(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2025년01월21일
(11) 등록번호 10-2758237
(24) 등록일자 2025년01월17일

(51) 국제특허분류(Int. Cl.)
G06F 3/06 (2006.01)
(52) CPC특허분류
G06F 3/0604 (2013.01)
G06F 3/0614 (2013.01)
(21) 출원번호 10-2022-0151605
(22) 출원일자 2022년11월14일
심사청구일자 2022년11월14일
(65) 공개번호 10-2024-0070139
(43) 공개일자 2024년05월21일
(56) 선행기술조사문헌
JP2013196646 A
KR1020210054440 A
KR102287774 B1

(73) 특허권자
성균관대학교산학협력단
경기도 수원시 장안구 서부로 2066 (천천동, 성균관대학교내)
(72) 발명자
신동군
서울특별시 강남구 역삼로 314 개나리푸르지오 305동 1004호
정지윤
경기도 수원시 장안구 화산로 85 천천푸르지오 101동 1502호
(74) 대리인
김준석, 박민욱

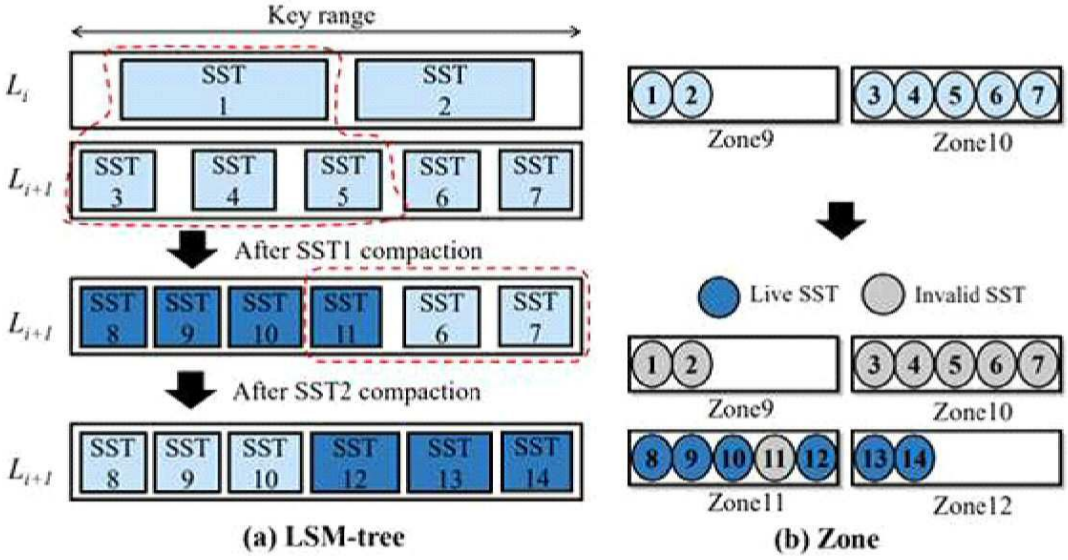
전체 청구항 수 : 총 10 항 심사관 : 한현명

(54) 발명의 명칭 **ZNS기반 SSD의 컴팩션 방법 및 그 방법이 적용된 컴팩션 장치**

(57) 요 약

본 발명의 일 실시예에 따른 ZNS기반 SSD의 컴팩션 방법은 LSM-tree에서 소정의 기준에 따라 컴팩션 대상이 되는 레벨인 대상레벨(L_i)을 결정하는 단계; 상기 대상레벨(L_i)에서 컴팩션 포인터(Compaction Pointer, CP_i)의 위치에 대응되는 SST인 대상SST(T^i_j)에 기반하여, 병합될 SST의 집합인 병합대상집합과 컴팩션에 포함되는 키 레인지를 나타내는 컴팩션 윈도우를 초기화하는 단계; 상기 컴팩션 포인터(CP_i)의 다음 컴팩션 포인터 위치, 상기 컴팩션 윈도우 및 상기 대상레벨(L_i)의 하위레벨(L_{i+1})에 포함된 복수의 SST(T^{i+1}_k)를 이용하여, 상기 병합대상집합 및 상기 컴팩션윈도우를 갱신하는 단계; 및 상기 병합대상집합을 이용하여 컴팩션을 수행하는 단계;를 포함한다.

대 표 도 - 도4



(52) CPC특허분류
G06F 3/0679 (2013.01)
G06F 2212/7202 (2013.01)

이 발명을 지원한 국가연구개발사업

과제고유번호	1711152452
과제번호	2017-0-00914-006
부처명	과학기술정보통신부
과제관리(전문)기관명	정보통신기획평가원
연구사업명	SW컴퓨팅산업원천기술개발
연구과제명	(SW 스타랩) 지능형 IoT 장치용 소프트웨어 프레임워크
기 여 율	1/1
과제수행기관명	성균관대학교 산학협력단
연구기간	2022.01.01 ~ 2022.12.31

공지예외적용 : 있음

명 세 서

청구범위

청구항 1

LSM-tree에서 소정의 기준에 따라 컴팩션 대상이 되는 레벨인 대상레벨(L_i)을 결정하는 단계;

상기 대상레벨(L_i)에서 컴팩션 포인터(Compaction Pointer, CP_i)의 위치에 대응되는 SST인 대상SST(T^i_j)에 기반하여, 병합될 SST의 집합인 병합대상집합과 컴팩션에 포함되는 키 레인지를 나타내는 컴팩션 윈도우를 초기화하는 단계;

상기 컴팩션 포인터(CP_i)의 다음 컴팩션 포인터 위치, 상기 컴팩션 윈도우 및 상기 대상레벨(L_i)의 하위레벨(L_{i+1})에 포함된 복수의 SST(T^{i+1}_k)를 이용하여, 상기 병합대상집합 및 상기 컴팩션윈도우를 갱신하는 단계; 및

상기 병합대상집합을 이용하여 컴팩션을 수행하는 단계;

를 포함하는 것을 특징으로 하는 ZNS기반 SSD의 컴팩션 방법.

청구항 2

제1항에 있어서,

상기 병합대상집합 및 상기 컴팩션윈도우를 갱신하는 단계는

상기 복수의 SST(T^{i+1}_k) 중에서 키 레인지의 전부 또는 일부가 상기 컴팩션 윈도우와 중복되는 SST인 제1 SST 및 상기 대상SST(T^i_j)의 최대키(largest key)값보다 최소키(smallest key)값이 크고, 상기 다음 컴팩션 포인터 위치보다 최대키(largest key)값이 작은 SST인 제2 SST을 이용하여, 상기 병합대상집합 및 상기 컴팩션윈도우를 갱신하는 것을 특징으로 하는 ZNS기반 SSD의 컴팩션 방법.

청구항 3

LSM-tree에서 소정의 기준에 따라 컴팩션 대상이 되는 레벨인 대상레벨(L_i)을 결정하는 단계;

상기 대상레벨(L_i)에서 컴팩션 포인터(CP_i)의 위치에 대응되는 SST인 대상SST(T^i_j)에 기반하여, 병합될 SST의 집합인 병합대상집합과 컴팩션에 포함되는 키 레인지를 나타내는 컴팩션 윈도우를 초기화하는 단계;

상기 컴팩션 윈도우 및 상기 대상레벨(L_i)의 하위레벨(L_{i+1})에 포함된 복수의 SST(T^{i+1}_k)를 이용하여, 상기 병합대상집합 및 상기 컴팩션윈도우를 갱신하는 단계; 및

상기 병합대상집합을 이용하여 컴팩션을 수행하는 단계;

를 포함하고,

상기 컴팩션을 수행하는 단계는

컴팩션 과정에서 생성되는 SST가 상기 컴팩션 포인터(CP_i)의 다음 컴팩션 포인터 위치를 포함하면, 상기 SST를 상기 다음 컴팩션 포인터 위치를 기준으로 2개의 SST로 분리하는 것을 특징으로 하는 ZNS기반 SSD의 컴팩션 방법.

청구항 4

제3항에 있어서,

상기 컴팩션을 수행하는 단계는

컴팩션 과정에서 생성되는 SST가 상기 하위레벨(L_{i+1})의 컴팩션 포인터(CP_{i+1})의 위치를 포함하면, 상기 SST를 상기 하위레벨(L_{i+1})의 컴팩션 포인터(CP_{i+1}) 위치를 기준으로 2개의 SST로 분리하는 것을 특징으로 하는 ZNS기반 SSD의 컴팩션 방법.

청구항 5

제3항에 있어서,

상기 컴팩션을 수행하는 단계는

상기 분리된 2개의 SST중에서, 상기 다음 컴팩션 포인터 위치의 이후에 위치하는 SST를 임시로 SST를 저장하기 위하여 설정된 T-zone(Temporary zone)에 분리하여 저장하는 것을 특징으로 하는 ZNS기반 SSD의 컴팩션 방법

청구항 6

LSM-tree에서 소정의 기준에 따라 컴팩션 대상이 되는 레벨인 대상레벨(L_i)을 결정하는 대상결정부;

상기 대상레벨(L_i)에서 컴팩션 포인터(CP_i)의 위치에 대응되는 SST인 대상SST(T^i_j)에 기반하여, 병합될 SST의 집합인 병합대상집합과 컴팩션에 포함되는 키 레인지를 나타내는 컴팩션 윈도우를 초기화하고,

상기 컴팩션 포인터(CP_i)의 다음 컴팩션 포인터 위치, 상기 컴팩션 윈도우 및 상기 대상레벨(L_i)의 하위레벨(L_{i+1})에 포함된 복수의 SST(T^{i+1}_k)를 이용하여, 상기 병합대상집합 및 상기 컴팩션윈도우를 갱신하는 컴팩션연산부; 및

상기 병합대상집합을 이용하여 컴팩션을 수행하는 컴팩션수행부;

를 포함하는 것을 특징으로 하는 라이프타임-레벨링을 수행하는 컴팩션 장치.

청구항 7

제6항에 있어서,

상기 컴팩션연산부는

상기 복수의 SST(T^{i+1}_k) 중에서 키 레인지의 전부 또는 일부가 상기 컴팩션 윈도우와 중복되는 SST인 제1 SST 및 상기 대상SST(T^i_j)의 최대키값보다 최소키값이 크고, 상기 다음 컴팩션 포인터 위치보다 최대키값이 작은 SST인 제2 SST을 이용하여, 상기 병합대상집합 및 상기 컴팩션윈도우를 갱신하는 것을 특징으로 하는 라이프타임-레벨링을 수행하는 컴팩션 장치.

청구항 8

LSM-tree에서 소정의 기준에 따라 컴팩션 대상이 되는 레벨인 대상레벨(L_i)을 결정하는 대상결정부;

상기 대상레벨(L_i)에서 컴팩션 포인터(CP_i)의 위치에 대응되는 SST인 대상SST(T^i_j)에 기반하여, 병합될 SST의 집합인 병합대상집합과 컴팩션에 포함되는 키 레인지를 나타내는 컴팩션 윈도우를 초기화하고,

상기 컴팩션 윈도우 및 상기 대상레벨(L_i)의 하위레벨(L_{i+1})에 포함된 복수의 SST(T^{i+1}_k)를 이용하여, 상기 병합대상집합 및 상기 컴팩션윈도우를 갱신하는 컴팩션연산부; 및

상기 병합대상집합을 이용하여 컴팩션을 수행하는 컴팩션수행부;

를 포함하고,

상기 컴팩션수행부는

컴팩션 과정에서 생성되는 SST가 상기 컴팩션 포인터(CP_i)의 다음 컴팩션 포인터 위치를 포함하면, 상기 SST를 상기 다음 컴팩션 포인터 위치를 기준으로 2개의 SST로 분리하는 것을 특징으로 하는 라이프타임-레벨링을 수행

하는 컴팩션 장치.

청구항 9

제8항에 있어서,

상기 컴팩션수행부는

컴팩션 과정에서 생성되는 SST가 상기 하위레벨(L_{i+1})의 컴팩션 포인터(CP_{i+1})의 위치를 포함하면, 상기 SST를 상기 하위레벨(L_{i+1})의 컴팩션 포인터(CP_{i+1}) 위치를 기준으로 2개의 SST로 분리하는 것을 특징으로 하는 라이프타임-레벨링을 수행하는 컴팩션 장치.

청구항 10

제8항에 있어서,

상기 컴팩션수행부는

상기 분리된 2개의 SST중에서, 상기 다음 컴팩션 포인터 위치의 이후에 위치하는 SST를 임시로 SST를 저장하기 위하여 설정된 T-zone(Temporary zone)에 분리하여 저장하는 것을 특징으로 하는 라이프타임-레벨링을 수행하는 컴팩션 장치.

발명의 설명

기 술 분 야

[0001] 본 발명은 ZNS(Zoned Namespace)기반의 SSD(Solid State Drive)에서 SST(Sorted String Table)의 라이프타임을 일정하게 유지하기 위한 컴팩션(Compaction) 방법 및 그 방법이 적용된 컴팩션 장치에 관한 것이다.

배 경 기 술

[0002] ZNS기반 SSD는 Zoned storage로, 스토리지가 여러 개의 Zone을 host에게 제공하며, Zone 내부에서는 순차적 쓰기(Sequential write)만 가능하다는 특징을 가진다. 또한 Zone을 재활용하기 위해서는 Zone 전체를 리셋(reset)해야 하는 특징을 가진다.

[0003] 한편, LSM(Log-structed Merge) tree는 순차적 쓰기만을 활용하는 SST(Sorted String Table) 파일을 통해 Database를 관리한다. LSM-tree의 순차적 쓰기만을 활용하는 특징은 ZNS기반 SSD의 제약조건에 상응한다. 만약 SST의 크기가 Zone의 크기보다 작아진다면 하나의 Zone 내부에 여러 개의 SST가 들어가게 될 수 있다. 위와 같은 상황에서 Zone 내부의 SST들은 각각 다른 라이프타임(Lifetime)을 가지게 되며, 하나의 Zone에는 Valid한 SST와 Invalid한 SST가 동시에 존재할 수 있다. 이는 Zone 내부의 Valid SST 비율인 Zone utilization을 감소시키게 되며, 스토리지 전체의 Space Amplification을 크게 증가시킬 수 있다.

[0004] 따라서, LSM-tree를 ZNS기반 SSD에서 사용할 때 Zone 내부 SST의 라이프타임을 일정하게 유지하기 위한 라이프타임-레벨링(Lifetime-Leveling) 컴팩션 방법 및 그 방법이 적용된 컴팩션 장치에 대한 필요성이 대두되고 있다.

발명의 내용

해결하려는 과제

[0005] 본 발명은 Zone 내부의 SST의 라이프타임을 일정하게 유지하여 존 이용도(Zone utilization)를 증가시키고, 공간 증폭(Space Amplification)을 억제하는 ZNS기반 SSD의 컴팩션 방법 및 그 방법이 적용된 컴팩션 장치를 제공하고자 한다.

과제의 해결 수단

[0006] 상술한 목적을 달성하기 위한 본 발명의 일 실시예에 따른 ZNS기반 SSD의 컴팩션 방법은 LSM-tree에서 소정의 기준에 따라 컴팩션 대상이 되는 레벨인 대상레벨(L_i)을 결정하는 단계; 상기 대상레벨(L_i)에서 컴팩션 포인터

(Compaction Pointer, CP_i)의 위치에 대응되는 SST인 대상SST(T^i_j)에 기반하여, 병합될 SST의 집합인 병합대상집합과 컴팩션에 포함되는 키 레인지를 나타내는 컴팩션 윈도우를 초기화하는 단계; 상기 컴팩션 포인터(CP_i)의 다음 컴팩션 포인터 위치, 상기 컴팩션 윈도우 및 상기 대상레벨(L_i)의 하위레벨(L_{i+1})에 포함된 복수의 SST(T^{i+1}_k)를 이용하여, 상기 병합대상집합 및 상기 컴팩션윈도우를 갱신하는 단계; 및 상기 병합대상집합을 이용하여 컴팩션을 수행하는 단계;를 포함한다.

[0007] 바람직하게는, 상기 병합대상집합 및 상기 컴팩션윈도우를 갱신하는 단계는 상기 복수의 SST(T^{i+1}_k) 중에서 키 레인지의 전부 또는 일부가 상기 컴팩션 윈도우와 중복되는 SST인 제1 SST 및 상기 대상SST(T^i_j)의 최대키(largest key)값보다 최소키(smallest key)값이 크고, 상기 다음 컴팩션 포인터 위치보다 최대키(largest key)값이 작은 SST인 제2 SST을 이용하여, 상기 병합대상집합 및 상기 컴팩션윈도우를 갱신할 수 있다.

[0008] 또한, 상술한 목적을 달성하기 위한 본 발명의 다른 실시예에 따른 ZNS기반 SSD의 컴팩션 방법은 LSM-tree에서 소정의 기준에 따라 컴팩션 대상이 되는 레벨인 대상레벨(L_i)을 결정하는 단계; 상기 대상레벨(L_i)에서 컴팩션 포인터(CP_i)의 위치에 대응되는 SST인 대상SST(T^i_j)에 기반하여, 병합될 SST의 집합인 병합대상집합과 컴팩션에 포함되는 키 레인지를 나타내는 컴팩션 윈도우를 초기화하는 단계; 상기 컴팩션 윈도우 및 상기 대상레벨(L_i)의 하위레벨(L_{i+1})에 포함된 복수의 SST(T^{i+1}_k)를 이용하여, 상기 병합대상집합 및 상기 컴팩션윈도우를 갱신하는 단계; 및 상기 병합대상집합을 이용하여 컴팩션을 수행하는 단계;를 포함하고, 상기 컴팩션을 수행하는 단계는 컴팩션 과정에서 생성되는 SST가 상기 컴팩션 포인터(CP_i)의 다음 컴팩션 포인터 위치를 포함하면, 상기 SST를 상기 다음 컴팩션 포인터 위치를 기준으로 2개의 SST로 분리한다.

[0009] 바람직하게는, 상기 컴팩션을 수행하는 단계는 컴팩션 과정에서 생성되는 SST가 상기 하위레벨(L_{i+1})의 컴팩션 포인터(CP_{i+1})의 위치를 포함하면, 상기 SST를 상기 하위레벨(L_{i+1})의 컴팩션 포인터(CP_{i+1}) 위치를 기준으로 2개의 SST로 분리할 수 있다.

[0010] 바람직하게는, 상기 컴팩션을 수행하는 단계는 상기 분리된 2개의 SST중에서, 상기 다음 컴팩션 포인터 위치의 이후에 위치하는 SST를 임시로 SST를 저장하기 위하여 설정된 T-zone(Temporary zone)에 분리하여 저장할 수 있다.

[0011] 또한, 상술한 목적을 달성하기 위한 본 발명의 일 실시예에 따른 라이프타임-레벨링을 수행하는 컴팩션 장치는 LSM-tree에서 소정의 기준에 따라 컴팩션 대상이 되는 레벨인 대상레벨(L_i)을 결정하는 대상결정부; 상기 대상레벨(L_i)에서 컴팩션 포인터(CP_i)의 위치에 대응되는 SST인 대상SST(T^i_j)에 기반하여, 병합될 SST의 집합인 병합대상집합과 컴팩션에 포함되는 키 레인지를 나타내는 컴팩션 윈도우를 초기화하고, 상기 컴팩션 포인터(CP_i)의 다음 컴팩션 포인터 위치, 상기 컴팩션 윈도우 및 상기 대상레벨(L_i)의 하위레벨(L_{i+1})에 포함된 복수의 SST(T^{i+1}_k)를 이용하여, 상기 병합대상집합 및 상기 컴팩션윈도우를 갱신하는 컴팩션연산부; 및 상기 병합대상집합을 이용하여 컴팩션을 수행하는 컴팩션수행부;를 포함한다.

[0012] 바람직하게는, 상기 컴팩션연산부는 상기 복수의 SST(T^{i+1}_k) 중에서 키 레인지의 전부 또는 일부가 상기 컴팩션 윈도우와 중복되는 SST인 제1 SST 및 상기 대상SST(T^i_j)의 최대키값보다 최소키값이 크고, 상기 다음 컴팩션 포인터 위치보다 최대키값이 작은 SST인 제2 SST을 이용하여, 상기 병합대상집합 및 상기 컴팩션윈도우를 갱신할 수 있다.

[0013] 또한, 상술한 목적을 달성하기 위한 본 발명의 다른 실시예에 따른 라이프타임-레벨링을 수행하는 컴팩션 장치는 LSM-tree에서 소정의 기준에 따라 컴팩션 대상이 되는 레벨인 대상레벨(L_i)을 결정하는 대상결정부; 상기 대상레벨(L_i)에서 컴팩션 포인터(CP_i)의 위치에 대응되는 SST인 대상SST(T^i_j)에 기반하여, 병합될 SST의 집합인 병합대상집합과 컴팩션에 포함되는 키 레인지를 나타내는 컴팩션 윈도우를 초기화하고, 상기 컴팩션 윈도우 및 상

기 대상레벨(L_i)의 하위레벨(L_{i+1})에 포함된 복수의 SST(T^{i+1}_k)를 이용하여, 상기 병합대상집합 및 상기 컴팩션윈도우를 갱신하는 컴팩션연산부; 및 상기 병합대상집합을 이용하여 컴팩션을 수행하는 컴팩션수행부;를 포함하고, 상기 컴팩션수행부는 컴팩션 과정에서 생성되는 SST가 상기 컴팩션 포인터(CP_i)의 다음 컴팩션 포인터 위치를 포함하면, 상기 SST를 상기 다음 컴팩션 포인터 위치를 기준으로 2개의 SST로 분리할 수 있다.

[0014] 바람직하게는, 상기 컴팩션수행부는 컴팩션 과정에서 생성되는 SST가 상기 하위레벨(L_{i+1})의 컴팩션 포인터(CP_{i+1})의 위치를 포함하면, 상기 SST를 상기 하위레벨(L_{i+1})의 컴팩션 포인터(CP_{i+1}) 위치를 기준으로 2개의 SST로 분리할 수 있다.

[0015] 바람직하게는, 상기 컴팩션수행부는 상기 분리된 2개의 SST중에서, 상기 다음 컴팩션 포인터 위치의 이후에 위치하는 SST를 임시로 SST를 저장하기 위하여 설정된 T-zone(Temporary zone)에 분리하여 저장할 수 있다.

발명의 효과

[0016] 본 발명은 ZNS기반 SSD에서 Zone 내부의 SST의 라이프타임을 일정하게 유지하여 존 이용도(Zone utilization)를 증가시키고, 공간 증폭(Space Amplification)을 억제하는 효과가 있다.

도면의 간단한 설명

[0017] 도 1은 본 발명의 일 실시예에 따른 ZNS기반 SSD의 컴팩션 방법을 설명하기 위한 흐름도이다.

도 2는 본 발명의 다른 실시예에 따른 ZNS기반 SSD의 컴팩션 방법을 설명하기 위한 흐름도이다.

도 3는 본 발명의 일 실시예에 따른 롱-리브드 SST(Long-lived SST)를 설명하기 위한 도면이다.

도 4는 본 발명의 일 실시예에 따른 숏-리브드 SST(Short-lived SST)를 설명하기 위한 도면이다.

도 5는 본 발명의 일 실시예에 따른 숏-리브드 SST의 컴팩션 방법을 설명하기 위한 도면이다.

도 6은 본 발명의 일 실시예에 따른 라이프타임-레벨링을 수행하는 컴팩션 장치를 설명하기 위한 블록도이다.

도 7 내지 12는 본 발명에 대한 실험 결과를 설명하기 위한 도면이다.

발명을 실시하기 위한 구체적인 내용

[0018] 본 발명은 다양한 변경을 가할 수 있고 여러 가지 실시예를 가질 수 있는 바, 특정 실시예들을 도면에 예시하고 상세한 설명에 상세하게 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 것으로 이해되어야 한다. 각 도면을 설명하면서 유사한 참조부호를 유사한 구성요소에 대해 사용하였다.

[0019] 어떤 구성요소가 다른 구성요소에 "연결되어" 있다거나 "접속되어" 있다고 언급된 때에는, 그 다른 구성요소에 직접적으로 연결되어 있거나 또는 접속되어 있을 수도 있지만, 중간에 다른 구성요소가 존재할 수도 있다고 이해되어야 할 것이다. 반면에, 어떤 구성요소가 다른 구성요소에 "직접 연결되어" 있다거나 "직접 접속되어" 있다고 언급된 때에는, 중간에 다른 구성요소가 존재하지 않는 것으로 이해되어야 할 것이다.

[0020] 본 출원에서 사용한 용어는 단지 특정한 실시예를 설명하기 위해 사용된 것으로, 본 발명을 한정하려는 의도가 아니다. 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다. 본 출원에서, "포함하다" 또는 "가지다" 등의 용어는 명세서상에 기재된 특징, 숫자, 단계, 동작, 구성요소, 부품 또는 이들을 조합한 것이 존재함을 지정하려는 것이지, 하나 또는 그 이상의 다른 특징들이나 숫자, 단계, 동작, 구성요소, 부품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.

[0021] 다르게 정의되지 않는 한, 기술적이거나 과학적인 용어를 포함해서 여기서 사용되는 모든 용어들은 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자에 의해 일반적으로 이해되는 것과 동일한 의미를 가지고 있다. 일반적으로 사용되는 사전에 정의되어 있는 것과 같은 용어들은 관련 기술의 문맥 상 가지는 의미와 일치하는 의미를 가지는 것으로 해석되어야 하며, 본 출원에서 명백하게 정의하지 않는 한, 이상적이거나 과도하게 형식적인 의미로 해석되지 않는다.

[0022] 본 발명에서 LSM-tree가 LevelDB임을 가정하여 컴팩션에 대하여 설명한다. LevelDB에서는 컴팩션의 대상이 되는 SST를 키 레인지(Key-range)에 따라 라운드로빈(round-robin) 방식으로 선택한다. 즉, 레벨별로 컴팩션이 시작

되는 Key인 컴팩션 포인터(Compaction Pointer, CP)를 두고, 그 CP보다 더 큰 최소키값(Smallest key)를 가진 CP와 가장 Key의 거리가 가까운 SST를 컴팩션 대상으로 선택할 수 있다. 한편, LevelDB에서는 레벨별로 존(Zone)을 분리하여 할당함으로써, 라운드로빈 방식을 유지할 때, 존 이용도(Zone utilization)를 극대화할 수 있다.

[0023] 하지만, 본 발명의 LSM-tree가 LevelDB로 한정되는 것은 아니며, 다양한 종류의 LSM-tree가 사용될 수 있음은 통상의 기술자에게 자명할 것이다.

[0024] 이하, 첨부된 도면을 참조하여 본 발명에 대해 설명한다.

[0025] 도 1은 본 발명의 일 실시예에 따른 ZNS기반 SSD의 컴팩션 방법을 설명하기 위한 흐름도이다.

[0026] 단계 S110에서는, ZNS기반 SSD가 LSM-tree에서 소정의 기준에 따라 컴팩션 대상이 되는 레벨인 대상레벨(L_i)을 결정한다.

[0027] 예컨대, ZNS기반 SSD는 LSM-tree에서 각 레벨별로 미리 할당된 임계치(threshold)과 현재 레벨의 크기의 비율이 소정의 기준치를 초과하는 경우에, 그 레벨을 대상레벨(L_i)로 결정할 수 있다.

[0028] 단계 S120에서는, ZNS기반 SSD가 그 대상레벨(L_i)에서 컴팩션 포인터(Compaction Pointer, CP_i)의 위치에 대응되는 SST인 대상SST(T^i_j)에 기반하여, 병합될 SST의 집합인 병합대상집합과 컴팩션에 포함되는 키 레인지를 나타내는 컴팩션 윈도우(Compaction Window)를 초기화한다.

[0029] 즉, ZNS기반 SSD는 현재의 컴팩션 포인터(CP_i)가 가리키고 있는 SST인 대상SST(T^i_j)를 선택하고, 그 대상SST(T^i_j)를 이용하여 병합대상집합과 컴팩션 윈도우를 초기화할 수 있다.

[0030] 보다 구체적으로, ZNS기반 SSD는 병합대상집합인 $M = \{T^i_j\}$로, 컴팩션 윈도우인 $W = [\kappa_s(T^i_j), \kappa_e(T^i_j)]$로 초기화할 수 있다. 이때, κ_s는 해당 SST의 최소키값(smallest key)을 나타내고, κ_e는 해당 SST의 최대키값(largest key)을 나타낸다.

[0031] 단계 S130에서는, ZNS기반 SSD가 그 컴팩션 포인터(CP_i)의 다음 컴팩션 포인터 위치, 컴팩션 윈도우 및 그 대상레벨(L_i)의 하위레벨(L_{i+1})에 포함된 복수의 SST(T^{i+1}_k)를 이용하여, 병합대상집합 및 컴팩션윈도우를 갱신한다.

[0032] 우선, ZNS기반 SSD는 복수의 SST(T^{i+1}_k) 중에서 $\kappa_s(T^{i+1}_k) \leq \kappa_e(W)$이고, $\kappa_s(W) \leq \kappa_e(T^{i+1}_k)$인 SST를 선택하여, 병합대상집합 M에 추가시킬 수 있다. 그리고, ZNS기반 SSD는 컴팩션 윈도우인 W를 아래 수학식 1과 같이 확장시킬 수 있다. 또한, ZNS기반 SSD는 그 과정을 반복하여 수행함으로써 병합대상집합 및 컴팩션윈도우를 갱신할 수 있다.

[0033] [수학식 1]

[0034] $$\kappa_s(W) = \kappa_s(T^{i+1}_k) \text{ if } \kappa_s(T^{i+1}_k) < \kappa_s(W)$$

[0035] $$\kappa_e(W) = \kappa_e(T^{i+1}_k) \text{ if } \kappa_e(W) < \kappa_e(T^{i+1}_k)$$

[0036] 또한, ZNS기반 SSD는 복수의 SST(T^{i+1}_k) 중에서 그 대상레벨(L_i)의 모든 SST와 겹치는 키 레인지가 존재하지 않으며, 그 컴팩션 포인터(CP_i)의 다음 컴팩션 포인터 위치보다 작은 경우에는, 해당 SST를 병합대상집합 M에 추가하고, 컴팩션 윈도우인 W를 아래 수학식 2와 같이 확장시킬 수 있다. 또한, ZNS기반 SSD는 그 과정을 반복하여 수행함으로써 병합대상집합 및 컴팩션윈도우를 갱신할 수 있다. 한편, 이에 대하여는 아래의 실시예에서 보다 구체적으로 설명한다.

[0037] [수학식 2]

[0038] $$\kappa_e(W) = \kappa_e(T^{i+1}_k)$$

[0039] 다른 실시예에서는, ZNS기반 SSD는 그 복수의 SST(T^{i+1}_k) 중에서 키 레인지의 전부 또는 일부가 컴팩션 윈도우와 중복되는 SST인 제1 SST 및 대상SST(T^i_j)의 최대키(largest key)값보다 최소키(smallest key)값이 크고, 그 다음 컴팩션 포인터 위치보다 최대키(largest key)값이 작은 SST인 제2 SST을 이용하여, 병합대상집합 및 컴팩션 윈도우를 갱신할 수 있다.

[0040] 이때, 제1 SST는 그 복수의 SST(T^{i+1}_k) 중에서 키 레인지의 전부가 현재의 컴팩션 윈도우에 포함되거나, 일부가 현재의 컴팩션 윈도우와 중복되는 모든 SST를 나타낸다.

[0041] 예컨대, 도 3을 참조하면, 대상SST가 SST1일 때, SST3과 SST4가 SST1으로 초기화된 컴팩션 윈도우와 키 레인지가 일부 중복되는 제1 SST일 수 있다.

[0042] 또한, 제2 SST는 그 복수의 SST(T^{i+1}_k) 중에서 대상SST(T^i_j)의 최대키(largest key)값보다 최소키(smallest key)값이 크고, 그 다음 컴팩션 포인터 위치보다 최대키(largest key)값이 작은 모든 SST를 나타낸다.

[0043] 예컨대, 도 3을 참조하면, 대상SST가 SST1일 때, SST5가 SST1의 최대키값보다 최소키값이 더 크고, 그 다음 컴팩션 포인터 위치인 SST2보다 최대키값이 더 작은 제2 SST일 수 있다.

[0044] 마지막으로 단계 S140에서는, ZNS기반 SSD가 그 병합대상집합을 이용하여 컴팩션을 수행한다.

[0045] 즉, ZNS기반 SSD는 그 병합대상집합에 포함된 모든 SST를 머지(Merge)하여 새로운 SST를 하위레벨(L_{i+1})에 작성할 수 있다. 그리고, ZNS기반 SSD는 그 병합대상집합에 포함된 모든 SST를 LSM-tree에서 삭제할 수 있다.

[0046] 또한, ZNS기반 SSD는 대상레벨(L_i)의 컴팩션 포인터(CP_i)의 위치를 그 컴팩션 포인터(CP_i)보다 큰 $\kappa_s(T^i_j)$를 갖는 SST 중 가장 작은 SST의 $\kappa_s(T^i_j)$로 변경할 수 있다. 즉, ZNS기반 SSD는 컴팩션 포인터(CP_i)의 위치를 그 컴팩션 포인터(CP_i)의 다음 컴팩션 포인터 위치로 변경할 수 있다.

[0047] 이때, 도 3을 참조하면, 종래의 기술에서 컴팩션이 발생할 때, 존(Zone)이 어떻게 변경되는지 나타나 있다.

[0048] 컴팩션 이전에 SST1과 SST2는 대상레벨(L_i)에 존재하며, Zone 9에 위치하고 있다. 또한, 하위레벨(L_{i+1})에 존재하는 SST들은 Zone 10에 위치하고 있다. 이때, ZNS기반 SSD는 대상레벨(L_i)과 하위레벨(L_{i+1})의 컴팩션을 수행할 때, 먼저 SST1을 대상레벨(L_i)에서 선택하고, SST1과 겹치는 키 레인지를 보유한 SST3, SST4를 하위레벨(L_{i+1})에서 선택할 수 있다. ZNS기반 SSD는 컴팩션을 통해 새롭게 합쳐진 SST들(SST8,9,10)을 하위레벨(L_{i+1})에 생성할 수 있다. 그리고, 이 데이터는 Zone 11에 순차적으로 작성될 수 있다. 한편, 하위레벨(L_{i+1})의 SST 중에서 SST3,4가 컴팩션 과정에서 삭제되었기 때문에, Zone 10의 SST3,4가 Invalid SST로 설정될 수 있다. 만약 다음 컴팩션이 대상레벨(L_i)에서 발생한다면, 마찬가지로 SST2,6,7이 컴팩션의 대상으로 선택되며, 새로운 SST인 SST 11,12,13은 Zone 11과 12에 순차적으로 작성될 수 있다. 그리고 하위레벨(L_{i+1})의 SST인 SST6,7은 Zone10에서 Invalid SST로 설정될 수 있다. 이때, SST5는 대상레벨(L_i)의 SST와 겹치는 키 레인지가 존재하지 않아 컴팩션에 참여하지 못하게 될 수 있다. 따라서, 비록 Zone 10이 Invalid space가 많더라도, SST5이 valid SST이기 때문에 존 리셋(zone reset)이 불가능하다. 따라서, SST5가 삭제되기 위해서는 대상레벨(L_i)의 CP가 SST5의 키 레인지를 포함해야 하며 그때서야 비로서 SST5는 컴팩션에 참여하여 Zone 10의 리셋이 가능해진다. 이와 같은 롱-리브드(Long-lived) SST의 경우 공간 증폭(Space Amplification)을 증가시키는 원인이 된다.

[0049] 여기서, 본 발명의 일 실시예에 따른 ZNS기반 SSD의 컴팩션 방법이 적용되면 이와 같은 문제를 방지할 수 있다.

[0050] 보다 구체적으로, 도 3의 SST5의 경우에 현재의 컴팩션 포인터(CP_i)가 위치한 SST1과 다음 컴팩션 포인터가 위치할 SST2의 사이에 위치하고 있기 때문에, 앞서 설명한 제2 SST에 해당하게 되므로, SST5는 병합대상집합에 포함되어 컴팩션에 참여할 수 있게 된다.

[0051] 도 2는 본 발명의 다른 실시예에 따른 ZNS기반 SSD의 컴팩션 방법을 설명하기 위한 흐름도이다.

[0052] 단계 S210에서는, ZNS기반 SSD가 LSM-tree에서 소정의 기준에 따라 컴팩션 대상이 되는 레벨인 대상레벨(L_i)을

결정한다.

[0053] 단계 S220에서는, ZNS기반 SSD가 대상레벨(L_i)에서 컴팩션 포인터(CP_i)의 위치에 대응되는 SST인 대상SST(T^i_j)에 기반하여, 병합될 SST의 집합인 병합대상집합과 컴팩션에 포함되는 키 레인지를 나타내는 컴팩션 윈도우를 초기화한다.

[0054] 단계 S230에서는, ZNS기반 SSD가 그 컴팩션 윈도우 및 대상레벨(L_i)의 하위레벨(L_{i+1})에 포함된 복수의 SST(T^{i+1}_k)를 이용하여, 병합대상집합 및 컴팩션윈도우를 갱신한다.

[0055] 단계 S240에서는, ZNS기반 SSD가 그 병합대상집합을 이용하여 컴팩션을 수행한다.

[0056] 이때, ZNS기반 SSD는 컴팩션 과정에서 생성되는 SST가 컴팩션 포인터(CP_i)의 다음 컴팩션 포인터 위치를 포함하면, 그 SST를 그 다음 컴팩션 포인터 위치를 기준으로 2개의 SST로 분리한다.

[0057] 이때, 도 4를 참조하면, 종래의 기술에 대한 설명이 나타나 있다.

[0058] 우선, ZNS기반 SSD가 대상레벨(L_i)과 하위레벨(L_{i+1})의 컴팩션을 수행할 때, 대상SST가 SST1일 수 있다. 그리고, ZNS기반 SSD는 SST1과 키 레인지가 겹치는 SST3,4,5를 선택하고, SST1,3,4,5로부터 새로운 SST인 SST8,9,10,11을 생성할 수 있다. 또한, SSD 컨트롤러는 이를 Zone 11에 작성하고, SST3,4,5를 Zone 10에서 삭제할 수 있다.

[0059] 여기서, ZNS기반 SSD는 연속적으로 SST2에 대하여 컴팩션을 수행하는 경우를 가정할 수 있다. 이때, ZNS기반 SSD는 SST2,11,6,7로부터 새로운 SST인 SST12,13,14를 생성하고, Zone 11과 Zone 12에 순차적으로 작성할 수 있다.

[0060] 이때, SST11의 경우, SST1에 대한 첫번째 컴팩션에 의해 생성된 즉시, SST2에 대한 두번째 컴팩션에서 삭제될 수 있다. 이는, SST11의 키 레인지가 첫번째 컴팩션과 두번째 컴팩션에서 모두 겹쳐 있기 때문이다. 즉, SST11의 라이프타임은 하위레벨(L_{i+1})에 존재하는 다른 SST들에 비해 매우 짧으며, 이런 짧은 수명을 지닌 SST는 Zone 11처럼 존 내부에 홀(hole)을 생성하게 된다. 이처럼 홀이 생성된 존의 경우 다음 컴팩션까지 리셋(reset)이 불가능하며, 이로 인하여 공간 증폭(Space Amplification)이 증가할 수 있다.

[0061] 여기서, 본 발명의 다른 실시예에 따른 ZNS기반 SSD의 컴팩션 방법이 적용되면 이와 같은 문제를 방지할 수 있다.

[0062] 보다 구체적으로, 도 5를 참조하면, ZNS기반 SSD가 SST1에 대하여 컴팩션을 수행할 때, 하위레벨(L_{i+1})의 Zone에는 4개의 새로운 SST가 생성되며, 마지막 SST인 SST10,11의 경우 그 다음 컴팩션 포인터 위치에서 분리(Split)할 수 있다. 이때, SST10이 SST 크기 임계치(threshold)보다 작더라도, SST는 그 다음 컴팩션 포인터 위치에서 분할되며, 새로운 SST는 그 다음 컴팩션 포인터 위치에서부터 생성된다. 만약 해당하는 분할이 이루어지지 않을 경우 SST10은 SST11과 합쳐진 큰 SST가 되어 다음 컴팩션에 참여할 수 있다.

[0063] 이처럼, ZNS기반 SSD가 SST10과 SST11을 그 다음 컴팩션 포인터 위치를 기준으로 분리함으로 인하여, 존에 홀이 생성되는 문제를 방지할 수 있다.

[0064] 다른 실시예에서는, ZNS기반 SSD가 그 분리된 2개의 SST중에서, 그 다음 컴팩션 포인터 위치의 이후에 위치하는 SST를 임시로 SST를 저장하기 위하여 설정된 T-zone(Temporary zone)에 분리하여 저장할 수 있다.

[0065] 즉, 도 4를 참조하면, ZNS기반 SSD는 SST11이 숏-리브드(short-lived) SST이므로, Zone 10이 아닌 T-Zone이라는 별도의 Zone에 작성할 수 있다. 이때, T-Zone에 작성된 SST의 경우 항상 숏-리브드 SST이므로, 쉽게 리클레임(reclaim)될 수 있다.

[0066] 또 다른 실시예에서는, ZNS기반 SSD가 컴팩션 과정에서 생성되는 SST가 하위레벨(L_{i+1})의 컴팩션 포인터(CP_{i+1})의 위치를 포함하면, 그 SST를 그 하위레벨(L_{i+1})의 컴팩션 포인터(CP_{i+1}) 위치를 기준으로 2개의 SST로 분리할 수 있다.

[0067] 한편, ZNS기반 SSD가 하위레벨(L_{i+1})에 SST를 작성하는 경우에, SST는 하위레벨(L_{i+1})의 컴팩션 포인터(CP_{i+1})의 위치를 포함할 수 없다. 이는, ZNS기반 SSD가 SST의 중간지점에서부터 컴팩션을 수행할 수 없기 때문에, 컴팩션 포인터(CP_{i+1})의 위치가 보장되지 않는다면 하위레벨(L_{i+1})의 컴팩션 포인터(CP_{i+1})는 하위레벨(L_{i+1})의 컴팩션 포

인터(CP_{i+1})보다 큰 다음 SST의 최소키값이 되어, 기존의 컴팩션 포인터(CP_{i+1})부터 컴팩션을 시작하지 못한다. 즉, 하위레벨(L_{i+1})의 컴팩션 포인터(CP_{i+1})가 대상레벨(L_i)과 하위레벨(L_{i+1})의 컴팩션에 의해 변경된다면, 대상레벨(L_i)과 하위레벨(L_{i+1})의 컴팩션 시작 위치가 변경되어, 롱-리브드 SST를 다수 생성시킬 수 있다. 이는 존 이용도(Zone utilization) 저하에 치명적이므로 컴팩션 과정에서 하위레벨(L_{i+1})의 컴팩션 포인터(CP_{i+1})를 필수적으로 보장해야 한다.

[0068] 예컨대, 도 5를 참조하면, SST2로 인해 발생한 컴팩션은 해당 실시예를 구체적으로 보여준다. 즉, ZNS기반 SSD는 SST2,11,5를 컴팩션에 포함하고, 새로운 SST를 생성한다. 이때 ZNS기반 SSD는 하위레벨(L_{i+1})의 컴팩션 포인터(CP_{i+1})인 SST13의 최소키값의 위치에서 SST 12를 분할(Split)하여 작은 크기의 SST를 생성할 수 있다. SST13의 경우 하위레벨(L_{i+1})의 컴팩션 포인터(CP_{i+1})부터 시작되기 때문에 하위레벨(L_{i+1})의 컴팩션 포인터(CP_{i+1})는 대상레벨(L_i)과 하위레벨(L_{i+1})의 컴팩션 과정에서 손상되지 않는다.

[0069] 도 6은 본 발명의 일 실시예에 따른 라이프타임-레벨링을 수행하는 컴팩션 장치를 설명하기 위한 블록도이다.

[0070] 도 6을 참조하면, 본 발명의 일 실시예에 따른 라이프타임-레벨링을 수행하는 컴팩션 장치(600)는 대상결정부(610), 컴팩션연산부(620), 컴팩션수행부(630)를 포함한다.

[0071] 이때, 본 발명의 일 실시예에 따른 컴팩션 장치(600)는 ZNS기반 SSD에 탑재될 수 있으며, LevelDB 등과 같은 LSM-tree의 컴팩션을 수행할 수 있다.

[0072] 대상결정부(610)는 LSM-tree에서 소정의 기준에 따라 컴팩션 대상이 되는 레벨인 대상레벨(L_i)을 결정한다.

[0073] 컴팩션연산부(620)는 대상레벨(L_i)에서 컴팩션 포인터(CP_i)의 위치에 대응되는 SST인 대상SST(T^i_j)에 기반하여, 병합될 SST의 집합인 병합대상집합과 컴팩션에 포함되는 키 레인지를 나타내는 컴팩션 윈도우를 초기화하고, 그 컴팩션 포인터(CP_i)의 다음 컴팩션 포인터 위치, 컴팩션 윈도우 및 대상레벨(L_i)의 하위레벨(L_{i+1})에 포함된 복수의 SST(T^{i+1}_k)를 이용하여, 그 병합대상집합 및 컴팩션윈도우를 갱신한다.

[0074] 마지막으로 컴팩션수행부(630)는 병합대상집합을 이용하여 컴팩션을 수행한다.

[0075] 다른 실시예에서는, 컴팩션연산부(620)는 복수의 SST(T^{i+1}_k) 중에서 키 레인지의 전부 또는 일부가 상기 컴팩션 윈도우와 중복되는 SST인 제1 SST 및 대상SST(T^i_j)의 최대키값보다 최소키값이 크고, 그 다음 컴팩션 포인터 위치보다 최대키값이 작은 SST인 제2 SST을 이용하여, 그 병합대상집합 및 컴팩션윈도우를 갱신할 수 있다.

[0076] 또 다른 실시예에서는, 컴팩션연산부(620)가 컴팩션 윈도우 및 대상레벨(L_i)의 하위레벨(L_{i+1})에 포함된 복수의 SST(T^{i+1}_k)만을 이용하여, 그 병합대상집합 및 컴팩션윈도우를 갱신할 때, 컴팩션수행부(630)는 컴팩션 과정에서 생성되는 SST가 컴팩션 포인터(CP_i)의 다음 컴팩션 포인터 위치를 포함하면, 그 SST를 다음 컴팩션 포인터 위치를 기준으로 2개의 SST로 분리할 수 있다.

[0077] 또 다른 실시예에서는, 컴팩션수행부(630)는 그 분리된 2개의 SST중에서, 다음 컴팩션 포인터 위치의 이후에 위치하는 SST를 임시로 SST를 저장하기 위하여 설정된 T-zone(Temporary zone)에 분리하여 저장할 수 있다.

[0078] 또 다른 실시예에서는, 컴팩션수행부(630)는 컴팩션 과정에서 생성되는 SST가 하위레벨(L_{i+1})의 컴팩션 포인터(CP_{i+1})의 위치를 포함하면, 그 SST를 하위레벨(L_{i+1})의 컴팩션 포인터(CP_{i+1}) 위치를 기준으로 2개의 SST로 분리할 수 있다.

[0079] 도 7 내지 12는 본 발명의 효과를 설명하기 위한 도면이다.

[0080] 여기서, LL(LL-Compaction)은 본 발명의 Lifetime-Leveling이 적용된 경우이다. 또한, BL(BaseLine)은 기존 LevelDB에서 Zone size를 무제한으로 설정한 경우이다. 또한, GC는 SSD size를 29GB로 설정하며 Garbage collection을 수행한 경우이다. 또한, LS는 SSD size를 29GB로 설정하며 Level별로 Zone의 분리하고 GC를 수행한 경우이다. 또한, Gear는 SSD size를 32GB로 설정한 GearDB를 수행한 경우이다.

[0081] 한편, GearDB의 경우 29GB의 SSD를 사용하는 경우 SSD공간이 모자라서 수행되지 못하였으며, write 성능 등의 비교를 위해 더 큰 SSD 공간을 할당하였다. Garbage collection의 경우 ZNS의 Zone이 1개가 남은 경우 triggering 되며, Greedy policy를 사용하여 구현하였다.

[0082] 도 7은 Fill random workload에서 전체적인 성능을 나타낸다. 전반적인 성능은 BL과 LL이 비슷한 성능을 보였고 GC가 일어나는 LS와 GC의 경우 추가적인 write amplification으로 인해 성능이 크게 감소하였다. GearDB의 경우 물리적인 Seektime이 존재하지 않는 ZNS에서 성능이 크게 감소하였다. 심지어 LS 및 GC에 비해 더 큰 공간을 할당했는데도, Gear Compaction의 비효율성으로 인해 성능이 감소하였다. 이는 LL-Compaction이 GearDB에 비해 효과적으로 Compaction을 최적화한 것을 의미한다.

[0083] 도 8은 시간에 따른 OPS의 변화를 측정한 그래프이다. BL 및 LL-Compaction의 경우 처음 시작할 때 Compaction이 시작하고 Level이 증가함에 따라 성능이 감소한다, 전체 level의 개수가 안정화 되가면서, 성능이 일정수준으로 유지된다. GC의 경우 Level별 분리를 하지 않아 GC가 triggering이 되는 시점이 매우 빠르게 일어난다. 40분부터 GC가 발생하며 이때부터 Baseline에 비해 성능이 크게 감소한다. LS의 경우 Level별 분리를 수행하였기 때문에 GC가 발생하는 시간이 늦어진다. 약 80분부터 GC가 triggering되며, Level별 분리를 수행하여 대부분의 Zone utilization이 높기 때문에 GC가 시작되면 성능이 급격하게 감소한다. GearDB의 경우 논문에서 언급된 것과 동일하게 compaction의 volume이 매우 커져 Performance가 안정적이지 못하다. 또한 GearCompaction의 자체 overhead로 인해 성능이 크게 감소한다.

[0084] 도 9는 각 Zone별로 Valid한 SST의 비율을 오름차순으로 sorting해둔 그래프이다. 다른 기법들의 경우 GC/LL-Compaction/Gear-Compaction 기법을 통해 전반적인 Space Amplification이 심해지지 않도록 관리하나, BaseLine의 경우 Zone이 무한대로 있다고 가정해, SA를 위한 추가적인 기법을 수행하지 않는다. 따라서 BaseLine의 경우 다른 기법들에 비해 50%더 많은 Zone을 활용하였으며, 이는 BaseLine의 Space Amplification이 크게 증가했음을 의미한다. GC/LS의 경우Free Zone이 하나 남았을 경우 Garbage collection의 triggering 되므로 SSD의 모든 Zone을 사용하였다. LL-Compaction의 경우 GC/LS GearDB의 경우보다 5%더 적은 Zone을 활용하였으며, 이는 LL-Compaction이 GC 없이 효율적으로 Space를 관리하는 것을 의미한다. 반면 GearDB의 경우 GC/LS에 비해 10%더 많은 Zone을 사용하였다. 이는 GearDB가 Space 관리를 LL-Compaction에 비해 비효율적으로 하고 있음을 의미한다.

[0085] 도 10은 모든 기법들에 대해 write의 총량을 L0 write/Compaction/GC로 breakdown한 결과에 대한 그래프이다. LL-Compaction은 BL에 비해 Compaction으로 인한 Write가 소폭 감소하였는데, 이는 Short-Lived File의 Split으로 인해, rewrite 되는 SST의 총량을 줄여 Compaction의 효율이 좋아졌기 때문이다. GC/LS와 같은 기법들은 GC로 인해 write의 총량이 크게 증가하였으며, GearDB의 경우 Gear Compaction으로 인해 write가 크게 증가하였다. 이는 GearDB의 Gear Compaction이 write amplification을 GC보다 크게 증가시킴을 의미한다.

[0086] 다음으로 다양한 workload에서의 실험을 위해 YCSB workload를 통해 LL-Compaction을 분석하였다. YCSB는 1억 3500만회의 insert를 진행하였으며, 이는 총 78GB의 공간을 사용한다. SSD크기는 80GB로 설정하였으며, GearDB의 경우 더 많은 공간을 필요로 하기 때문에 본 실험에서는 제외하였다. Workload는 load/run 모두 zipfian을 활용하였다.

[0087] 도 11은 YCSB workload를 수행하였을 때, GC 성능 대비 얼마나 성능이 상승했는지를 보여주는 그래프이다. GC/LS에 비해 LL-Compaction이 GC가 존재하지 않아 전반적인 성능이 더 좋게 나왔다. 이는 Uniform workload 이외에도 성능이 증가한다는 것, Read/Write가 섞여있는 workload에서도 LL-Compaction의 성능이 크게 상승한다는 것을 의미한다.

[0088] 도 12는 YCSB load workload에서 시간당 성능의 변화를 관찰한 그래프이다. Uniform workload와는 다르게 Key의 Zipfian 분포로 인해 LL/GC/LS 모두 초반부터 OPS가 크게 흔들리는 것을 확인할 수 있다. 하지만 GC로 인한 손해는 LS/GC 모두 존재해 전체적인 성능은 LL-Compaction이 타 기법에 비해 유리함을 알 수 있다.

[0089] 해당하는 실험 결과들을 통해 알 수 있듯이, LL-Compaction의 경우 ZNS를 사용할 때 Zone size가 SST size보다 큰 경우에 효과적으로 DB의 Space Amplification을 줄일 수 있다.

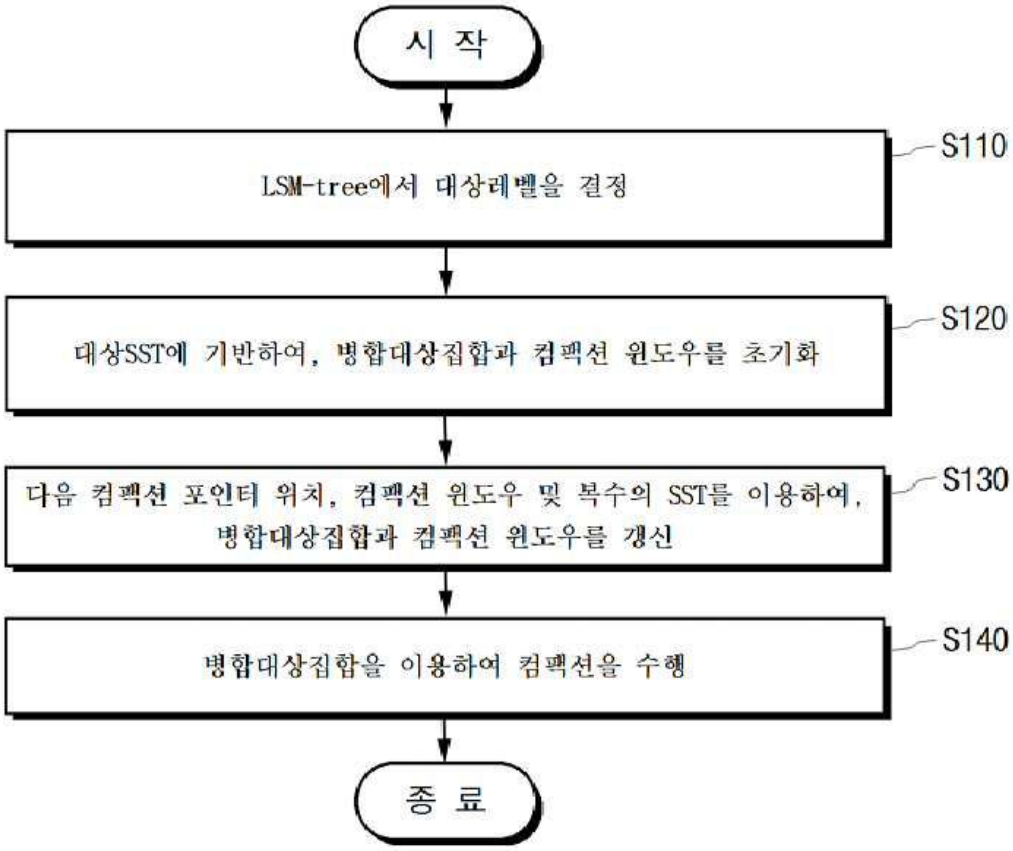
[0090] 이상의 설명은 본 발명의 기술 사상을 예시적으로 설명한 것에 불과한 것으로, 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 사람이라면 본 발명의 본질적인 특성에서 벗어나지 않는 범위에서 다양한 수정 및 변형이 가능할 것이다.

[0091] 따라서, 본 발명에 실행된 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 동등한 범위 내에 있는 모든 기술 사상은 본 발명의 권리범위에 포함되는 것으로 해석되어야 할 것이다.

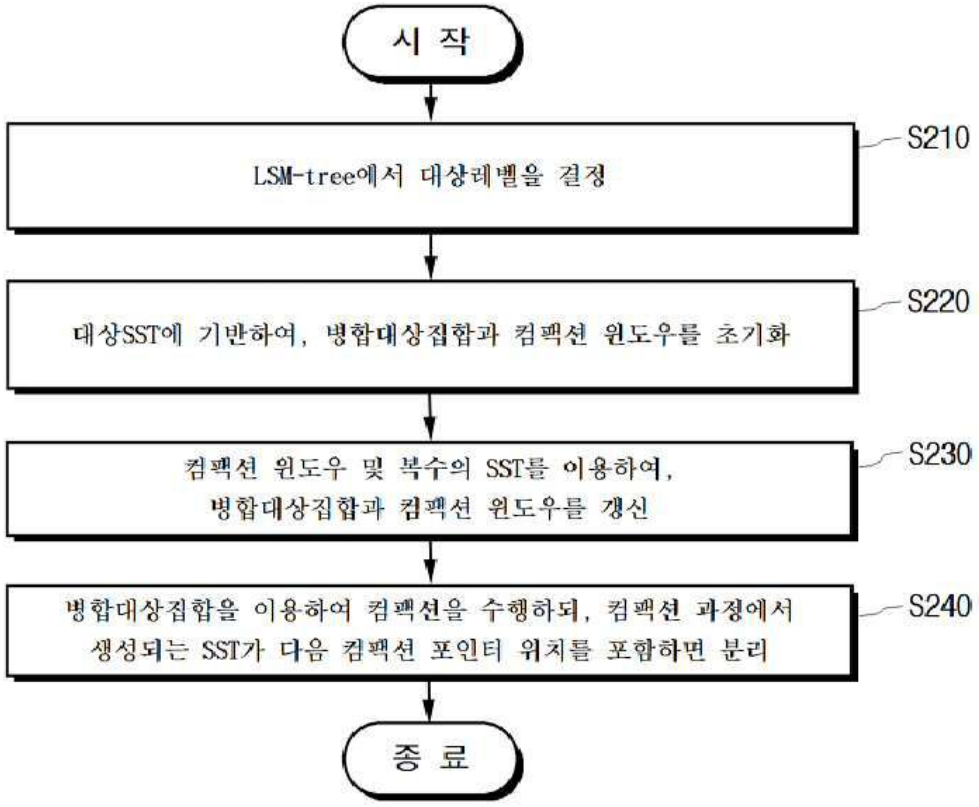
부호의 설명

도면

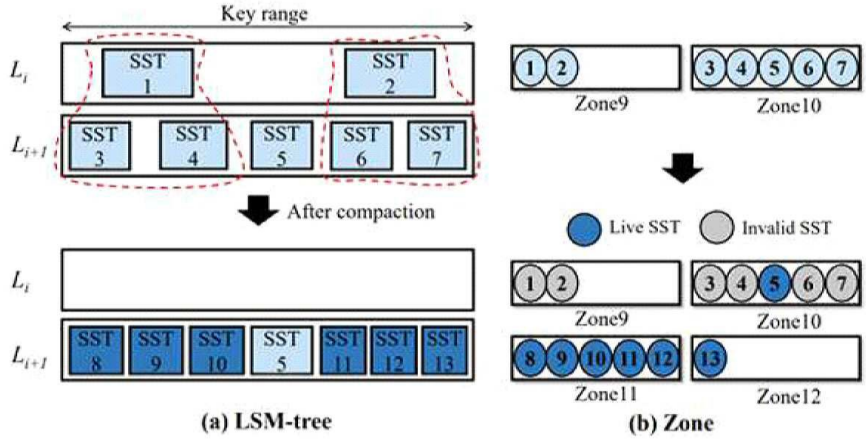
도면1



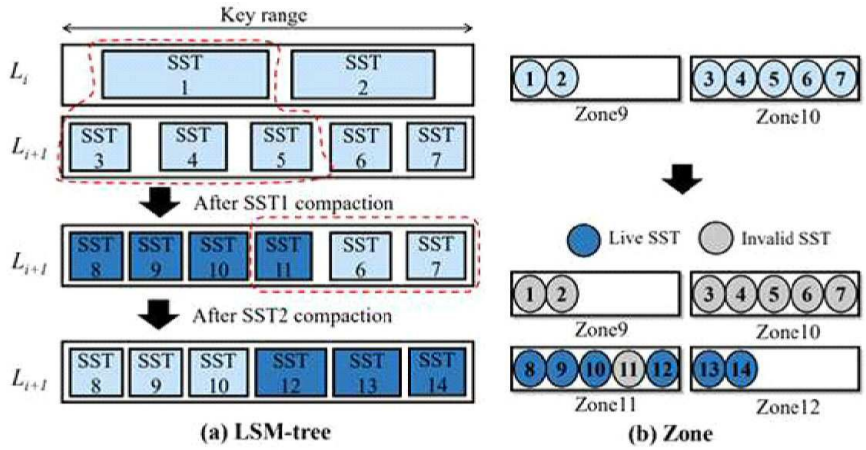
도면2


시 작
LSM-tree에서 대상레벨을 결정
S210
대상SST에 기반하여, 병합대상집합과 컴팩션 윈도우를 초기화
S220
컴팩션 윈도우 및 복수의 SST를 이용하여,
병합대상집합과 컴팩션 윈도우를 갱신
S230
병합대상집합을 이용하여 컴팩션을 수행하되, 컴팩션 과정에서
생성되는 SST가 다음 컴팩션 포인터 위치를 포함하면 분리
S240
종 료

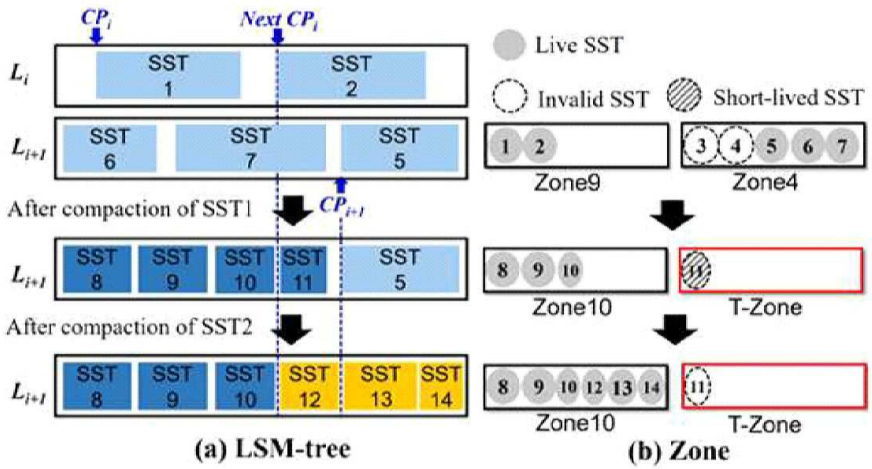
도면3


Key range
L_i
L_{i+1}
SST 1
SST 2
SST 3
SST 4
SST 5
SST 6
SST 7
After compaction
SST 8
SST 9
SST 10
SST 11
SST 12
SST 13
(a) LSM-tree
Zone9
Zone10
Zone11
Zone12
Live SST
Invalid SST
(b) Zone

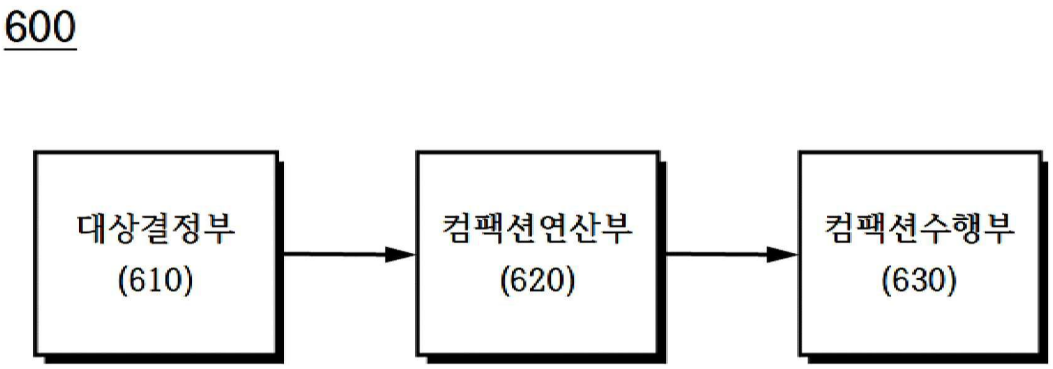
도면4


Key range
L_i
L_{i+1}
SST 1
SST 2
SST 3
SST 4
SST 5
SST 6
SST 7
After SST1 compaction
SST 8
SST 9
SST 10
SST 11
After SST2 compaction
SST 12
SST 13
SST 14
(a) LSM-tree
Zone9
Zone10
Live SST
Invalid SST
Zone11
Zone12
(b) Zone

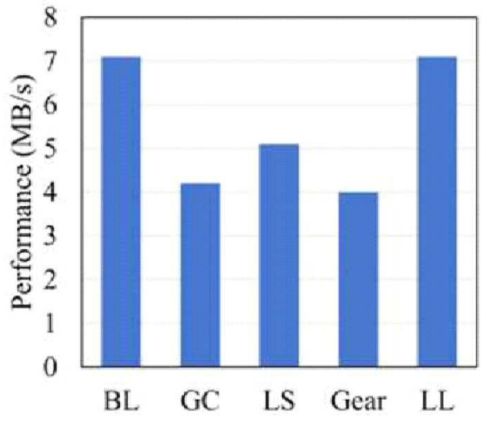
도면5


CP_i
$Next\ CP_i$
L_i
L_{i+1}
SST 1
SST 2
SST 6
SST 7
SST 5
After compaction of SST1
CP_{i+1}
SST 8
SST 9
SST 10
SST 11
After compaction of SST2
SST 12
SST 13
SST 14
(a) LSM-tree
Live SST
Invalid SST
Short-lived SST
Zone9
Zone4
Zone10
T-Zone
(b) Zone

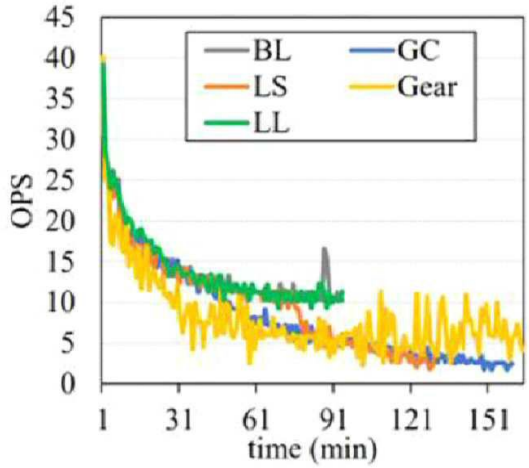
도면6


600
대상결정부
(610)
컴팩션연산부
(620)
컴팩션수행부
(630)

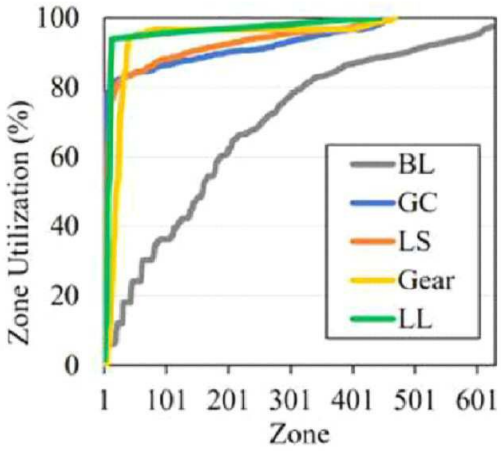
도면7


Performance (MB/s)
0
1
2
3
4
5
6
7
8
BL
GC
LS
Gear
LL

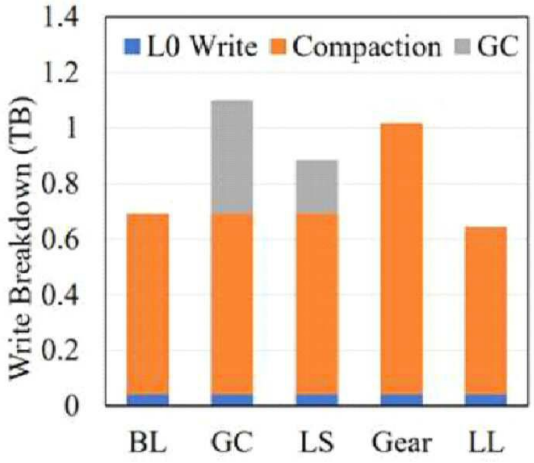
도면8


OPS
0
5
10
15
20
25
30
35
40
45
1
31
61
91
121
151
time (min)
BL
GC
LS
Gear
LL

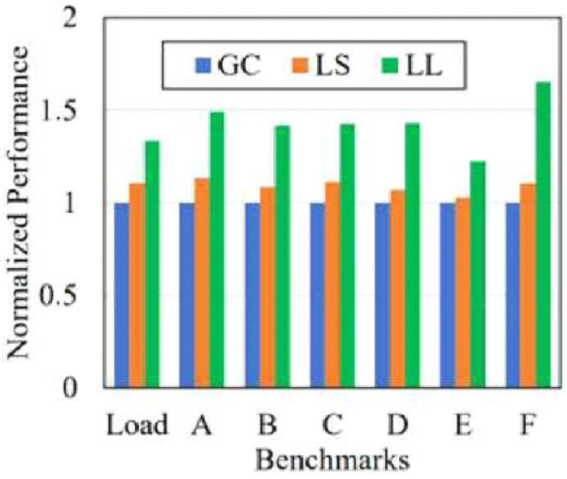
도면9


Zone Utilization (%)
0
20
40
60
80
100
1
101
201
301
401
501
601
Zone
BL
GC
LS
Gear
LL

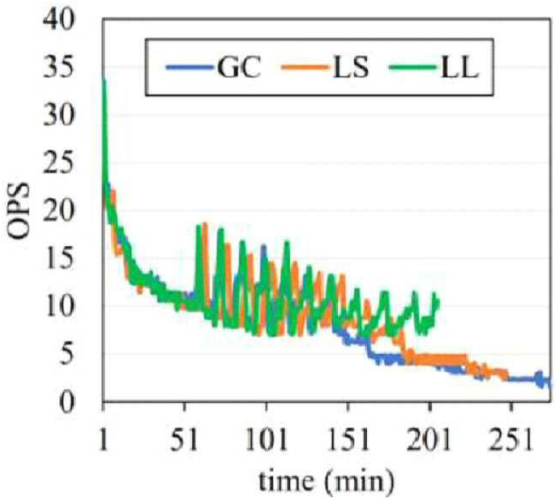
도면10



도면11



도면12



【심사관 직권보정사항】

【직권보정 1】

【보정항목】 청구범위

【보정세부항목】 청구항 7

【변경전】

제6항에 있어서,

상기 컴팩션연산부는

상기 복수의 SST(T^{i+1}_k) 중에서 키 레인지의 전부 또는 일부가 상기 컴팩션 윈도우와 중복되는 SST인 제1 SST 및 상기 대상SST(T^i_j)의 최대키값보다 최소키값이 크고, 상기 다음 컴팩션 포인터 위치보다 최대키값이 작은 SST인 제2 SST을 이용하여, 상기 병합대상집합 및 상기 컴팩션윈도우를 갱신하는 것을 특징으로 하는 라이프타임-레벨링을 수행하는컴팩션 장치.

【변경후】

제6항에 있어서,

상기 컴팩션연산부는

상기 복수의 SST(T^{i+1}_k) 중에서 키 레인지의 전부 또는 일부가 상기 컴팩션 윈도우와 중복되는 SST인 제1 SST 및 상기 대상SST(T^i_j)의 최대키값보다 최소키값이 크고, 상기 다음 컴팩션 포인터 위치보다 최대키값이 작은 SST인 제2 SST을 이용하여, 상기 병합대상집합 및 상기 컴팩션윈도우를 갱신하는 것을 특징으로 하는 라이프타임-레벨링을 수행하는 컴팩션 장치.

【직권보정 2】

【보정항목】 청구범위

【보정세부항목】 청구항 8

【변경전】

LSM-tree에서 소정의 기준에 따라 컴팩션 대상이 되는 레벨인 대상레벨(L_i)을 결정하는 대상결정부;

상기 대상레벨(L_i)에서 컴팩션 포인터(CP_i)의 위치에 대응되는 SST인 대상SST(T^i_j)에 기반하여, 병합될 SST의 집합인 병합대상집합과 컴팩션에 포함되는 키 레인지를 나타내는 컴팩션 윈도우를 초기화하고,

상기 컴팩션 윈도우 및 상기 대상레벨(L_i)의 하위레벨(L_{i+1})에 포함된 복수의 SST(T^{i+1}_k)를 이용하여, 상기 병합대상집합 및 상기 컴팩션윈도우를 갱신하는 컴팩션연산부; 및

상기 병합대상집합을 이용하여 컴팩션을 수행하는 컴팩션수행부;

를 포함하고,

상기 컴팩션수행부는

컴팩션 과정에서 생성되는 SST가 상기 컴팩션 포인터(CP_i)의 다음 컴팩션 포인터 위치를 포함하면, 상기 SST를 상기 다음 컴팩션 포인터 위치를 기준으로 2개의 SST로 분리하는 것을 특징으로 하는 라이프타임-레벨링을 수행하는컴팩션 장치.

【변경후】

LSM-tree에서 소정의 기준에 따라 컴팩션 대상이 되는 레벨인 대상레벨(L_i)을 결정하는 대상결정부;

상기 대상레벨(L_i)에서 컴팩션 포인터(CP_i)의 위치에 대응되는 SST인 대상SST(T^i_j)에 기반하여, 병합될 SST의 집합인 병합대상집합과 컴팩션에 포함되는 키 레인지를 나타내는 컴팩션 윈도우를 초기화하고,

상기 컴팩션 윈도우 및 상기 대상레벨(L_i)의 하위레벨(L_{i+1})에 포함된 복수의 SST(T^{i+1}_k)를 이용하여, 상기 병합대상집합 및 상기 컴팩션윈도우를 갱신하는 컴팩션연산부; 및

상기 병합대상집합을 이용하여 컴팩션을 수행하는 컴팩션수행부;

를 포함하고,

상기 컴팩션수행부는

컴팩션 과정에서 생성되는 SST가 상기 컴팩션 포인터(CP_i)의 다음 컴팩션 포인터 위치를 포함하면, 상기 SST를 상기 다음 컴팩션 포인터 위치를 기준으로 2개의 SST로 분리하는 것을 특징으로 하는 라이프타임-레벨링을 수행하는 컴팩션 장치.

【직권보정 3】

【보정항목】 청구범위

【보정세부항목】 청구항 9

【변경전】

제8항에 있어서,

상기 컴팩션수행부는

컴팩션 과정에서 생성되는 SST가 상기 하위레벨(L_{i+1})의 컴팩션 포인터(CP_{i+1})의 위치를 포함하면, 상기 SST를 상기 하위레벨(L_{i+1})의 컴팩션 포인터(CP_{i+1}) 위치를 기준으로 2개의 SST로 분리하는 것을 특징으로 하는 라이프타임-레벨링을 수행하는컴팩션 장치.

【변경후】

제8항에 있어서,

상기 컴팩션수행부는

컴팩션 과정에서 생성되는 SST가 상기 하위레벨(L_{i+1})의 컴팩션 포인터(CP_{i+1})의 위치를 포함하면, 상기 SST를 상기 하위레벨(L_{i+1})의 컴팩션 포인터(CP_{i+1}) 위치를 기준으로 2개의 SST로 분리하는 것을 특징으로 하는 라이프타임-레벨링을 수행하는 컴팩션 장치.