 기능적 구성의 예를 도시한다.
- 도 3은 본 발명의 일 실시예에 따른 싱글-레벨 데이터 저장 방법의 각 단계의 흐름을 도시한다.
- 도 4는 본 발명의 일 실시예에 따른 싱글-레벨 데이터 저장 방법의 읽기(read) 단계의 흐름을 개념적으로 도시한다.
- 도 5는 본 발명의 일 실시예에 따른 싱글-레벨 데이터 저장 방법의 쓰기(write) 단계의 흐름을 개념적으로 도시한다.

발명을 실시하기 위한 구체적인 내용

- [0025] 본 발명의 이점 및 특징, 그리고 그것들을 달성하는 방법은 첨부되는 도면과 함께 상세하게 후술되어 있는 실시예들을 참조하면 명확해질 것이다. 그러나 본 발명은 이하에서 개시되는 실시예들에 한정되는 것이 아니라 다양한 형태로 구현될 수 있으며, 단지 본 실시예들은 본 발명의 개시가 완전하도록 하고, 본 발명이 속하는 기술분야에서 통상의 지식을 가진 자에게 발명의 범주를 완전하게 알려주기 위해 제공되는 것이며, 본 발명의 범주는 청구항에 의해 정의될 뿐이다.
- [0026] 본 발명의 실시예들을 설명함에 있어서 공지 기능 또는 구성에 대한 구체적인 설명은 본 발명의 실시예들을 설명함에 있어 실제로 필요한 경우 외에는 생략될 것이다. 그리고 후술되는 용어들은 본 발명의 실시예에서의 기능을 고려하여 정의된 용어들로서 이는 사용자, 운용자의 의도 또는 관례 등에 따라 달라질 수 있다. 그러므로 그 정의는 본 명세서 전반에 걸친 내용을 토대로 내려져야 할 것이다.
- [0027] 본 발명은 다양한 변경을 가할 수 있고 여러 가지 실시예들을 포함할 수 있는바, 특정 실시예들을 도면에 예시하고 상세한 설명에 설명하고자 한다. 그러나 이는 본 발명을 특정한 실시 형태에 대해 한정하려는 것이 아니며, 본 발명의 사상 및 기술 범위에 포함되는 모든 변경, 균등물 내지 대체물을 포함하는 것으로서 이해되어야 한다.
- [0028] 제 1, 제 2 등과 같이 서수를 포함하는 용어는 다양한 구성요소들을 설명하는데 사용될 수 있지만, 해당 구성요소들은 이와 같은 용어들에 의해 한정되지는 않는다. 이 용어들은 하나의 구성요소들을 다른 구성요소로부터 구별하는 목적으로만 사용된다.
- [0029] 어떤 구성요소가 다른 구성요소에 '연결되어' 있거나 '접속되어' 있다고 언급된 때에는, 그 다른 구성요소에 직접적으로 연결되어 있거나 또는 접속되어 있을 수도 있지만, 중간에 다른 구성요소가 존재할 수도 있다고 이해되어야 할 것이다.
- [0030] 도 1은 본 발명의 일 실시예에 따른 싱글-레벨 데이터 저장 방법의 개념도를 도시한다. 이러한 싱글-레벨 데이터 저장 방법은 싱글-레벨 데이터 저장 장치(100)에 의해 수행되는 것일 수 있다.
- [0031] 도 1을 참조하면, 싱글-레벨 데이터 저장 장치(100)는 비휘발성 메모리(1)와 디스크(2)를 포함할 수 있다. 비휘발성 메모리(1)에는 제1 메모리 테이블(11), 제2 메모리 테이블(13), 컴팩션 로그(15), 비-트리 인덱스(17)가 저장될 수 있다. 비휘발성 메모리(1)는 NVRAM(non-volatile memory)일 수 있고, 디스크(2)는 비휘발성 메모리(1)와는 구분되는 메모리의 일종일 수 있다. 예를 들어, 디스크(2)는 하드 디스크 또는 SSD일 수 있다.
- [0032] 데이터가 입력되면, 1차적으로 제1 메모리 테이블(11)에 저장될 수 있다. 제1 메모리 테이블(11)은 버퍼(uptade buffer)의 역할을 수행하는 구성으로서, 예를 들면, MemTable일 수 있다. 제1 메모리 테이블(11)은 일종의 스킵 리스트 데이터 스트럭처(skip list data structure)일 수 있다. 스킵 리스트 데이터 스트럭처는 점차 스퍼스해지는 데이터의 시퀀스를 연결하여 계층 구조로 정렬하는 데이터 저장 방식의 일종일 수 있다. 제1 메모리 테이블(11)은 영구적이며, 로깅(logging)을 피할 수 있다.
- [0033] 입력되는 모든 데이터는 제1 메모리 테이블(11)을 통해 데이터베이스 내에 업데이트 될 수 있다.
- [0034] 제1 메모리 테이블(11)의 용량이 소정 값 이상인 경우, 제1 메모리 테이블(11)에 저장된 적어도 일부의 데이터는 제2 메모리 테이블(13)에 저장될 수 있다. 예를 들어, 제1 메모리 테이블(11)의 용량은 기지정되어 있을 수 있고, 이러한 용량이 가득 차는 경우 제1 메모리 테이블(11)에 저장되는 정보는 제2 메모리 테이블(13)에 저장되어 제2 메모리 테이블(13)에 플러시 계획이 수립될 수 있다. 여기서, 플러시는 디스크 버전의 memtable인 SSTable로 제2 메모리 테이블(13)에 저장되는 정보를 저장하는 것을 의미할 수 있으며, 이와 관련된 내용은 통상의 기술자에게 용이한바 자세한 설명은 생략하겠다.
- [0035] 제2 메모리 테이블(13)은 제1 메모리 테이블(11)과 유사한 기능을 수행할 수 있다. 다만, 제2 메모리 테이블(13)은 불변적(immutable)이며, 읽는 동작만 가능하고 업데이트 또는 쓰기 동작이 불가능 할 수 있다. 제2 메모리 테이블(13)은 플러시 수행부(20)에 의해 플러시가 수행됨에 기초하여, 가득 찬 데이터가 삭제되면 새로운 memtable을 생성 할 수 있다.
- [0036] 보다 구체적으로, 제1 메모리 테이블(11)에 정보가 가득차게 되면, 제1 메모리 테이블(11)의 정보가 제2 메모리 테이블(13)에 저장되도록 하여 플러시가 수행되도록 하고, 제1 메모리 테이블(11)에 새로운 정보가 입력될 수 있다. 경우에 따라, 제1 메모리 테이블(11)에 정보가 가득차게 되면, 제1 메모리 테이블(11)을 제2 메모리 테이블(13)로 변환하여 플러시가 수행되도록 하고, 새로운 제1 메모리 테이블(11)이 생성되도록 할 수 있다.
- [0037] 만약 제2 메모리 테이블(13)에 대한 플러시가 수행되지 못하면, 플러시가 수행될 때까지 새로운 정보의 입력,

즉 새로운 제1 메모리 테이블(11)의 생성 또는 제1 메모리 테이블(11)에 대한 새로운 정보의 입력을 대기할 수 있다.

[0038] 플러시 수행부(20)에 의해 플러시가 수행되면, 제2 메모리 테이블(13)에 저장된 데이터 중 적어도 일부에 대해 제2 메모리 테이블(13)로부터 디스크(2)로 데이터가 전달될 수 있다.

[0039] 디스크(2)는 파일로서 SSTable을 포함할 수 있는데, 플러시가 수행되면 전달된 데이터(예: 키 값 페어(이하, KV 페어(pair))가 디스크(2)의 SSTable에 저장될 수 있다. SSTable에는 하나 또는 이상의 KV 페어가 저장될 수 있다. 보다 구체적으로, 플러시 된 데이터는 디스크(2)에서 싱글-레벨 형태의 SSTable에 저장될 수 있다. 싱글-레벨은 계층적인 구조의 멀티-레벨이 아닌 도시된 바와 같이 SSTable이 하나의 층(layer)를 이루어 저장되는 단일 레벨 구조를 의미할 수 있다.

[0040] 디스크(2)는 이러한 싱글-레벨 구조를 가지기 때문에 싱글-레벨로 저장된 데이터의 검색에 비-트리 인덱스(17)를 이용할 수 있다.

[0041] 보다 구체적으로 설명하면, 종래의 LSM 트리의 경우, 데이터를 멀티-레벨 구조, 즉, 계층적인 구조를 가지도록 저장하고 각 단계별 순차적으로 이동하면서 데이터를 검색하는 방식을 가진다. 반면에, 비-트리의 경우 싱글-레벨 구조로 저장된 데이터를 검색하도록 설정되어 있기 때문에, 디스크(2)는 비-트리 형태의 인덱스인 비-트리 인덱스(17)를 이용하여 데이터 검색을 수행할 수 있다. 이러한 경우, 디스크(2)에 저장되는 데이터가 정렬된 순서를 유지할 필요가 없으므로, 쓰기 증폭(write amplification)이 줄어들게 되어, 데이터 검색의 효율이 증가할 수 있다.

[0042] 여기서, 비-트리 인덱스(17)는 후술되는 디스크(2)에 저장되는 모든 데이터에 대한 인덱스를 가지는 데이터 스트럭처일 수 있다. 비-트리 인덱스(17)는 일반적인 비-트리와 유사한 형태를 가질 수 있다. 다만, 비휘발성 메모리(1) 내에 저장되어 영구적으로 보존될 수 있다.

[0043] 도 1에서는 디스크(2)에 6개의 SSTable이 존재하며 각각에 KV 페어가 하나씩 저장되는 경우의 예를 도시한다. 구체적으로, 디스크(2)에 KV 페어가 6개 저장되는 경우, 제1 KV 페어(21-1), 제2 KV 페어(21-2), 제3 KV 페어(21-3), 제4 KV 페어(21-4), 제5 KV 페어(21-5), 및 제6 KV 페어(21-6)가 각각 SSTable에 저장되어 하나의 층을 이룰 수 있다. 그러나, 이에 제한되는 것은 아니며, 하나의 SSTable에 두 개이상의 KV 페어가 저장될 수도 있다. 즉, 도시하지는 않았으나, 제1 KV 페어(21-1), 제2 KV 페어(21-2)가 하나의 SSTable에 저장되고, 제3 KV 페어(21-3), 제4 KV 페어(21-4), 제5 KV 페어(21-5), 및 제6 KV 페어(21-6)가 다른 하나의 SSTable에 저장될 수도 있다.

[0044] 한편, 주어진 키 값 범위 전체에 대한 스캔 동작(scan operation)에 대해 합리적인 성능을 제공하기 위해 디스크(2)에 저장된 KV 페어의 시퀀셜리티(sequentiality)가 소정 값 이상 유지될 필요가 있다. 여기서, 시퀀셜리티는 KV 페어가 디스크에 저장될 때 얼마나 잘 정렬된 순서로 저장되는지에 대한 정도를 의미할 수 있다. 이러한 시퀀셜리티는 후술되는 컴팩션이 수행됨으로써 일정 수준 이상으로 유지될 수 있다.

[0045] 한편, 디스크(2)에서 쓸모가 없어진 KV 페어는 가비지 컬렉션을 통해 삭제될 필요가 있다. 즉, 디스크(2)에 새로운 KV 페어가 들어와 업데이트가 수행되면 이전에 저장되어 있던 KV 페어의 이용 가치가 없어진다. 이에 따라, 이전에 저장되어 있던 KV 페어는 디스크(2)의 용량 측면을 고려하여 삭제될 필요가 있다.

[0046] 이러한 동작은 컴팩션으로 지칭될 수 있으며 컴팩션 수행부(23)는 컴팩션을 수행할 수 있다. 컴팩션 동작은 파일, 즉 SSTable마다 수행될 수 있으며, 컴팩션 동작에 기초하여 파일의 KV 페어들 중 쓸모가 없어진 KV 페어가 삭제되고, 남아있는 KV 페어는 키 값에 기초하여 정렬될 수 있다.

[0047] 보다 구체적으로 설명하면, KV 페어는 키와 해당 키에 대한 하나의 값의 형태로 나타날 수 있다. 예를 들어, 키가 5이고, 키에 대한 값이 100이면 <5, 100>로 표현될 수 있다. 이러한 경우, '5'라는 키를 요청하면, 100의 값을 획득할 수 있다. 경우에 따라, 키에 대한 값은 여러 번 업데이트될 수 있다. 예를 들어, '5'라는 키에 대해 현재 '100'이라는 값이 저장되어 있지만, 이후에 값을 '200'으로 업데이트할 수 있습니다. 이러한 경우, 처음에 <5,100>이라는 KV 페어가 파일 X에 쓰이면 비-트리가 키가 '5'인 경우에 대해서 X를 포인트(point)할 수 있다. 이후에 <5,200>이라는 KV 페어가 새로운 파일 Y에 쓰이면 비-트리가 키가 '5'인 경우에 대해서 Y를 포인트할 수 있다. 컴팩션 과정에서, 키가 5인 경우에 대해 비-트리가 어떤 파일을 포인트하는 지 확인할 수 있다. 이러한 경우, X 파일에 대해 컴팩션을 수행하는 경우에는 이전의 KV 페어는 삭제될 수 있다.

[0048] 다른 예를 들면, F1=(<1, 10>, <5, 20>, <10, 30>) 이고, F2=(<3, 30>, <4, 40>, <10, 100>) 이라고 하고

key=10에 대해서 F2의 <10, 100> 이 업데이트된 새로운 값이라고 가정하고 F1과 F2에 대한 컴팩션을 수행하면, F3, F4이라는 새로운 파일을 F3=<1, 10>, <3, 30>, <4, 40>, F4=<5, 20>, <10, 100>으로 생성할 수 있다. 이러한 과정에서 <10, 30>은 삭제될 수 있으며, F3, F4가 생성되어 디스크에 저장되면 기존의 F1, F2는 삭제될 수 있다.

- [0049] 비휘발성 메모리(1)에 저장되는 컴팩션 로그(15)는 컴팩션의 수행과 관련된 다양한 정보들을 기록할 수 있다.
- [0050] 데이터를 읽기 위한 입력이 인가되면, 비-트리 인덱스(17)를 기초로 데이터 검색이 이루어질 수 있다.
- [0051] 도 2는 본 발명의 일 실시예에 따른 싱글-레벨 데이터 저장 장치의 기능적 구성의 예를 도시한다. 이하 사용되는 '...부'등의 용어는 적어도 하나의 기능이나 동작을 처리하는 단위를 의미하며, 이는 하드웨어나 소프트웨어, 또는, 하드웨어 및 소프트웨어의 결합으로 구현될 수 있다. 이하 도 2의 설명에서는 도 1과 중복되는 내용은 생략될 수 있다.
- [0052] 도 2를 참조하면, 싱글-레벨 데이터 저장 장치(100)는 제1 메모리 테이블(110), 제2 메모리 테이블(120), 플러시 수행부(130), 싱글-레벨 저장부(140), 합병 정렬 수행부(150)를 포함할 수 있다.
- [0053] 제1 메모리 테이블(110)은 마이크로프로세서(microprocessor)를 포함하는 연산 장치에 의해 구현될 수 있으며, 이는 후술할 제2 메모리 테이블(120), 플러시 수행부(130), 싱글-레벨 저장부(140), 합병 정렬 수행부(150)에 있어서도 같다.
- [0054] 제1 메모리 테이블(110)은 비휘발성 메모리(1) 내에 포함되며, 데이터베이스의 형성을 위해 입력되는 데이터를 저장할 수 있다. 제1 메모리 테이블(110)은 도 1의 제1 메모리 테이블(11)에 대응하는 구성일 수 있다.
- [0055] 제2 메모리 테이블(120)은, 제1 메모리 테이블(110)에 저장된 데이터를 저장할 수 있다. 제2 메모리 테이블(120)은 immutable memtable로서, 제1 메모리 테이블(110)의 데이터를 불변적(immutable)으로 저장할 수 있다.
- [0056] 플러시 수행부(130)는 제2 메모리 테이블(120)에 저장된 데이터에 대해 플러시 동작을 수행할 수 있다. 플러시 수행부(130)는 제2 메모리 테이블(120)에 저장된 데이터를 디스크(2)로 플러시할 수 있다. 플러시가 수행되면, 그에 상응하는 데이터가 제2 메모리 테이블(120)에서 삭제될 수 있다.
- [0057] 싱글-레벨 저장부(140)는 플러시 동작에 기초하여 전달받은 데이터를 싱글-레벨로 저장할 수 있다. 구체적으로, 싱글-레벨 저장부(140)는 제2 메모리 테이블(120)에 저장된 데이터 중 적어도 일부를 비휘발성 메모리(1)와는 구분되는 디스크(2)에 싱글-레벨로 저장할 수 있다.
- [0058] 싱글-레벨 저장부(140)는 비-트리(B-tree) 데이터 구조에 기초하여 데이터가 검색되도록, 제2 메모리 테이블에 저장된 데이터 중 적어도 일부를 싱글-레벨로 저장할 수 있다.
- [0059] 컴팩션 수행부(150)는 싱글-레벨로 저장된 데이터 중 쓸모가 없어진 데이터에 대해 컴팩션을 수행할 수 있다. 컴팩션은 파일들을 키 값에 대한 합병 정렬(merge sort)을 수행하면서 새로운 파일을 만드는 과정을 의미할 수 있다.
- [0060] 합병 정렬을 수행하기에 앞서, 컴팩션 후보(compaction candidate)에 대한 리스트가 생성될 수 있으며, 이러한 리스트 안에 포함된 데이터에 대해 합병 정렬이 수행될 수 있다.
- [0061] 컴팩션 수행부(150)는 싱글-레벨로 저장된 데이터 중 필요가 없어진 데이터를 삭제하고, 나머지 데이터를 키 값에 기초하여 정렬할 수 있다. 삭제된 데이터는 컴팩션 후보로 추출되어 리스트로 생성되어 있을 수 있고, 리스트에 포함된 데이터 중 일부에 대해 합병 정렬이 수행될 수 있다. 경우에 따라, 데이터를 읽기 위한 입력이 인가되는 경우 비-트리 인덱스(17)를 이용하여 합병 정렬된 데이터에서 키 값을 기준으로 검색이 수행될 수 있다.
- [0062] 다른 예를 들면, 컴팩션 수행부(150)는 싱글-레벨로 저장된 데이터 중 저장된 시점이 업데이트 시점 이전의 데이터와 업데이트 시점 이후의 데이터의 비율을 식별할 수 있다. 여기서, 업데이트 시점 이전의 데이터는 예를 들면, 새로운 값이 쓰이거나(write) 업데이트된 경우, 그 이전의 값, 즉 유효하지 않은(invalid)(또는 obsolete) KV를 포함하고, 업데이트 시점 이후의 데이터는 새로운 값, 즉 유효한(valid) KV를 포함할 수 있다.
- [0063] 컴팩션 수행부(150)는 식별된 비율이 소정 값 보다 낮은 경우, 업데이트 시점 이전인 데이터 중 적어도 일부를 컴팩션 후보로 선택할 수 있다. 이에 따라, 컴팩션 수행부(150)는 업데이트 시점 이전인 데이터와 업데이트 시

점 이후의 데이터의 비율을 소정 값 이상으로 유지할 수 있다.

- [0064] 또 다른 예를 들면, 컴팩션 수행부(150)는 비-트리의 각 노드 중 소정 개수의 노드를 스캔할 수 있다. 이 때, 비-트리의 각 노드는 싱글-레벨로 저장된 데이터의 적어도 일부와 관련될 수 있고, 컴팩션 수행부(150)는 스캔된 노드와 관련된 데이터가 어떠한 데이터 테이블 파일에 분산되어 저장되는지를 식별할 수 있다. 데이터가 데이터 테이블 파일에 대해 분산된 정도가 소정 값 이상인 경우, 컴팩션 수행부(150)는 데이터가 저장된 데이터 테이블 파일 자체 또는 데이터 테이블 파일에 저장된 데이터를 컴팩션 후보로 선택할 수 있다.
- [0065] 여기서, 데이터 테이블 파일은 SStable 파일일 수 있고, KV 페어가 저장되는 파일을 의미할 수 있고, 컴팩션 수행부(150)의 비-트리 스캔 방식은 라운드-로빈(round-robin) 방식에 기초할 수 있으며, 이와 관련하여서는 통상의 기술자에게 용이한 바 생략하겠다.
- [0066] 또 다른 예를 들면, 컴팩션 수행부(150)는 싱글-레벨로 저장된 데이터를 읽기(read) 위해 요구되는 데이터 테이블 파일의 수를 산출하고, 산출된 데이터 테이블 파일의 수가 가장 큰 데이터와 관련된 데이터 테이블 파일을 컴팩션 후보로 선택할 수 있다.
- [0067] 컴팩션 수행부(150)는 컴팩션을 통해 쓸모 없는 KV 페어를 삭제할 수 있고, 그 과정에서 삭제되는 KV 페어를 포함하는 파일들에 있는 유효한 KV 페어를 선별하여 새로운 파일로 생성할 수 있다. 예를 들어, 두 개의 파일 F1, F2를 컴팩션하는 경우(이 때, F1=(<1, 10>, <5, 20>, <10, 30>) 이고, F2=(<3, 30>, <4, 40>, <10, 100>) 이라고 하고 key=10에 대해서 F2의 <10, 100> 이 업데이트된 새로운 값이라고 가정), 컴팩션 수행부(150)는 F3, F4라는 새로운 파일을 F3=(<1, 10>, <3, 30>, <4, 40>), F4=(<5, 20>, <10, 100>)으로 생성할 수 있다. 이러한 경우, 컴팩션 수행부(150)는 비-트리에서 키 1, 3, 4, 5, 10에 대해 F3과 F4를 포인트하도록 업데이트할 수 있다. 그 후 컴팩션 수행부(150)는 기존의 파일인 F1, F2를 삭제할 수 있다.
- [0068] 이러한 컴팩션이 수행됨으로 인해, 본 발명의 일 실시예에 따른 싱글-레벨 기반의 데이터 저장 장치(100)는 선택적으로 컴팩션을 수행하여 적절한 스캔 동작 성능을 제공하는 동시에, 디스크(2) 공간에 불필요한 데이터가 축적되는 것을 예방할 수 있다. 즉, 싱글-레벨 기반의 데이터 저장 장치(100)는 데이터의 읽기/쓰기 동작이 보다 효율적으로 수행되며, 디스크(2) 공간에 대한 활용도를 향상시킬 수 있다.
- [0069] 도 3은 본 발명의 일 실시예에 따른 싱글-레벨 데이터 저장 방법의 각 단계의 흐름을 도시한다. 도 3에 도시된 방법의 각 단계는 경우에 따라 도면에 도시된 바와 그 순서를 달리하여 수행될 수 있음은 물론이다.
- [0070] 도 3을 참조하면, 비휘발성 메모리(1)에 포함된 제1 메모리 테이블(110)에 데이터가 저장될 수 있다(S110). 제1 메모리 테이블(110)의 데이터는 제2 메모리 테이블(120)에 저장될 수 있다. 제2 메모리 테이블(120)도 비휘발성 메모리(1)에 포함되는 것으로, 제1 메모리 테이블(110)에 저장된 데이터 일부를 불변적으로 저장할 수 있다.
- [0071] 제2 메모리 테이블(120)에 저장된 데이터에 대해 플러시 동작이 수행될 수 있다(S130). 예를 들어, 제2 메모리 테이블(120)에 데이터가 가득 차는 경우, 플러시 동작이 수행될 수 있다(S130).
- [0072] 플러시 동작에 기초하여 제2 메모리 테이블(120)에서 디스크(2)로 데이터가 전달되며, 전달된 데이터가 싱글-레벨로 저장될 수 있다(S140).
- [0073] 싱글-레벨로 데이터가 저장되고, 저장된 데이터 중 업데이트가 수행되어 쓸모없는 데이터가 생기면 컴팩션이 수행될 수 있다(S150). 컴팩션이 수행될 때 합병 정렬이 수행될 수 있다. 합병 정렬이 수행될 데이터는 다양한 방법으로 결정될 수 있다.
- [0074] 예를 들어, 싱글-레벨로 저장된 데이터 중 쓸모없는 데이터가 삭제되고, 남은 데이터는 키 값에 기초하여 정렬될 수 있다. 쓸모없는 데이터는 업데이트 되기 전의 데이터일 수 있다.
- [0075] 다른 예를 들면, 싱글-레벨로 저장된 데이터 중 업데이트 시점 이전의 데이터와 업데이트 이후의 데이터(유효한(live) 데이터)의 비율을 식별하고, 식별된 비율이 소정 값 보다 낮은 경우, 소정 시점 이전인 데이터와 유효한 데이터의 비율이 소정 값을 초과하도록 오래된 데이터를 삭제한 후 남은 데이터를 정렬할 수 있다.
- [0076] 또 다른 예를 들면, 비-트리의 각 노드 중 소정 개수의 노드를 스캔하여, 스캔된 노드에 해당하는 구간에 존재하는 데이터가 몇 개의 데이터 테이블 파일에 분포하고 있는지를 식별할 수 있다. 이러한 데이터 테이블 파일에 대한 분포의 정도가 소정 값 이상인 경우, 즉 스캔된 노드에 해당하는 구간에 존재하는 데이터가 일정 개수 이상의 데이터 테이블 파일에 분포하고 있는 경우, 데이터가 분포되는 데이터 테이블 파일 자체를 컴팩션 후보로 선택할 수 있다.

- [0077] 또 다른 예를 들면, 싱글-레벨로 저장된 데이터를 읽기(read) 위해 요구되는 데이터 테이블 파일의 수를 산출하고, 산출된 데이터 테이블 파일의 수가 가장 큰 데이터의 단위를 컴팩션 후보로 선택할 수 있다. 한편, 여기서 읽기는 하나의 키에 대한 값을 읽는 것(point query)이 아니라 range query 또는 scan으로 주어진 범위의 key들, 예를 들면 200 내지 300의 범위의 키들에 대한 값들을 모두 읽는 동작일 수 있다. 이러한 읽기가 수행되는 경우, scan하는 범위를 서브-범위(sub-range)로 나누어서, 한 서브-범위를 읽기 위해 액세스한 파일들의 수를 기록하고 있다가, 액세스한 파일들의 수가 가장 큰 서브-범위에 해당하는 파일을 컴팩션 후보로 선택할 수 있다.
- [0078] 도 4는 본 발명의 일 실시예에 따른 싱글-레벨 데이터 저장 방법의 읽기(read) 단계의 흐름을 개념적으로 도시한다.
- [0079] 도 4를 참조하면, 따른 싱글-레벨 데이터 저장 장치(100)는 데이터를 읽기 위해, 첫번째 단계로 제1 메모리 테이블(110)에서 데이터를 검색할 수 있다. 만약 제1 메모리 테이블(110)에서 데이터를 찾지 못하는 경우, 제2 메모리 테이블(120)에서 데이터를 검색할 수 있다.
- [0080] 제2 메모리 테이블(120)에서도 데이터를 찾지 못하는 경우, 비-트리 구조에 기반하여 디스크(2)에서 데이터를 검색할 수 있다. 한편, 데이터를 포함하는 파일은 블록(block)들로 구성될 수 있고, 비-트리의 인덱스는 키에 대해 디스크에 저장된 파일 정보, 예를 들어 키가 저장된 파일의 블록의 오프셋 정보를 포함할 수 있다. 이에 따라, 비-트리 구조, 즉, 비-트리의 인덱스를 이용하여 데이터를 검색하는 경우, 보다 용이하게 KV의 검색이 수행되도록 할 수 있다.
- [0081] 상술한 바와 같이, 데이터는 싱글-레벨 형태로 디스크(2)에 저장되기 때문에, 비-트리 구조를 이용하여 검색이 수행될 수 있다.
- [0082] 도 5는 본 발명의 일 실시예에 따른 싱글-레벨 데이터 저장 방법의 쓰기(write) 단계의 흐름을 개념적으로 도시한다.
- [0083] 도 5를 참조하면, 입력되는 모든 데이터는 제1 메모리 테이블(110)에 인가될 수 있다. 제1 메모리 테이블(110)이 가득 찬 경우, 제1 메모리 테이블(110)에 저장된 적어도 일부의 데이터는 제2 메모리 테이블(120)에 인가되어 불변적인 형태로 저장될 수 있다. 그 후 제2 메모리 테이블(120)의 상태에 따라 플러시가 수행될 수 있고, 경우에 따라서는 플러시되어 싱글-레벨로 저장된 데이터에 대해 컴팩션이 수행될 수 있다.
- [0084] 플러시가 수행되는 경우, 싱글-레벨로 디스크(2)에 저장되는 데이터에 대한 정보, 예를 들면 파일과 오프셋에 대한 정보는 비-트리에 인덱스 형태로 저장되거나 업데이트될 수 있다. 이에 기초하여, 컴팩션이 수행되는 경우 비-트리로 검색을 수행하여 컴팩션의 대상이 되는 파일에 있는 KV 페어가 유효한지 여부를 확인할 수 있다.
- [0085] 본 발명의 일 실시예에 따른 싱글-레벨 데이터 저장 장치 및 방법은, 싱글-레벨로 데이터를 저장하여 데이터의 검색이 신속하게 수행되도록 함으로 효율적으로 검색이 이루어지도록 하며, 또한 컴팩션 후보를 결정하고 컴팩션 후보 중 일부에 대해 합병 정렬을 수행하여 불필요한 데이터의 저장을 방지함으로써 효율적으로 데이터 저장되도록 한다.
- [0086] 본 발명의 일 실시예에 따른 싱글-레벨 데이터 저장 장치 및 방법은, 제1 메모리 테이블(110)과 제2 메모리 테이블(120)에 데이터를 저장함으로써 고성능의 데이터 쓰기가 가능하며, 비-트리를 이용함으로써 고성능의 데이터 읽기가 가능하도록 한다.
- [0087] 본 명세서에 첨부된 블록도의 각 블록과 흐름도의 각 단계의 조합들은 컴퓨터 프로그램 인스트럭션들에 의해 수행될 수도 있다. 이들 컴퓨터 프로그램 인스트럭션들은 범용 컴퓨터, 특수용 컴퓨터 또는 기타 프로그램 가능한 데이터 프로세싱 장비의 프로세서에 탑재될 수 있으므로, 컴퓨터 또는 기타 프로그램 가능한 데이터 프로세싱 장비의 프로세서를 통해 수행되는 그 인스트럭션들이 블록도의 각 블록 또는 흐름도의 각 단계에서 설명된 기능들을 수행하는 수단을 생성하게 된다. 이들 컴퓨터 프로그램 인스트럭션들은 특정 방식으로 기능을 구현하기 위해 컴퓨터 또는 기타 프로그램 가능한 데이터 프로세싱 장비를 지향할 수 있는 컴퓨터 이용 가능 또는 컴퓨터 판독 가능 메모리에 저장되는 것도 가능하므로, 그 컴퓨터 이용가능 또는 컴퓨터 판독 가능 메모리에 저장된 인스트럭션들은 블록도의 각 블록 또는 흐름도 각 단계에서 설명된 기능을 수행하는 인스트럭션 수단을 내포하는 제조 품목을 생산하는 것도 가능하다. 컴퓨터 프로그램 인스트럭션들은 컴퓨터 또는 기타 프로그램 가능한 데이터 프로세싱 장비 상에 탑재되는 것도 가능하므로, 컴퓨터 또는 기타 프로그램 가능한 데이터 프로세싱 장비 상에서 일련의 동작 단계들이 수행되어 컴퓨터로 실행되는 프로세스를 생성해서 컴퓨터 또는 기타 프로그램 가능

한 데이터 프로세싱 장비를 수행하는 인스트럭션들은 블록도의 각 블록 및 흐름도의 각 단계에서 설명된 기능들을 실행하기 위한 단계들을 제공하는 것도 가능하다.

[0088] 또한, 각 블록 또는 각 단계는 특정된 논리적 기능(들)을 실행하기 위한 하나 이상의 실행 가능한 인스트럭션들을 포함하는 모듈, 세그먼트 또는 코드의 일부를 나타낼 수 있다. 또, 몇 가지 대체 실시예들에서는 블록들 또는 단계들에서 언급된 기능들이 순서를 벗어나서 발생하는 것도 가능함을 주목해야 한다. 예컨대, 잇달아 도시되어 있는 두 개의 블록들 또는 단계들은 사실 실질적으로 동시에 수행되는 것도 가능하고 또는 그 블록들 또는 단계들이 때때로 해당하는 기능에 따라 역순으로 수행되는 것도 가능하다.

[0089] 이상의 설명은 본 발명의 기술 사상을 예시적으로 설명한 것에 불과한 것으로서, 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자라면 본 발명의 본질적인 품질에서 벗어나지 않는 범위에서 다양한 수정 및 변형이 가능할 것이다. 따라서, 본 명세서에 개시된 실시예들은 본 발명의 기술 사상을 한정하기 위한 것이 아니라 설명하기 위한 것이고, 이러한 실시예에 의하여 본 발명의 기술 사상의 범위가 한정되는 것은 아니다. 본 발명의 보호 범위는 아래의 청구범위에 의하여 해석되어야 하며, 그와 균등한 범위 내에 있는 모든 기술사상은 본 발명의 권리범위에 포함되는 것으로 해석되어야 할 것이다.

부호의 설명

[0090] 1: 비휘발성 메모리

2: 디스크

100: 싱글-레벨 데이터 저장 장치

110: 제1 메모리 테이블

120: 제2 메모리 테이블

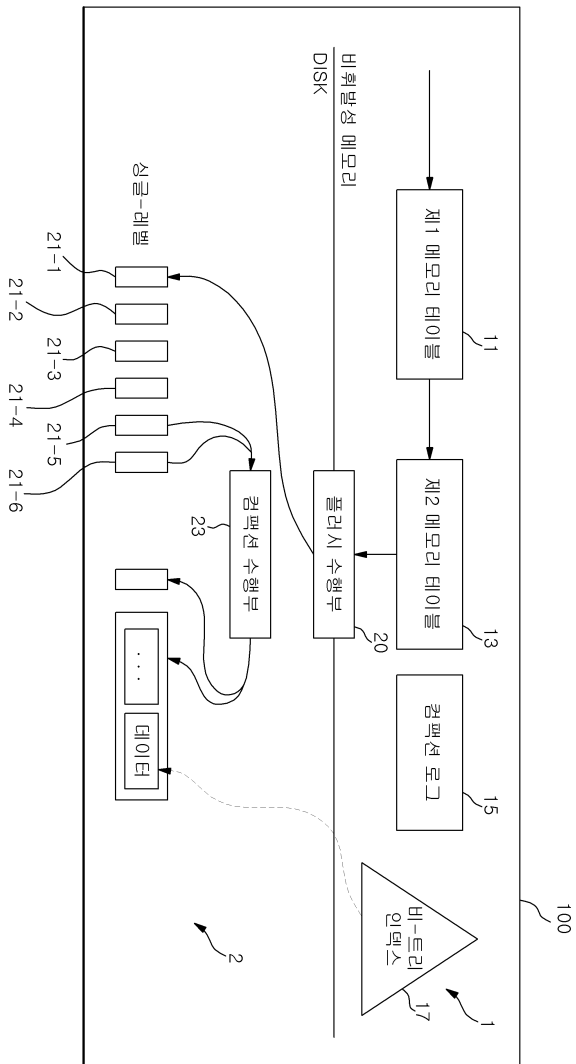
130: 플러시 수행부

140: 싱글-레벨 저장부

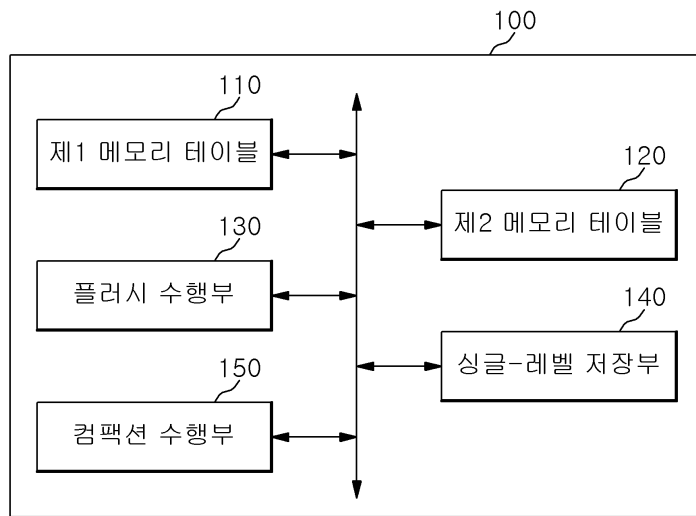
150: 컴팩션 수행부

도면

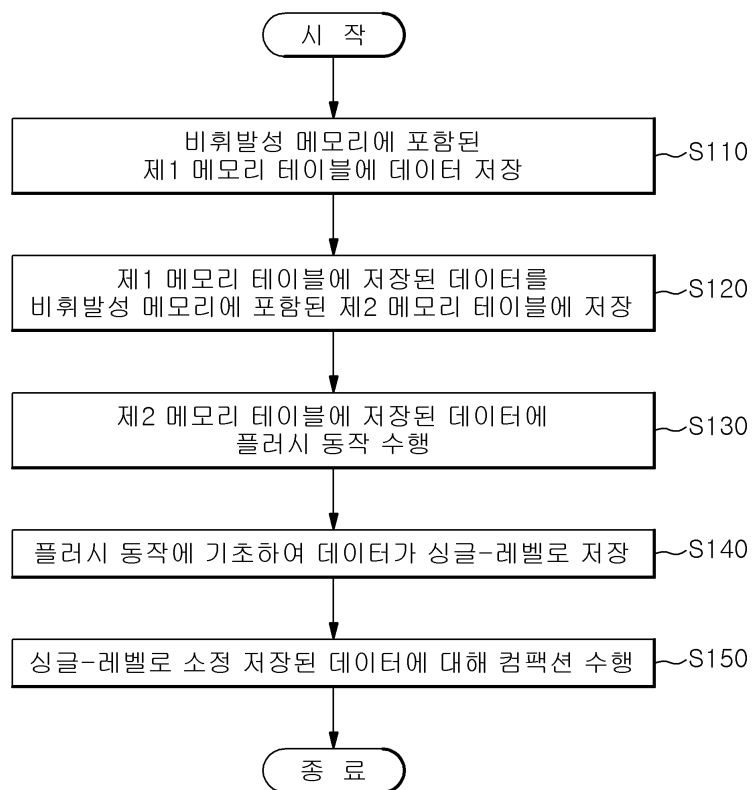
도면1



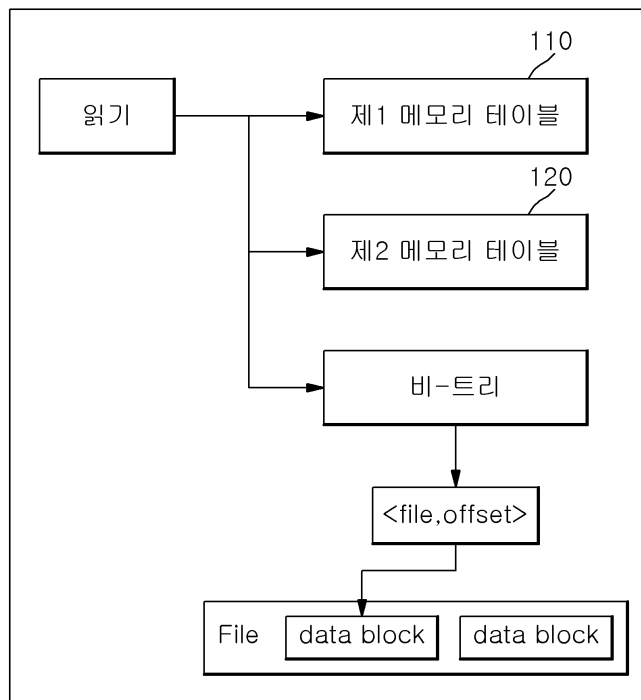
도면2



도면3



도면4



도면5

