(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2025-0093808
(43) 공개일자 2025년06월25일

(51) 국제특허분류(Int. Cl.)
G06F 16/21 (2019.01) *G06F 16/18* (2019.01)
G06F 16/22 (2019.01) *G06F 16/23* (2019.01)
(52) CPC특허분류
G06F 16/219 (2019.01)
G06F 16/1873 (2019.01)
(21) 출원번호 10-2023-0184040
(22) 출원일자 2023년12월18일
심사청구일자 2023년12월18일

(71) 출원인
(주) 글루시스
경기도 안양시 동안구 시민대로327번길 11-31 , 5층(관양동, 파낙스알앤디센터)
서강대학교산학협력단
서울특별시 마포구 백범로 35 (신수동, 서강대학교)
(72) 발명자
박정환
서울특별시 강동구 양재대로 1716, 1동 406호 (고덕동, 대우아파트)
박여현
서울특별시 서초구 매헌로 64, 5층(양재동, 엠제이)
(뒷면에 계속)
(74) 대리인
수안특허법인, 민영준

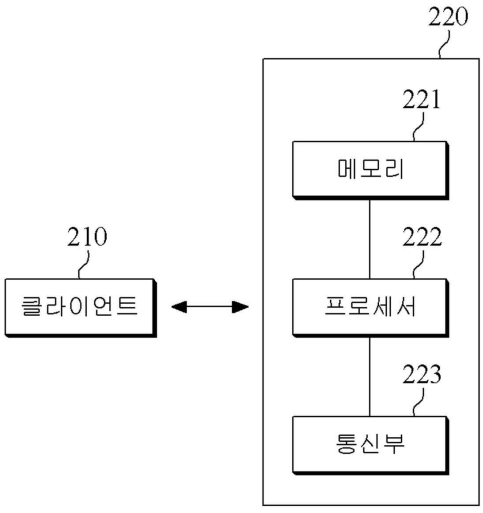
전체 청구항 수 : 총 11 항

(54) 발명의 명칭 **스토리지 서버 및 스토리지 서버의 데이터 입출력 방법**

(57) 요 약

버저닝 기법을 이용하여, 데이터 손실로 인한 피해를 줄일 수 있는 스토리지 서버 및 스토리지 서버의 데이터 입출력 방법이 개시된다. 개시된 스토리지 서버의 데이터 입출력 방법은 계산 노드로부터, 제1키-값에 대한 쓰기 요청을 수신하는 단계; 상기 제1키-값의 키에 대한 해시값을 이용하여, 상기 제1키-값에 대한 해시 테이블 및 버킷을 결정하는 단계; 상기 제1키-값의 버전에 따라서, 상기 버킷에 할당된 연결 리스트에서, 상기 제1키-값이 할당되는 제1노드의 위치를 결정하는 단계; 및 상기 제1노드에, 상기 제1키-값의 키 및 상기 제1노드와 제2노드의 연결 관계를 나타내는 포인터를 저장하는 단계를 포함한다.

대 표 도 - 도2



(52) CPC특허분류
G06F 16/2228 (2019.01)
G06F 16/2329 (2019.01)

(72) 발명자
김영재
서울특별시 서초구 잠원로12길 4, 101동 1808호(잠원동, 잠원현대아파트)
김경표
광주광역시 동구 용산3길 17(용산동, 모아엘가 에듀파크)
박성순
경기도 군포시 수리산로 244, 995동 901호(산본동, 한양백두아파트)

이 발명을 지원한 국가연구개발사업

과제고유번호	1711193184
과제번호	2020-0-00104-004
부처명	과학기술정보통신부
과제관리(전문)기관명	정보통신기획평가원
연구사업명	차세대엣지컴퓨팅시스템기술개발(R&D)
연구과제명	저지연 입출력 집약적 엣지 데이터 처리를 위한 스토리지 모듈 기술 개발
기 여 율	1/1
과제수행기관명	(주)글루시스
연구기간	2023.01.01 ~ 2023.12.31

명 세 서

청구범위

청구항 1

계산 노드로부터, 제1키-값에 대한 쓰기 요청을 수신하는 단계;

상기 제1키-값의 키에 대한 해시값을 이용하여, 상기 제1키-값에 대한 해시 테이블 및 버킷을 결정하는 단계;

상기 제1키-값의 버전에 따라서, 상기 버킷에 할당된 연결 리스트에서, 상기 제1키-값이 할당되는 제1노드의 위치를 결정하는 단계; 및

상기 제1노드에, 상기 제1키-값의 키 및 상기 제1노드와 제2노드의 연결 관계를 나타내는 포인터를 저장하는 단계

를 포함하는 스토리지 서버의 데이터 입출력 방법.

청구항 2

제 1항에 있어서,

제1노드의 위치를 결정하는 단계는

상기 연결 리스트의 노드에서, 상기 제1키-값의 키와 동일한 키가 할당된 제2키-값에 대한 노드를 검색하는 단계; 및

상기 검색된 노드의 위치보다 상위 노드의 위치로, 상기 제1노드의 위치를 결정하는 단계

를 포함하는 스토리지 서버의 데이터 입출력 방법.

청구항 3

제 2항에 있어서,

상기 제2노드는

상기 검색된 노드 중, 최상위 노드인

스토리지 서버의 데이터 입출력 방법.

청구항 4

제 3항에 있어서,

상기 제2노드를 나타내는 포인터를 저장하는 단계는

상기 제1키-값의 키와 상이한 키가 저장된 제3노드와, 상기 제2노드가 연결된 상태에서, 상기 제1키-값에 대한 쓰기 요청이 수신된 경우, 상기 제1노드와 상기 제3노드의 연결 관계를 나타내는 포인터를 추가로 상기 제1노드에 저장하는

스토리지 서버의 데이터 입출력 방법.

청구항 5

제 4항에 있어서,

상기 제1노드의 위치를 결정하는 단계는

상기 제1키-값의 키와 동일한 키를 저장하고 있는 노드가 검색되지 않은 경우, 상기 제3노드의 하위 노드로 상기 제1노드의 위치를 결정하는

스토리지 서버의 데이터 입출력 방법.

청구항 6

제 1항에 있어서,

상기 제1키-값에 대한 해시 테이블 및 버킷을 결정하는 단계는

상기 해시값 중 LSB에 대응되는 아이디의 해시 테이블을, 상기 제1키-값에 대한 해시 테이블로 결정하고,

상기 해시값 중 상기 LSB의 다음 상위 비트값에 대응되는 아이디의 버켓을, 상기 제1키-값에 대한 버킷으로 결정하는

스토리지 서버의 데이터 입출력 방법.

청구항 7

제 1항에 있어서,

상기 계산 노드로부터, 상기 제1키-값에 대한 읽기 요청을 수신하는 단계;

상기 제1키-값의 키에 대한 해시값을 이용하여, 상기 제1노드를 검색하는 단계;

상기 제1노드에 저장된, 상기 제1키-값의 값에 대한 저장 주소를 이용하여, 상기 제1키-값의 값을 로딩하는 단계

를 더 포함하는 스토리지 서버의 데이터 입출력 방법.

청구항 8

제 1항에 있어서,

상기 버전은

동일한 키가 할당된 키-값들이 상기 스토리지 서버에 저장된 순서에 대응되며,

상기 제1노드에 저장되는 상기 제1키-값의 버전값은,

상기 동일한 키가 할당된 키-값에 대한 쓰기 요청이 발생되는 횟수에 비례하여 증가하는

스토리지 서버의 데이터 입출력 방법.

청구항 9

제 8항에 있어서,

상기 제1키-값의 키와 동일한 키가 할당된 제2키-값에 대한 노드의 개수가 임계 개수 이상이 되는 경우, 쓰기 요청 순서의 역순으로, 상기 제2키-값에 대한 노드를 제거하는 단계

를 더 포함하는 스토리지 서버의 데이터 입출력 방법.

청구항 10

메모리;

상기 메모리와 전기적으로 연결된 적어도 하나의 프로세서; 및

계산 노드로부터, 키-값에 대한 쓰기 요청을 수신하는 통신부를 포함하며,

상기 프로세서는

상기 키-값의 키에 대한 해시값을 이용하여, 상기 키-값에 대한 해시 테이블 및 버킷을 결정하고,

상기 키-값의 버전에 따라서, 상기 버킷에 할당된 연결 리스트에서의 상기 키-값이 할당되는 제1노드의 위치를 결정하며,

상기 제1노드에, 상기 키-값의 키 및 상기 제1노드와 제2노드의 연결 관계를 나타내는 포인터를 저장하는

스토리지 서버.

청구항 11

제 10항에 있어서,

상기 프로세서는

상기 연결 리스트의 노드에서, 상기 키-값의 키와 동일한 키를 저장하고 있는 적어도 하나의 노드를 검색하고,

상기 검색된 노드의 위치보다 상위 노드의 위치로, 상기 제1노드의 위치를 결정하는

스토리지 서버.

발명의 설명

기 술 분 야

[0001] 본 발명은 스토리지 서버 및 스토리지 서버의 데이터 입출력 방법에 관한 것으로서, 보다 상세하게는 버저닝 기법을 이용하여, 데이터 손실로 인한 피해를 줄일 수 있는 스토리지 서버 및 스토리지 서버의 데이터 입출력 방법에 관한 것이다.

배 경 기 술

[0003] 전통적인 데이터 센터는 CPU, 메모리, 가속기, 스토리지를 모두 포함하는 서버들이 네트워크로 연결된 클러스터 구조로 되어 있다. 하지만 최근 데이터 센터의 규모가 커짐에 따라, 기존 데이터 센터의 클러스터 구조는 확장성 한계에 직면하고 있다. 따라서 모든 자원들을 단일 서버에 포함하는 서버 중심(server-centric) 구조에서, 각 서버를 자원의 특성에 따라 파티셔닝하여 자원의 풀을 구성하는 자원 중심(resource-centric) 구조로의 전환 시도가 활발하게 진행되고 있다.

[0004] 특히 스토리지 분리화(storage disaggregation)는 계산 노드와 스토리지 노드를 물리적으로 분리하여, 각각 계산 노드 풀과 스토리지 풀로 구성한다. 계산 노드와 스토리지 노드는 고속의 네트워크로 연결되며, NVMe-oF(NVMe over Fabrics) 프로토콜 기반으로, 스토리지 노드에 읽기와 쓰기가 수행된다.

[0005] 한편 최근에는 빅데이터 저장소로, 간단한 키-값 인터페이스를 제공하며 저장 및 관리가 용이한 키-값 스토어(key-value store)가 활발하게 사용되고 있다. 키-값 데이터는 단순한 키-값 메소드를 사용하는 비관계형 데이터로서, 키-값 인터페이스를 제공하는 키-값 스토어는 대용량 데이터의 저장 및 조회가 편리하여 널리 사용되고 있다. 스토리지 분리화 환경에서 키-값 스토어는 계산 노드에서 구동되고, NVMe-oF 프로토콜을 통해 계산 노드와 스토리지 노드 사이에서 네트워크 I/O가 수행된다.

[0006] 관련 선행문헌으로 대한민국 공개특허 제2021-0075845호, 제2021-0123236호. 대한민국 등록특허 제10-2471966호가 있다.

발명의 내용

해결하려는 과제

[0008] 본 발명은 버저닝 기법을 이용하여, 데이터 손실로 인한 피해를 줄일 수 있는 스토리지 서버 및 스토리지 서버의 데이터 입출력 방법을 제공하기 위한 것이다.

과제의 해결 수단

[0010] 상기한 목적을 달성하기 위한 본 발명의 일 실시예에 따르면, 계산 노드로부터, 제1키-값에 대한 쓰기 요청을 수신하는 단계; 상기 제1키-값의 키에 대한 해시값을 이용하여, 상기 제1키-값에 대한 해시 테이블 및 버킷을 결정하는 단계; 상기 제1키-값의 버전에 따라서, 상기 버킷에 할당된 연결 리스트에서, 상기 제1키-값이 할당되는 제1노드의 위치를 결정하는 단계; 및 상기 제1노드에, 상기 제1키-값의 키 및 상기 제1노드와 제2노드의 연결 관계를 나타내는 포인터를 저장하는 단계를 포함하는 스토리지 서버의 데이터 입출력 방법이 제공된다.

[0011] 또한 상기한 목적을 달성하기 위한 본 발명의 다른 실시예에 따르면, 메모리; 상기 메모리와 전기적으로 연결된 적어도 하나의 프로세서; 및 계산 노드로부터, 제1키-값에 대한 쓰기 요청을 수신하는 통신부를 포함하며, 상기 프로세서는 상기 제1키-값의 키에 대한 해시값을 이용하여, 상기 제1키-값에 대한 해시 테이블 및 버킷을 결정하고, 상기 제1키-값의 버전에 따라서, 상기 버킷에 할당된 연결 리스트에서의 상기 제1키-값이 할당되는 제1노드의 위치를 결정하며, 상기 제1노드에, 상기 제1키-값의 키 및 상기 제1노드와 제2노드의 연결 관계를 나타내는 포인터를 저장하는 스토리지 서버가 제공된다.

발명의 효과

[0013] 본 발명의 일실시예에 따르면, 해시 테이블 자료 구조를 통해, 서로 다른 버전의 데이터가 효율적으로 관리될 수 있다.

[0014] 또한 본 발명의 일실시예에 따르면, 최신 버전의 데이터가 삭제되더라도 이전 버전의 데이터로부터 데이터가 복원될 수 있으므로, 데이터 손실이 최소화될 수 있다.

도면의 간단한 설명

[0016] 도 1은 스토리지 분리화 환경에서 이용되는 네트워크 기반 키-값 저장소를 설명하기 위한 도면이다.

도 2는 본 발명의 일실시예에 따른 스토리지 시스템을 설명하기 위한 도면이다.

도 3은 본 발명의 일실시예에 따른 스토리지 서버의 데이터 입출력 방법을 설명하기 위한 도면이다.

도 4는 본 발명의 일실시예에 따른 연결 리스트를 설명하기 위한 도면이다.

발명을 실시하기 위한 구체적인 내용

[0017] 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.

[0018] 이하에서, 본 발명에 따른 실시예들을 첨부된 도면을 참조하여 상세하게 설명한다.

[0020] 도 1은 스토리지 분리화 환경에서 이용되는 네트워크 기반 키-값 저장소를 설명하기 위한 도면이다.

[0021] 도 1에 도시된 바와 같이, 스토리지 분리화 환경에서 키-값 저장소(122)는 계산 노드 즉 클라이언트(110)가 아

닌 스토리지 서버(120)에 위치한다. 복수의 클라이언트(110)는 응용(111, application) 및 NVMe-oF 드라이버(112)를 포함하며, 스토리지 서버(120)는 NVMe-oF 타겟(121), 키-값 저장소(122), NVMe 드라이버(123) 및 NVMe SSD(124)를 포함한다. NVMe-oF 타겟(121), 키-값 저장소(122), NVMe 드라이버(123)는 SPDK를 이용하여 구현될 수 있다.

[0022] 클라이언트의 응용(111)은 키-값에 대한 읽기 또는 쓰기 요청과 같은 입출력(I/O) 요청을 NVMe-oF 드라이버(112)로 전달하고, NVMe-oF 드라이버(112)는 입출력 요청을 NVMe 명령어로 변환한다. 즉, 입출력 요청은 클라이언트의 키-값 저장소와 커널 영역의 파일 시스템을 거치지 않고, 직접 NVMe-oF 드라이버(112)로 전달된다. 입출력 요청이 클라이언트의 파일 시스템을 통과하여, NVMe-oF 드라이버(112)에 의해 NVMe 명령어로 변환되며, 키-값 저장소(122)는 SPDK에 기반하여 스토리지 서버(120)의 사용자 공간에 구현되기 때문에, 커널 I/O 스택의 오버헤드가 감소할 수 있다.

[0023] NVMe-oF 드라이버(112)는 입출력 요청을, 입출력 요청에 대응되는 NVMe 명령어로 변환한 후 스토리지 서버(120)로 전송한다.

[0024] 스토리지 서버(120)로 전송된 NVMe 명령어는 NVMe-oF 타겟(121)에 의해 키-값 저장소(222)로 전달되며, 키-값 저장소(122)는 NVMe 명령어를 블록 I/O 요청으로 변환하여 NVMe 드라이버(123)으로 전달한다. 그리고 NVMe SSD(124)가 블록 I/O 요청에 따라 데이터를 저장하거나 읽어옴으로써, 입출력 요청에 대한 서비스가 클라이언트(110)로 제공될 수 있다.

[0025] 한편, 스토리지 서버에 저장된 데이터는, 랜섬웨어와 같은 악성 소프트웨어나 사용자의 실수로 인해 변조되거나 삭제될 수가 있다. 랜섬웨어는 데이터를 암호화 한 뒤 이를 복원하는 방법에 대가를 지불하지 않을 시 데이터를 파괴하여 기관이나 개인에 막대한 피해를 입힌다. 또한, 사용자의 실수로 데이터가 삭제되거나 갱신되는 일이 빈번히 발생하고 있다.

[0026] 따라서, 스토리지 서버에 저장된 데이터의 손실을 방지할 수 있는 방법이 요구되고 있으며, 본 발명은 저장된 데이터의 손실을 줄일 수 있는 스토리지 서버 및 스토리지 서버의 데이터 입출력 방법을 제안한다. 본 발명의 일실시예는 저장된 데이터의 손실을 줄이기 위해, 버저닝(versioning) 기법을 이용한다. 버저닝 기법은 데이터를 버전(version) 별로 관리하는 기법으로서, 최신 버전의 데이터가 삭제되더라도 이전 버전의 데이터의 복원을 가능하도록 하여, 데이터 손실이 최소화될 수 있다.

[0028] 도 2는 본 발명의 일실시예에 따른 스토리지 시스템을 설명하기 위한 도면이다.

[0029] 도 2를 참조하면 본 발명의 일실시예에 따른 스토리지 시스템은 복수의 클라이언트(210)와 스토리지 서버(220)를 포함한다.

[0030] 클라이언트(210)는 스토리지 서버(220)로 데이터의 입력 또는 출력을 요청하며, 스토리지 서버(220)는 클라이언트(210)의 입출력 요청에 대한 서비스를 클라이언트(210)로 제공한다. 즉, 스토리지 서버(220)는 클라이언트(210)의 입력 요청된 데이터를 저장하고, 출력 요청된 데이터를 클라이언트(210)로 전송한다. 스토리지 서버(220)는 메모리(221), 적어도 하나의 프로세서(222) 및 통신부(223)를 포함한다.

[0031] 통신부(223)는 클라이언트(210) 즉, 계산 노드로부터, 키-값에 대한 쓰기 요청 또는 읽기 요청을 수신한다. 통신부(223)는 입력 요청된 데이터를 클라이언트(210)로부터 수신하고, 출력 요청된 데이터를 클라이언트(210)로 전송한다.

[0032] 프로세서(222)는 메모리(221)와 전기적으로 연결되며, 클라이언트(210)의 입출력 요청에 응답하여, 데이터 입출력을 위한 일련의 과정을 수행한다. 프로세서(222)는 일실시예로서 해시 테이블 자료 구조를 이용하여, 데이터의 입출력을 처리할 수 있다.

[0033] 프로세서(222)는 클라이언트(210)로부터 수신된 키-값에 대한 쓰기 요청에 따라서, 키-값의 값을 스토리지에 저장한다. 이 때 프로세서(222)는 쓰기 요청된 키-값의 이전 버전의 값을 삭제하지 않고, 최신 버전의 키-값의 값을 스토리지에 추가로 저장한다. 이 때 프로세서(222)는 해시 테이블 자료 구조를 이용하여, 버전별 값을 관리할 수 있다.

[0034] 프로세서(222)는 쓰기 요청된 키-값의 키에 대한 해시값을 이용하여, 키-값에 대한 해시 테이블(hash table) 및 버킷(bucket)을 결정한다. 그리고 키-값의 버전에 따라서, 결정된 버킷에 할당된 연결 리스트(linked list)에서

의 키-값이 할당되는 제1노드의 위치를 결정한다. 프로세서(222)는 연결 리스트의 노드에서, 쓰기 요청된 키-값의 키와 동일한 키를 저장하고 있는 적어도 하나의 노드를 검색하여, 제1노드의 위치를 결정할 수 있다. 그리고 제1노드에, 키-값의 키 및 제1노드와 제2노드의 연결 관계를 나타내는 포인터를 저장한다. 여기서, 제2노드는 쓰기 요청된 키-값의 이전 버전의 키-값에 대한 노드일 수 있다.

[0035] 즉, 프로세서(222)는 최신 버전의 키-값에 대한 노드의 위치를 결정하고, 결정된 위치에 제1노드를 생성하되, 제2노드는 삭제하지 않고 연결 리스트를 갱신하며, 최신 버전의 키-값의 값을 추가로 스토리지에 저장한다.

[0036] 이와 같이 본 발명의 일실시예는 해시 테이블 자료 구조를 이용하여, 서로 다른 버전의 데이터를 관리하며, 본 발명의 일실시예에 따르면, 최신 버전의 데이터가 삭제되더라도 이전 버전의 데이터로부터 데이터가 복원될 수 있으므로, 데이터 손실이 최소화될 수 있다.

[0038] 도 3은 본 발명의 일실시예에 따른 스토리지 서버의 데이터 입출력 방법을 설명하기 위한 도면이며, 도 4는 본 발명의 일실시예에 따른 연결 리스트를 설명하기 위한 도면이다.

[0039] 도 3을 참조하면 본 발명의 일실시예에 따른 스토리지 서버는 계산 노드로부터, 제1키-값에 대한 쓰기 요청을 수신(S310)하고, 쓰기 요청된 제1키-값의 키에 대한 해시값을 이용하여, 제1키-값에 대한 해시 테이블 및 버킷을 결정(S320)한다.

[0040] 단계 S320에서 스토리지 서버는 일실시예로서, 미리 설정된 해시 함수를 이용해, 제1키-값의 키로부터 4바이트의 해시값을 생성할 수 있다. 그리고 해시값 중 LSB(Least Significant Bit)에 대응되는 아이디의 해시 테이블을, 제1키-값에 대한 해시 테이블로 결정할 수 있다. LSB가 N(자연수) 비트로 설정된 경우, 해시 테이블은 2^N개로 이루어질 수 있으며, 2^N개의 해시 테이블 중 하나가, 제1키-값에 대한 해시 테이블로 결정될 수 있다.

[0041] 그리고 스토리지 서버는 해시값 중 LSB의 다음 상위 비트값에 대응되는 아이디의 버켓을, 제1키-값에 대한 버킷으로 결정할 수 있다. 상위 비트가 M(자연수) 비트로 설정된 경우, 버킷은 2^M개로 이루어질 수 있으며, 2^M개의 버킷 중 하나가, 제1키-값에 대한 버킷으로 결정될 수 있다.

[0042] 제1키-값에 대한 해시 테이블 및 버킷이 결정되면, 스토리지 서버는 제1키-값의 버전에 따라서, 결정된 버킷에 할당된 연결 리스트에서, 제1키-값이 할당되는 제1노드의 위치를 결정(S330)한다.

[0043] 여기서 연결 리스트는 복수의 노드가 연결된 형태로서, 노드에는 키-값에 대한 데이터가 저장된다. 그리고 키-값의 버전은, 동일한 키가 할당된 키-값들이 스토리지 서버에 저장된 순서에 대응되며, 최근에 저장된 키-값들일수록 최신 버전의 키-값이라고 볼 수 있다. 예컨대, 사용자가 A라는 파일에 대해 쓰기 요청한 이후, 파일 A를 수정하여 다시 쓰기 요청한다면, 수정전 파일 A에 대한 쓰기 요청에 이용된 키-값보다, 수정된 파일 A에 대한 쓰기 요청에 이용된 키-값의 버전이, 최신 버전이라고 할 수 있다. 동일한 파일의 입출력 요청에 이용되는 키-값에는, 동일한 키가 할당될 수 있다.

[0044] 제1노드의 위치를 결정하기 위해 스토리지 서버는, 연결 리스트의 노드에서, 제1키-값의 키와 동일한 키가 할당된 제2키-값에 대한 노드를 검색한다. 그리고 검색된 노드의 위치보다 상위 노드의 위치로, 제1노드의 위치를 결정한다. 검색된 노드가 존재한다는 것은, 제1키-값의 버전의 이전 버전인 제2키-값에 대한 쓰기 요청이 있었음을 의미하며, 스토리지 서버는 검색된 노드 중 최상위 노드보다 상위인 노드의 위치로, 제1노드의 위치를 결정한다. 여기서, 상위 노드란 연결 리스트의 헤드 노드에 상대적으로 가까운 노드에 대응되며, 하위 노드란 헤드 노드에 상대적으로 멀리 떨어진 노드에 대응된다.

[0045] 스토리지 서버는 제1노드의 위치가 결정되면, 결정된 위치에 제1노드를 생성하고, 제1노드에, 제1키-값의 키 및 제1노드와 제2노드의 연결 관계를 나타내는 포인터(pointer)를 저장(S340)한다. 제2노드는 단계 S330에서 검색된 노드 중, 헤드 노드에 가장 가까운 노드 즉, 최상위 노드에 대응된다. 다시 말해, 제1노드는, 검색된 노드 중 최상위 노드와 연결된다.

[0046] 실시예에 따라서 제1노드에는 제1키-값의 값에 대한 저장 주소, 제1키-값에 대한 버전값이 더 포함될 수 있다. 버전값은 동일한 키가 할당된 키-값에 대한 쓰기 요청이 발생되는 횟수에 비례하여 증가할 수 있으며, 버전값이 높을수록 최신 버전을 나타낸다.

[0047] 도 4는 본 발명의 일실시예에 따른 연결 리스트를 나타내는 도면으로서, 연결 리스트는 버킷(420) 중 하나에 할

당되고, 버킷(420)은 해시 테이블(410) 중 하나에 할당된다. 도 4(a)에 도시된 연결 리스트, 제1키-값의 키(K1)와 동일한 키(K1)를 저장하고 있는 제2노드(432) 및 제4노드(434)가 검색된 경우, 스토리지 서버는 도 4(b)와 같이, 제2 및 제4노드(432, 434) 중 최상위 노드인 제2노드(432)보다 상위 노드로서, 제1노드(431)를 생성한다.

[0048] 그리고 도 4(a)에 도시된 바와 같이, 제1키-값의 키(K1)와 상이한 키(K4)가 저장된 제3노드(433)와, 제2노드(432)가 연결된 상태에서, 제1키-값에 대한 쓰기 요청이 수신된 경우, 도 4(b)에 도시된 바와 같이, 스토리지 서버는 제1노드(431)를 제3노드(433)와 연결시키고, 제1노드(431)와 제3노드(433)의 연결 관계를 나타내는 포인터를 추가로 제1노드(431)에 저장할 수 있다. 도 4에서 화살표는, 포인터를 나타낸다. 이 때, 스토리지 서버는 동시에 수신되는 복수의 키-값 쓰기 요청을 동시에 처리하기 위해, Fine-grained 락(lock) 기법을 적용할 수 있다.

[0049] 만일, 단계 S330에서, 제1키-값의 키와 동일한 키를 저장하고 있는 노드가 검색되지 않은 경우, 스토리지 서버는 제3노드의 하위 노드로 제1노드의 위치를 결정할 수 있다.

[0050] 전술된 바와 같이, 스토리지 서버는 제1키-값 쓰기 요청이 수신된 경우, 제1키-값에 대한 노드를 생성하여, 연결 리스트를 갱신하고, 제1키-값의 값을 스토리지에 저장한다. 저장 위치는 전술된 저장 주소에 대응되며, 스토리지는 NVMe SSD일 수 있다.

[0051] 이후, 스토리지가 계산 노드로부터, 제1키-값에 대한 읽기 요청을 수신(S350)하면, 제1키-값의 키에 대한 해시값을 이용하여, 연결 리스트에서 제1노드를 검색(S360)하고, 제1노드에 저장된, 제1키-값의 값에 대한 저장 주소를 이용하여, 제1키-값의 값을 스토리지로부터 로딩(S370)할 수 있다.

[0052] 한편, 데이터의 버전이 지속적으로 갱신됨에도 이전 버전의 데이터가 삭제되지 않는다면, 스토리지의 가용 저장 공간이 급격히 감소할 수 있다. 따라서, 스토리지 서버는 제1키-값의 키와 동일한 키가 할당된 제2키-값에 대한 노드의 개수가 임계 개수 이상이 되는 경우, 쓰기 요청 순서의 역순으로, 제2키-값에 대한 노드를 제거할 수 있다.

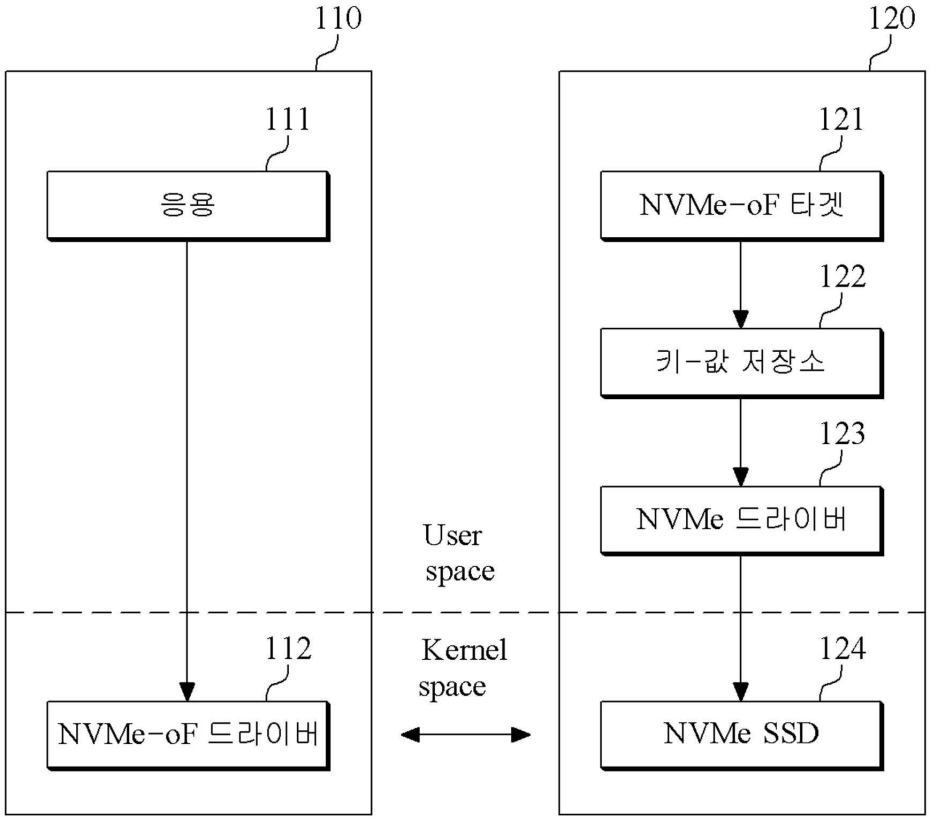
[0053] 이 때, 임계 개수는 스토리지의 가용 저장 공간에 따라서 결정되거나, 키-값에 대한 읽기 요청 횟수에 따라서 결정될 수 있다. 임계 개수는 스토리지의 가용 저장 공간에 비례하도록 결정되거나, 읽기 요청 횟수에 비례하도록 결정될 수 있다.

[0055] 앞서 설명한 기술적 내용들은 다양한 컴퓨터 수단을 통하여 수행될 수 있는 프로그램 명령 형태로 구현되어 컴퓨터 판독 가능 매체에 기록될 수 있다. 상기 컴퓨터 판독 가능 매체는 프로그램 명령, 데이터 파일, 데이터 구조 등을 단독으로 또는 조합하여 포함할 수 있다. 상기 매체에 기록되는 프로그램 명령은 실시예들을 위하여 특별히 설계되고 구성된 것들이거나 컴퓨터 소프트웨어 당업자에게 공지되어 사용 가능한 것일 수도 있다. 컴퓨터 판독 가능 기록 매체의 예에는 하드 디스크, 플로피 디스크 및 자기 테이프와 같은 자기 매체(magnetic media), CD-ROM, DVD와 같은 광기록 매체(optical media), 플롭티컬 디스크(floptical disk)와 같은 자기-광 매체(magneto-optical media), 및 롬(ROM), 램(RAM), 플래시 메모리 등과 같은 프로그램 명령을 저장하고 수행하도록 특별히 구성된 하드웨어 장치가 포함된다. 프로그램 명령의 예에는 컴파일러에 의해 만들어지는 것과 같은 기계어 코드뿐만 아니라 인터프리터 등을 사용해서 컴퓨터에 의해서 실행될 수 있는 고급 언어 코드를 포함한다. 하드웨어 장치는 실시예들의 동작을 수행하기 위해 하나 이상의 소프트웨어 모듈로서 작동하도록 구성될 수 있으며, 그 역도 마찬가지이다.

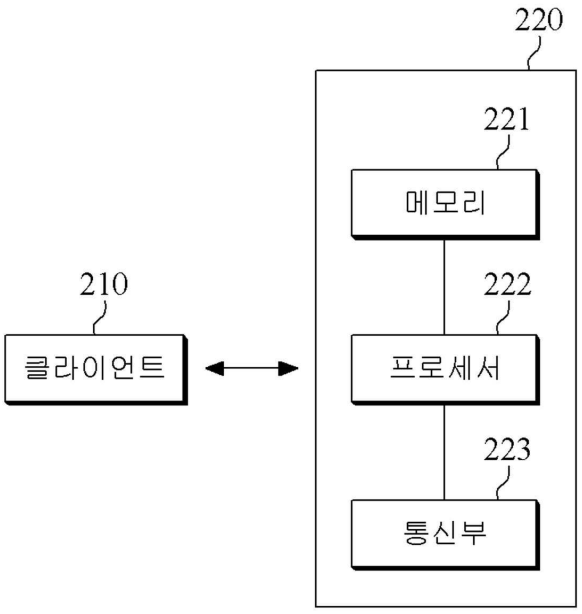
[0057] 이상과 같이 본 발명에서는 구체적인 구성 요소 등과 같은 특정 사항들과 한정된 실시예 및 도면에 의해 설명되었으나 이는 본 발명의 보다 전반적인 이해를 돕기 위해서 제공된 것일 뿐, 본 발명은 상기의 실시예에 한정되는 것은 아니며, 본 발명이 속하는 분야에서 통상적인 지식을 가진 자라면 이러한 기재로부터 다양한 수정 및 변형이 가능하다. 따라서, 본 발명의 사상은 설명된 실시예에 국한되어 정해져서는 아니되며, 후술하는 특허청구범위뿐 아니라 이 특허청구범위와 균등하거나 등가적 변형이 있는 모든 것들은 본 발명 사상의 범주에 속한다고 할 것이다.

도면

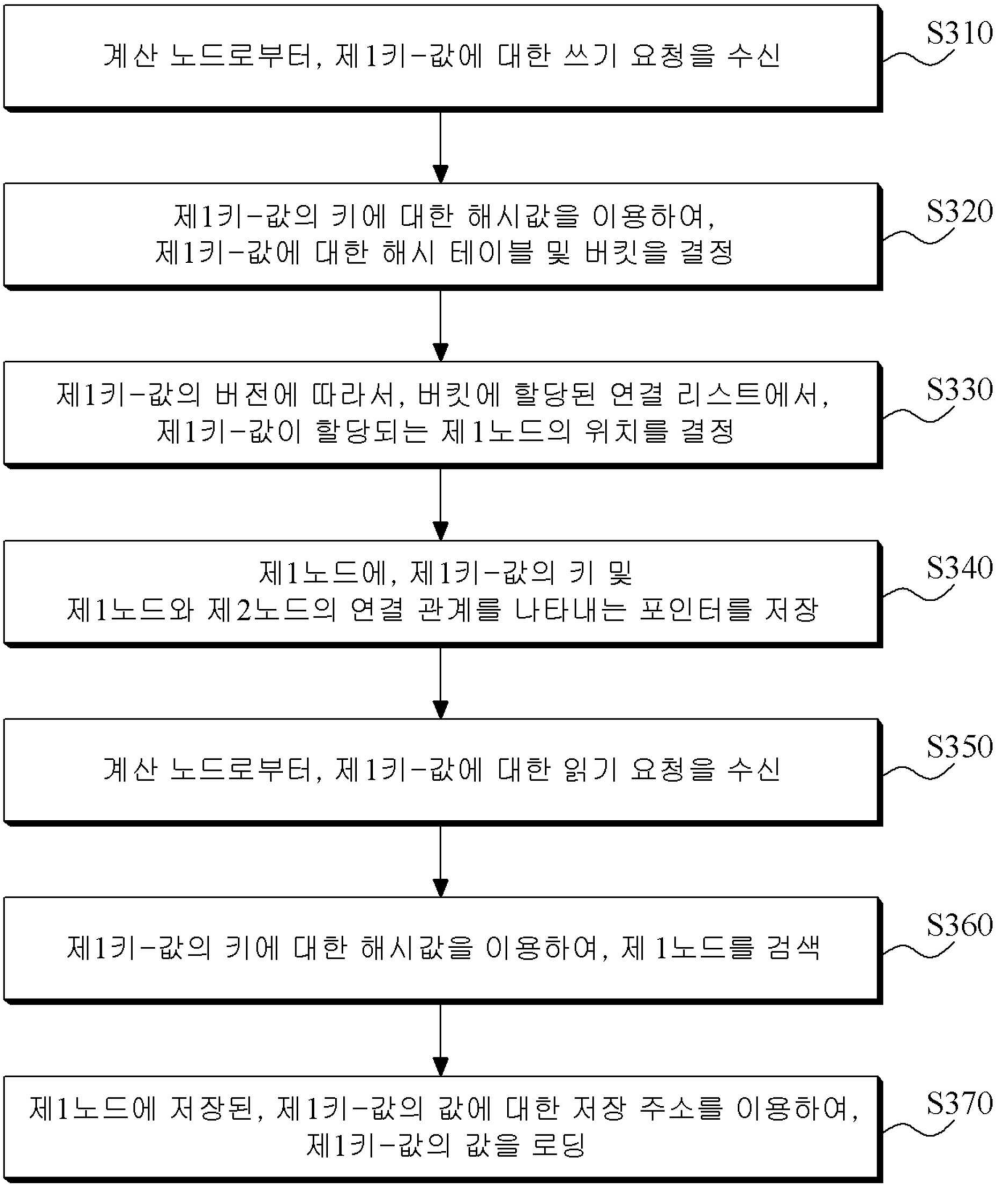
도면1


110
120
111
응용
121
NVMe-oF 타겟
122
키-값 저장소
123
NVMe 드라이버
User space
Kernel space
112
NVMe-oF 드라이버
124
NVMe SSD

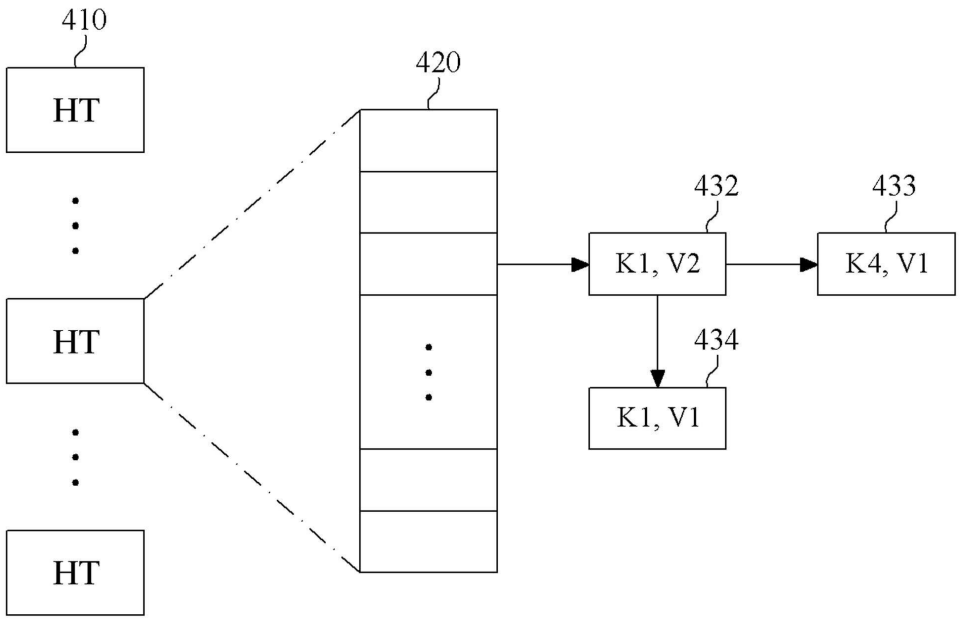
도면2


220
221
메모리
210
222
클라이언트
프로세서
223
통신부

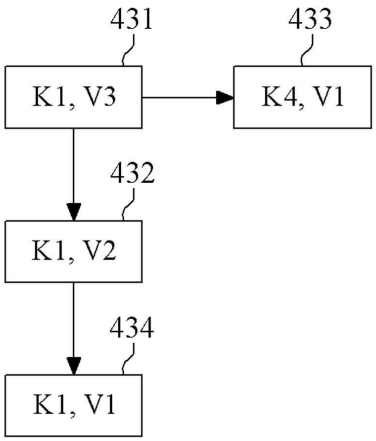
도면3


계산 노드로부터, 제1키-값에 대한 쓰기 요청을 수신
S310
제1키-값의 키에 대한 해시값을 이용하여,
제1키-값에 대한 해시 테이블 및 버킷을 결정
S320
제1키-값의 버전에 따라서, 버킷에 할당된 연결 리스트에서,
제1키-값이 할당되는 제1노드의 위치를 결정
S330
제1노드에, 제1키-값의 키 및
제1노드와 제2노드의 연결 관계를 나타내는 포인터를 저장
S340
계산 노드로부터, 제1키-값에 대한 읽기 요청을 수신
S350
제1키-값의 키에 대한 해시값을 이용하여, 제1노드를 검색
S360
제1노드에 저장된, 제1키-값의 값에 대한 저장 주소를 이용하여,
제1키-값의 값을 로딩
S370

도면4


410
HT
HT
HT
420
432
K1, V2
433
K4, V1
434
K1, V1

(a)


431
K1, V3
433
K4, V1
432
K1, V2
434
K1, V1

(b)