



(19) 대한민국특허청(KR)  
(12) 등록특허공보(B1)

(45) 공고일자 2017년03월29일  
(11) 등록번호 10-1720602  
(24) 등록일자 2017년03월22일

- (51) 국제특허분류(Int. Cl.)  
*G06F 17/30* (2006.01)
- (52) CPC특허분류  
*G06F 17/30587* (2013.01)  
*G06F 17/30315* (2013.01)
- (21) 출원번호 10-2016-7005763
- (22) 출원일자(국제) 2013년08월29일  
심사청구일자 2016년03월03일
- (85) 번역문제출일자 2016년03월03일
- (65) 공개번호 10-2016-0040282
- (43) 공개일자 2016년04월12일
- (86) 국제출원번호 PCT/CN2013/082544
- (87) 국제공개번호 WO 2015/027425  
국제공개일자 2015년03월05일
- (56) 선행기술조사문현  
US20020095421 A1  
US7493313 A  
US6769124 A  
US6754670 A

- (73) 특허권자  
후아웨이 테크놀러지 컴퍼니 리미티드  
중국 518129 광동 샌젠 롱강 디스트릭트 반티안  
후아웨이 어드미니스트레이션 빌딩
- (72) 발명자  
비 지에샨  
중국 518129 광동 샌젠 롱강 디스트릭트 반티안  
후아웨이 어드미니스트레이션 빌딩  
지 웨이  
중국 518129 광동 샌젠 롱강 디스트릭트 반티안  
후아웨이 어드미니스트레이션 빌딩
- (74) 대리인  
유미특허법인

전체 청구항 수 : 총 13 항

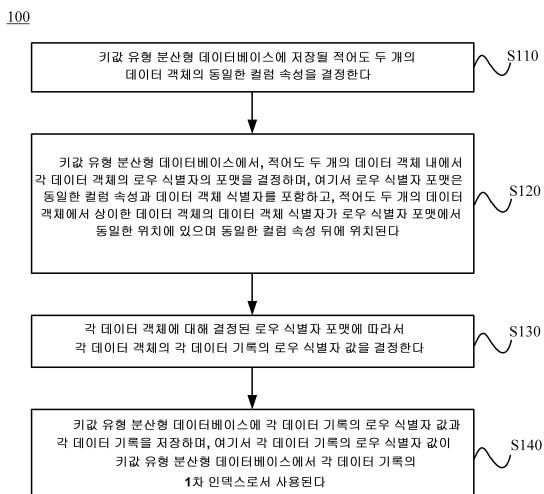
심사관 : 김경완

## (54) 발명의 명칭 데이터를 저장하기 위한 방법 및 장치

**(57) 요약**

본 발명은 데이터 저장 방법 및 장치를 개시한다. 본 방법은, 키값 유형 분산형 데이터베이스(KeyValue type distributed database)에 저장될 적어도 두 개의 데이터 객체의 동일한 컬럼 속성을 결정하는 단계; 상기 키값 유형 분산형 데이터베이스에서, 상기 적어도 두 개의 데이터 객체에서 각 데이터 객체의 로우(row) 식별자의 포

(뒷면에 계속)

**대 표 도 - 도1**

맷을 결정하는 단계 - 상기 로우 식별자의 포맷이 상기 동일한 컬럼 속성과 데이터 객체 식별자를 포함하고, 상기 적어도 두 개의 데이터 객체에서 상이한 데이터 객체의 데이터 객체 식별자가 상기 로우 식별자의 포맷에서 동일한 위치에 있으며 상기 동일한 컬럼 속성 뒤에 위치됨 -; 각 데이터 객체에 대해 결정된 상기 로우 식별자의 포맷에 따라서 각 데이터 객체의 각 데이터 기록의 로우 식별자 값을 결정하는 단계; 및 각 데이터 기록의 상기 로우 식별자 값과 각 데이터 기록을 상기 키값 유형 분산형 데이터베이스에 저장하는 단계를 포함하며, 각 데이터 기록의 상기 로우 식별자 값이 상기 키값 유형 분산형 데이터베이스에서 각 데이터 기록의 1차 인덱스(primary index)로서 사용된다. 본 발명의 실시예에 따른 데이터 저장 방법 및 장치는 데이터 쿼리 효율을 향상 시킬 수 있다.

(52) CPC특허분류

*G06F 17/30321* (2013.01)

---

## 명세서

### 청구범위

#### 청구항 1

컴퓨터 장치에 의해 수행되는 데이터 저장 방법으로서,

키값 유형 분산형 데이터베이스(KeyValue type distributed database)에 저장될 적어도 두 개의 데이터 객체의 동일한 컬럼 속성을 결정하는 단계 - 각 데이터 객체는 데이터 기록을 가짐 -;

상기 키값 유형 분산형 데이터베이스에서, 상기 적어도 두 개의 데이터 객체의 각 데이터 객체의 로우(row) 식별자의 포맷을 결정하는 단계 - 상기 로우 식별자의 포맷이 상기 동일한 컬럼 속성과 데이터 객체 식별자를 포함하고, 상기 적어도 두 개의 데이터 객체에서 상이한 데이터 객체의 데이터 객체 식별자가 상기 로우 식별자의 포맷에서 동일한 위치에 있으며 상기 동일한 컬럼 속성 뒤에 위치함 -;

상기 결정된 상기 로우 식별자의 포맷에 따라서 각 데이터 객체의 각 데이터 기록에 대한 로우 식별자 값을 결정하는 단계;

상기 데이터 기록 모두의 로우 식별자 값에 따라 제1 순서로 상기 적어도 두 개의 데이터 객체의 데이터 기록 모두를 배열하는 단계 - 상기 적어도 두 개의 데이터 객체의 데이터 기록 모두가 클러스터 테이블(cluster table)을 형성함 -; 및

상기 클러스터 테이블 내의 각 데이터 기록 및 각 데이터 기록의 로우 식별자 값을 상기 키값 유형 분산형 데이터베이스에 저장하는 단계 - 각 데이터 기록의 상기 로우 식별자 값은, 상기 키값 유형 분산형 데이터베이스에서, 각 데이터 기록의 1차 인덱스(primary index)로서 사용됨 -;

상기 적어도 두 개의 데이터 객체에서 데이터 기록을 쿼리(query)하거나 또는 획득하도록 지시하는 데이터 요청을 수신하는 단계;

상기 로우 식별자의 포맷에 따라서, 상기 키값 유형 분산형 데이터베이스를 쿼리하기 위한, 상기 동일한 컬럼 속성을 포함하는 쿼리 정보를 구성하는 단계; 및

상기 쿼리 정보에 따라서, 1차 인덱스가 상기 쿼리 정보를 포함하는 데이터 기록을 상기 키값 유형 분산형 데이터베이스로부터 내보내는(export) 단계

를 포함하는 데이터 저장 방법.

#### 청구항 2

제1항에 있어서,

상기 제1 순서가 사전식 순서(lexicographic order)인,

데이터 저장 방법.

#### 청구항 3

제1항에 있어서,

상기 키값 유형 분산형 데이터베이스에서, 상기 적어도 두 개의 데이터 객체의 각 데이터 객체의 로우(row) 식별자의 포맷을 결정하는 단계가,

상기 적어도 두 개의 데이터 객체에 대해 상기 키값 유형 분산형 데이터베이스에서 상기 로우 식별자의 제1 프리픽스(prefix)를 특정하는 단계 - 상기 제1 프리픽스는 상수임 -; 및

상기 적어도 두 개의 데이터 객체 중 제1 데이터 객체의 로우 식별자의 포맷이 상기 제1 프리픽스, 상기 동일한 컬럼 속성, 및 상기 제1 데이터 객체의 데이터 객체 식별자를 포함하는 것으로 결정하는 단계

를 포함하는,

데이터 저장 방법.

#### 청구항 4

제1항에 있어서

상기 키값 유형 분산형 데이터베이스에서, 상기 적어도 두 개의 데이터 객체의 각 데이터 객체의 로우(row) 식별자의 포맷을 결정하는 단계가,

상기 적어도 두 개의 데이터 객체에 대해 상기 키값 유형 분산형 데이터베이스에서 상기 로우 식별자의 제1 프리픽스(prefix)를 특정하는 단계 - 상기 제1 프리픽스는 상수임 -; 및

상기 적어도 두 개의 데이터 객체 중 제2 데이터 객체의 로우 식별자의 포맷이 상기 제1 프리픽스, 상기 동일한 컬럼 속성, 상기 제2 데이터 객체의 데이터 객체 식별자, 및 상기 제2 데이터 객체의 컬럼 속성에 있으면서 상기 제2 데이터 객체의 데이터 기록 모두를 구별할 수 있는 컬럼 속성을 포함하는 것으로 결정하는 단계

를 포함하는,

데이터 저장 방법.

#### 청구항 5

제3항 또는 제4항에 있어서,

상기 키값 유형 분산형 데이터베이스에서, 상기 적어도 두 개의 데이터 객체 중 임의의 데이터 객체의 2차 인덱스(secondary index)의 포맷이 제2 프리픽스, 제2 속성, 및 상기 동일한 컬럼 속성인 것으로 결정하는 단계 - 상기 제2 속성은 상기 임의의 데이터 객체의 컬럼 속성에 있으면서 상기 동일한 컬럼 속성과는 상이한 컬럼 속성이고, 상기 제2 프리픽스는 상기 제1 프리픽스와 상이한 상수임 -;

상기 임의의 데이터 객체의 상기 2차 인덱스의 포맷에 따라서 상기 임의의 데이터 객체의 상기 2차 인덱스를 결정하는 단계; 및

상기 키값 유형 분산형 데이터베이스에 상기 임의의 데이터 객체의 상기 2차 인덱스를 저장하는 단계

를 더 포함하는 데이터 저장 방법.

#### 청구항 6

프로세서와 메모리를 포함하는 데이터 저장 장치로서,

상기 메모리는 복수의 명령을 가지며, 상기 명령은 상기 프로세서에 의해 실행된 때 상기 프로세서로 하여금:

키값 유형 분산형 데이터베이스(KeyValue type distributed database)에 저장될 적어도 두 개의 데이터 객체의 동일한 컬럼 속성을 결정하는 단계 -;

상기 키값 유형 분산형 데이터베이스에서, 상기 적어도 두 개의 데이터 객체의 각 데이터 객체의 로우(row) 식별자의 포맷을 결정하는 단계 - 상기 로우 식별자의 포맷이 상기 동일한 컬럼 속성과 데이터 객체 식별자를 포함하고, 상기 적어도 두 개의 데이터 객체에서 상이한 데이터 객체의 데이터 객체 식별자가 상기 로우 식별자의 포맷에서 동일한 위치에 있으며 상기 동일한 컬럼 속성 뒤에 위치함 -;

상기 결정된 상기 로우 식별자의 포맷에 따라서 각 데이터 객체의 각 데이터 기록에 대한 로우 식별자 값을 결정하는 단계;

상기 데이터 기록 모두의 로우 식별자 값에 따라 제1 순서로 상기 적어도 두 개의 데이터 객체의 데이터 기록 모두를 배열하는 단계 - 상기 적어도 두 개의 데이터 객체의 데이터 기록 모두가 클러스터 테이블(cluster table)을 형성함 -; 및

상기 클러스터 테이블 내의 각 데이터 기록 및 각 데이터 기록의 로우 식별자 값을 상기 키값 유형 분산형 데이터베이스에 저장하는 단계 - 각 데이터 기록의 상기 로우 식별자 값은, 상기 키값 유형 분산형 데이터베이스에서, 각 데이터 기록의 1차 인덱스(primary index)로서 사용됨 -;

상기 적어도 두 개의 데이터 객체에서 데이터 기록을 쿼리(query)하거나 또는 획득하도록 지시하는 데이터 요청

을 수신하는 단계; 및

상기 로우 식별자의 포맷에 따라서, 상기 키값 유형 분산형 데이터베이스를 쿼리하기 위한, 상기 동일한 컬럼 속성을 포함하는 쿼리 정보를 구성하고, 상기 쿼리 정보에 따라서, 1차 인덱스가 상기 쿼리 정보를 포함하는 데이터 기록을 상기 키값 유형 분산형 데이터베이스로부터 내보내는(export) 단계

를 실행시키는,

데이터 저장 장치.

### 청구항 7

제6항에 있어서,

상기 제1 순서가 사전식 순서(lexicographic order)인,

데이터 저장 장치.

### 청구항 8

제6항에 있어서,

상기 명령이 추가로, 상기 프로세서로 하여금,

상기 적어도 두 개의 데이터 객체에 대해 상기 키값 유형 분산형 데이터베이스에서 상기 로우 식별자의 제1 프리픽스(prefix)를 특정하는 단계 - 상기 제1 프리픽스는 상수임 -; 및

상기 적어도 두 개의 데이터 객체 중 제1 데이터 객체의 로우 식별자의 포맷이 상기 제1 프리픽스, 상기 동일한 컬럼 속성, 및 상기 제1 데이터 객체의 데이터 객체 식별자를 포함하는 것으로 결정하는 단계

를 더 실행시키는

데이터 저장 장치.

### 청구항 9

제6항에 있어서

상기 명령이 추가로, 상기 프로세서로 하여금,

상기 적어도 두 개의 데이터 객체에 대해 상기 키값 유형 분산형 데이터베이스에서 상기 로우 식별자의 제1 프리픽스(prefix)를 특정하는 단계 - 상기 제1 프리픽스는 상수임 -; 및

상기 적어도 두 개의 데이터 객체 중 제2 데이터 객체의 로우 식별자의 포맷이 상기 제1 프리픽스, 상기 동일한 컬럼 속성, 상기 제2 데이터 객체의 데이터 객체 식별자, 및 상기 제2 데이터 객체의 컬럼 속성에 있으면서 상기 제2 데이터 객체의 데이터 기록 모두를 구별할 수 있는 컬럼 속성을 포함하는 것으로 결정하는 단계

를 더 실행시키는,

데이터 저장 장치.

### 청구항 10

제6항에 있어서,

상기 쿼리 정보가 제1 프리픽스 및 상기 동일한 컬럼 속성을 포함하는, 데이터 저장 장치.

### 청구항 11

제8항 또는 제9항에 있어서,

상기 명령이 추가로, 상기 프로세서로 하여금,

상기 키값 유형 분산형 데이터베이스에서, 상기 적어도 두 개의 데이터 객체 중 임의의 데이터 객체의 2차 인덱스(secondary index)의 포맷이 제2 프리픽스, 제2 속성, 및 상기 동일한 컬럼 속성인 것으로 결정하는 단계 - 상기 제2 속성은 상기 임의의 데이터 객체의 컬럼 속성에 있으면서 상기 동일한 컬럼 속성과는 상이한 컬럼 속

성이고, 상기 제2 프리픽스는 상기 제1 프리픽스와 상이한 상수임 -;

상기 임의의 데이터 객체의 상기 2차 인덱스의 포맷에 따라서 상기 임의의 데이터 객체의 상기 2차 인덱스를 결정하는 단계; 및

상기 키값 유형 분산형 데이터베이스에 상기 임의의 데이터 객체의 상기 2차 인덱스를 저장하는 단계

를 더 실행시키는,

데이터 저장 장치.

### 청구항 12

제6항에 있어서,

상기 로우 식별자의 포맷 내의 필드들 간에 분리자가 설정되어 있거나, 또는 상기 필드들이 고정 길이를 가지는, 데이터 저장 장치.

### 청구항 13

컴퓨터 명령을 저장하는 컴퓨터에 의해 관독 가능한 매체로서,

컴퓨터의 프로세서가 상기 컴퓨터 명령을 실행한 때, 상기 컴퓨터 명령이 상기 프로세서로 하여금:

키값 유형 분산형 데이터베이스(KeyValue type distributed database)에 저장될 적어도 두 개의 데이터 객체의 동일한 컬럼 속성을 결정하는 단계;

상기 키값 유형 분산형 데이터베이스에서, 상기 적어도 두 개의 데이터 객체의 각 데이터 객체의 로우(row) 식별자의 포맷을 결정하는 단계 - 상기 로우 식별자의 포맷이 상기 동일한 컬럼 속성과 데이터 객체 식별자를 포함하고, 상기 적어도 두 개의 데이터 객체에서 상이한 데이터 객체의 데이터 객체 식별자가 상기 로우 식별자의 포맷에서 동일한 위치에 있으며 상기 동일한 컬럼 속성 뒤에 위치함 -;

각 데이터 객체의 상기 결정된 상기 로우 식별자의 포맷에 따라서 각 데이터 객체의 각 데이터 기록에 대한 로우 식별자 값을 결정하는 단계;

상기 데이터 기록 모두의 로우 식별자 값에 따라 제1 순서로 상기 적어도 두 개의 데이터 객체의 데이터 기록 모두를 배열하는 단계 - 상기 적어도 두 개의 데이터 객체의 데이터 기록 모두가 클러스터 테이블(cluster table)을 형성함 -; 및

상기 클러스터 테이블 내의 각 데이터 기록 및 각 데이터 기록의 로우 식별자 값을 상기 키값 유형 분산형 데이터베이스에 저장하는 단계 - 각 데이터 기록의 상기 로우 식별자 값은, 상기 키값 유형 분산형 데이터베이스에서, 각 데이터 기록의 1차 인덱스(primary index)로서 사용됨 -;

상기 적어도 두 개의 데이터 객체에서 데이터 기록을 쿼리(query)하거나 또는 획득하도록 지시하는 데이터 요청을 수신하는 단계; 및

상기 로우 식별자의 포맷에 따라서, 상기 키값 유형 분산형 데이터베이스를 쿼리하기 위한, 상기 동일한 컬럼 속성을 포함하는 쿼리 정보를 구성하고, 상기 쿼리 정보에 따라서, 1차 인덱스가 상기 쿼리 정보를 포함하는 데이터 기록을 상기 키값 유형 분산형 데이터베이스로부터 내보내는(export) 단계

를 실행시키는,

컴퓨터에 의해 관독 가능한 매체.

### 청구항 14

삭제

### 청구항 15

삭제

### 청구항 16

삭제

청구항 17

삭제

청구항 18

삭제

## 발명의 설명

### 기술 분야

[0001]

본 발명은 정보 기술 분야에 관한 것으로, 구체적으로는 데이터 저장 방법 및 장치에 관한 것이다.

### 배경 기술

[0002]

분산형 컬럼 저장 데이터베이스(distributed column storage database)로서, 키값(KeyValue) 유형 분산형 데이터베이스는 높은 확장성과 견고성을 가지며, 점점 더 많은 시스템들에 폭넓게 응용되어 왔다. 키값 유형 분산형 데이터베이스의 사용자 테이블은 일반적으로 비교적 간단한 데이터 포맷이면서, 간단한 상관관계를 가지지만, 다양한 데이터를 저장하도록 설계된다. 예를 들어, 웹 페이지 어드레스 정보를 저장하거나, 통화 기록 정보를 저장하거나, 또는 네트워크 액세스 기록 정보 등을 저장하도록 설계된다. 키값 유형 분산형 데이터베이스는 데이터 레코드의 로우키(RowKey)에 따라서 고속의 쿼리(query)를 제공할 수 있으며, 고속의 쿼리는 데이터 양과는 상관없다. 물리 노드는, 현재의 저장 공간 사용이 임계값(threshold)에 도달하는 경우 키값 유형 분산형 데이터베이스에 동적으로 부가될 수 있다.

[0003]

기존의 기업의 데이터베이스 애플리케이션에서, 통상적으로 많은 사용자 데이터 테이블이 관련되고, 이들 데이터 테이블은 상이한 크기를 가지며, 테이블들의 데이터 사이에 내부의 상관관계가 존재할 수 있다. 그러나, 희소 매트릭스(sparse matrix)에 기초한 키값 유형 분산형 데이터베이스는 대량의 데이터에 테이블을 저장하는데 적합하다. 만약 이들 원형의 테이블이 키값 유형 분산형 데이터베이스에 직접 도입되는 경우 과도한 소형 테이블들이 존재하게 되며, 따라서, 크로스-테이블(cross-table)의 관련 쿼리를 구현하기가 어려우며, 관리 복잡도 또한 증가하게 된다. 즉, 종래 애플리케이션에서의 데이터가, 한 사용자 테이블과 다른 사용자 테이블 사이의 상관관계 쿼리를 완성하기 위해, 키값 유형 분산형 데이터베이스내로 이동되는 경우, 상이한 테이블들이 쿼리되어야 하며, 데이터 로케이팅(locating)이 끊임없이 수행되어야 하므로, 효율이 낮아지게 된다.

## 발명의 내용

[0004]

본 발명의 실시예는 데이터 저장 방법 및 장치를 제공하며, 데이터 쿼리 효율을 향상시킬 수 있다.

[0005]

제1 측면은 데이터 저장 방법을 제공하며, 키값 유형 분산형 데이터베이스(KeyValue type distributed database)에 저장될 적어도 두 개의 데이터 객체의 동일한 컬럼 속성을 결정하는 단계; 상기 키값 유형 클러스터 테이블 데이터베이스에서, 상기 적어도 두 개의 데이터 객체에서 각 데이터 객체의 로우(row) 식별자의 포맷을 결정하는 단계 - 상기 로우 식별자의 포맷이 상기 동일한 컬럼 속성과 데이터 객체 식별자를 포함하고, 상기 적어도 두 개의 데이터 객체에서 상이한 데이터 객체의 데이터 객체 식별자가 상기 로우 식별자의 포맷에서 동일한 위치에 있으며 상기 동일한 컬럼 속성 뒤에 위치함 -; 각 데이터 객체에 대해 결정된 상기 로우 식별자의 포맷에 따라서 각 데이터 객체의 각 데이터 기록의 로우 식별자 값을 결정하는 단계; 및 각 데이터 기록의 상기 로우 식별자 값과 각 데이터 기록을 상기 키값 유형 분산형 데이터베이스에 저장하는 단계를 포함하며, 각 데이터 기록의 상기 로우 식별자 값이, 상기 키값 유형 분산형 데이터베이스에서, 각 데이터 기록의 1차 인덱스(primary index)로서 사용된다.

[0006]

제1 가능한 구현예에서, 본 방법은, 상기 적어도 두 개의 데이터 객체의 데이터 기록 모두가 클러스터 테이블(cluster table)을 형성하도록, 상기 데이터 기록 모두의 로우 식별자 값에 대해, 제1 순서로 상기 데이터 기록 모두를 배열하는 단계를 더 포함한다.

[0007]

제1 측면의 제1 가능한 구현예를 참조한, 제2 가능한 구현예에서, 상기 제1 순서는 사전식 순서(lexicographic order)이다.

- [0008] 제1 측면 또는 제1 측면의 제1 또는 제2 가능한 구현예를 참조한, 제3 가능한 구현예에서, 본 방법은, 상기 적어도 두 개의 데이터 객체에서 데이터 기록을 쿼리(query)하거나 또는 획득하도록 지시하는 데이터 요청을 수신하는 단계; 상기 로우 식별자의 포맷에 따라서, 상기 키값 유형 분산형 데이터베이스를 쿼리하기 위한, 상기 동일한 컬럼 속성을 포함하는 쿼리 정보를 구성하는 단계; 및 상기 쿼리 정보에 따라서, 1차 인덱스(primary index)가 상기 쿼리 정보를 포함하는 데이터 기록을 상기 키값 유형 분산형 데이터베이스로부터 내보내는 (export) 단계를 더 포함한다.
- [0009] 제1 측면 또는 제1 측면의 제1 내지 제3 가능한 구현예 중 어느 하나의 가능한 구현예를 참조한, 제4 가능한 구현예에서, 상기 키값 유형 분산형 데이터베이스에서, 상기 적어도 두 개의 데이터 객체에서 각 데이터 객체의 로우 식별자의 포맷을 결정하는 단계가, 상기 적어도 두 개의 데이터 객체에 대해 상기 키값 유형 분산형 데이터베이스에서, 상기 로우 식별자의 제1 프리픽스(prefix)를 특정하는 단계 - 상기 제1 프리픽스는 상수임 -; 및 상기 적어도 두 개의 데이터 객체 중 제1 데이터 객체의 로우 식별자의 포맷이 상기 제1 프리픽스, 상기 동일한 컬럼 속성, 및 상기 제1 데이터 객체의 데이터 객체 식별자를 포함하는 것으로 결정하거나, 또는, 상기 적어도 두 개의 데이터 객체 중 제2 데이터 객체의 로우 식별자의 포맷이 상기 제1 프리픽스, 상기 동일한 컬럼 속성, 상기 제2 데이터 객체의 데이터 객체 식별자, 및 상기 제2 데이터 객체의 컬럼 속성에 있으면서 상기 제2 데이터 객체의 데이터 기록 모두를 구별할 수 있는 컬럼 속성을 포함하 것으로 결정하는 단계를 포함한다.
- [0010] 제1 측면의 제4 가능한 구현예를 참조한, 제5 가능한 구현예에서, 상기 쿼리 정보가 상기 제1 프리픽스와 상기 동일한 컬럼 속성을 포함한다.
- [0011] 제4 측면 또는 제1 측면의 제5 가능한 구현예를 참조한, 제6 가능한 구현예에서, 본 방법은, 상기 키값 유형 분산형 데이터베이스에서, 상기 적어도 두 개의 데이터 객체 중 임의의 데이터 객체의 2차 인덱스(secondary index)의 포맷이 제2 프리픽스, 제2 속성, 및 상기 동일한 컬럼 속성인 것으로 결정하는 단계 - 상기 제2 속성은 상기 임의의 데이터 객체의 컬럼 속성에 있으면서 상기 동일한 속성과 상이한 컬럼 속성이고, 상기 제2 프리픽스는 상기 제1 프리픽스와 상이한 상수임 -; 상기 임의의 데이터 객체의 상기 2차 인덱스의 포맷에 따라서 상기 임의의 데이터 객체의 상기 2차 인덱스를 결정하는 단계; 및 상기 키값 유형 분산형 데이터베이스에 상기 임의의 데이터 객체의 상기 2차 인덱스를 저장하는 단계를 더 포함한다.
- [0012] 제1 측면 또는 제1 측면의 제1 내지 제6 가능한 구현예 중 어느 하나의 가능한 구현예를 참조한, 제7 가능한 구현예에서, 상기 로우 식별자의 포맷 내의 필드(field)들 사이에 분리자(separator)가 설정되거나, 또는 상기 필드들이 고정 길이를 갖는다.
- [0013] 제2 측면은 데이터 저장 장치를 제공하며, 상기 키값 유형 분산형 데이터베이스(KeyValue type distributed database)에 저장될 적어도 두 개의 데이터 객체의 동일한 컬럼 속성을 결정하고; 상기 키값 유형 분산형 데이터베이스에서, 상기 적어도 두 개의 데이터 객체에서 각 데이터 객체의 로우(row) 식별자의 포맷을 결정하며; 각 데이터 객체에 대해 결정된 상기 로우 식별자의 포맷에 따라서 각 데이터 객체의 각 데이터 기록의 로우 식별자 값을 결정하도록 구성되는 결정 모듈 - 상기 로우 식별자의 포맷이 상기 동일한 컬럼 속성과 데이터 객체 식별자를 포함하고, 상기 적어도 두 개의 데이터 객체에서 상이한 데이터 객체의 데이터 객체 식별자가 상기 로우 식별자의 포맷에서 동일한 위치에 있으며 상기 동일한 컬럼 속성 뒤에 위치함 -; 및 각 데이터 기록의 상기 로우 식별자 값과 각 데이터 기록을 상기 키값 유형 분산형 데이터베이스에 저장하도록 구성되는 저장 모듈을 포함하며, 각 데이터 기록의 상기 로우 식별자 값이 상기 키값 유형 분산형 데이터베이스에서 각 데이터 기록의 1차 인덱스(primary index)로서 사용된다.
- [0014] 제1 가능한 구현예에서, 상기 저장 모듈이 상기 적어도 두 개의 데이터 객체의 데이터 기록 모두가 클러스터 테이블(cluster table)을 형성하도록, 상기 데이터 기록 모두의 로우 식별자 값에 대해, 제1 순서로 상기 데이터 기록 모두를 배열하도록 추가로 구성된다.
- [0015] 제2 측면의 제1 가능한 구현예를 참조한, 제2 가능한 구현예에서, 상기 제1 순서가 사전식 순서(lexicographic order)이다.
- [0016] 제2 측면 또는 제2 측면의 제1 또는 제2 가능한 구현예를 참조한, 제3 가능한 구현예에서, 상기 적어도 두 개의 데이터 객체에서 데이터 기록을 쿼리(query)하거나 또는 획득하도록 지시하는 데이터 요청을 수신하도록 구성되는 수신 모듈; 및 상기 로우 식별자의 포맷에 따라서, 상기 키값 유형 분산형 데이터베이스를 쿼리하기 위한, 상기 동일한 컬럼 속성을 포함하는 쿼리 정보를 구성하고, 상기 쿼리 정보에 따라서, 1차 인덱스(primary index)가 상기 쿼리 정보를 포함하는 데이터 기록을 상기 키값 유형 분산형 데이터베이스로부터 내보내는

(export)도록 구성되는 쿼리 모듈을 더 포함한다.

[0017] 제2 측면 또는 제2 측면의 제1 내지 제3 가능한 구현예 중 어느 하나의 가능한 구현예를 참조한 제4 가능한 구현예에서, 상기 결정 모듈이, 구체적으로, 상기 키값 유형 분산형 데이터베이스에서, 상기 적어도 두 개의 데이터 객체에 대해, 상기 로우 식별자의 제1 프리픽스(prefix)를 특정하고, 상기 적어도 두 개의 데이터 객체 중 제1 데이터 객체의 로우 식별자의 포맷이 상기 제1 프리픽스, 상기 동일한 컬럼 속성, 및 상기 제1 데이터 객체의 데이터 객체 식별자를 포함하는 것으로 결정하거나, 또는, 상기 적어도 두 개의 데이터 객체 중 제2 데이터 객체의 로우 식별자의 포맷이 상기 제1 프리픽스, 상기 동일한 컬럼 속성, 상기 제2 데이터 객체의 데이터 객체 식별자, 및 상기 제2 데이터 객체의 컬럼 속성에 있으면서 상기 제2 데이터 객체의 데이터 기록 모두를 구별할 수 있는 컬럼 속성을 포함하는 것으로 결정하도록 구성되고, 상기 제1 프리픽스는 상수이다.

[0018] 제2 측면의 제5 가능한 구현예를 참조한, 제5 가능한 구현예에서, 상기 쿼리 정보가 상기 제1 프리픽스와 상기 동일한 컬럼 속성을 포함한다.

[0019] 제4 측면 또는 제2 측면의 제5 가능한 구현예를 참조한, 제6 가능한 구현예에서, 상기 결정 모듈이, 상기 키값 유형 분산형 데이터베이스에서, 상기 적어도 두 개의 데이터 객체 중 임의의 데이터 객체의 2차 인덱스(secondary index)의 포맷이 제2 프리픽스, 제2 속성, 및 상기 동일한 컬럼 속성인 것으로 결정하고, 상기 임의의 데이터 객체의 상기 2차 인덱스의 포맷에 따라서 상기 임의의 데이터 객체의 상기 2차 인덱스를 결정하도록 추가로 구성되며, 상기 저장 모듈이 상기 키값 유형 분산형 데이터베이스에서 상기 임의의 데이터 객체의 상기 2차 인덱스를 저장하도록 추가로 구성되고, 상기 제2 속성은 상기 임의의 데이터 객체의 컬럼 속성에 있으면서 상기 동일한 속성과 상이한 컬럼 속성이고, 상기 제2 프리픽스는 상기 제1 프리픽스와 상이한 상수이다.

[0020] 제2 측면 또는 제2 측면의 제1 내지 제6 가능한 구현예 중 어느 하나의 가능한 구현예를 참조한 제7 가능한 구현예에서, 상기 로우 식별자의 포맷 내의 필드(field)들 사이에 분리자(separator)가 설정되거나, 또는 상기 필드들이 고정 길이를 갖는다.

[0021] 전술한 기술적 해결수단에 기초하여, 본 발명의 실시예에 따른 데이터 저장 방법 및 장치에서, 상관된 데이터 객체가 동일한 컬럼 속성을 포함하는 로우 식별자 포맷을 사용하여 키값 유형 분산형 데이터베이스에 저장되어, 상관된 쿼리가 복수의 데이터 객체에 대해 구현될 수 있으므로, 데이터 쿼리 효율이 향상될 수 있다.

### 도면의 간단한 설명

[0022] 본 발명의 실시예에서의 기술적 해결수단을 더욱 명확히 기술하기 위해, 이하에서 본 발명의 실시예를 설명할 때 필요한 첨부 도면을 간략하게 소개한다. 분명한 것은, 이어질 설명에서 첨부된 도면은 단지 본 발명의 몇 가지 실시예를 나타내며, 통상의 기술자라면 첨부된 도면으로부터 창작 능력 없이도 다른 도면을 도출해 낼 수 있다는 것이다.

도 1은 본 발명의 실시예에 따른 데이터 저장 방법의 개략적인 흐름도이다.

도 2는 본 발명의 실시예에 따라 로우 식별자 포맷을 결정하기 위한 방법의 개략적인 흐름도이다.

도 3은 본 발명의 다른 실시예에 따른 데이터 저장 방법의 개략적인 흐름도이다.

도 4는 본 발명의 또 다른 실시예에 따른 데이터 저장 방법의 개략적인 흐름도이다.

도 5는 본 발명의 실시예에 따른 데이터 저장 장치의 개략적인 블록 구성도이다.

도 6은 본 발명의 다른 실시예에 따른 데이터 저장 장치의 개략적인 블록 구성도이다.

도 7은 본 발명의 실시예에 따른 데이터 저장 장치의 개략적인 구조도이다.

### 발명을 실시하기 위한 구체적인 내용

[0023] 이하에서 본 발명의 실시예에 관한 첨부 도면을 참조하여 본 발명의 실시예의 기술적 해결 수단을 명확하고 상세하게 기술한다. 분명한 것은, 여기 기술된 실시예는 단지 본 발명의 실시예의 일부이며 전부가 아니라는 것이다. 본 발명의 실시예에 기초하여 창작 노력 없이 통상의 기술자에 의해 획득되는 다른 모든 실시예는 본 발명의 보호 범위 내에 포함될 것이다.

[0024] 본 발명의 실시예에서의 기술적 해결 수단은 키값 유형 분산형 데이터베이스(KeyValue Type distributed database)에 적용될 수 있다.

- [0025] 본 발명의 실시예에서, 데이터 객체는 데이터 테이블(또한 테이블로서 간단하게 명명함) 또는 데이터 테이블 포맷의 데이터이다. 예를 들어, 데이터는 통화 기록 데이터(call record data), 네트워크 액세스 기록 데이터 등일 수 있다. 설명의 편의를 위해, 이하의 실시예에서는, 데이터 테이블을 예로 사용하여 설명된다.
- [0026] 도 1은 본 발명의 실시예에 따른 데이터 저장 방법(100)의 개략적인 흐름도이다. 도 1에서의 방법은 데이터 저장 장치에 의해 실행된다. 도 1에 도시된 바와 같이, 본 방법(100)은 다음의 단계를 포함한다.
- [0027] 단계 S110: 키값 유형 분산형 데이터베이스에 저장될 적어도 두 개의 데이터 객체의 동일한 컬럼 속성을 결정한다.
- [0028] 단계 S120: 키값 유형 분산형 데이터베이스에서, 적어도 두 개의 데이터 객체 내에서 각 데이터 객체의 로우(row) 식별자의 포맷을 결정하며, 여기서 로우 식별자 포맷은 동일한 컬럼 속성과 데이터 객체 식별자를 포함하고, 적어도 두 개의 데이터 객체에서 상이한 데이터 객체의 데이터 객체 식별자는 로우 식별자 포맷에서 동일한 위치에 있으며 동일한 컬럼 속성 뒤에 위치된다.
- [0029] 단계 S130: 각 데이터 객체에 대해 결정된 로우 식별자 포맷에 따라서 각 데이터 객체의 각 데이터 기록(record)의 로우 식별자 값을 결정한다.
- [0030] 단계 S140: 키값 유형 분산형 데이터베이스에 각 데이터 기록의 로우 식별자 값과 각 데이터 기록을 저장하며, 여기서 각 데이터 기록의 로우 식별자 값은 키값 유형 분산형 데이터베이스에서 각 데이터 기록의 1차 인덱스(primary index)로서 사용된다.
- [0031] 본 발명의 실시예에서, 복수의(적어도 두 개의) 데이터 객체가 상관되는 경우, 즉 복수의 데이터 객체가 복수의 데이터 객체에 대해 상관된 쿼리를 가능하게 하기 위해 동일한 컬럼 속성을 가지는 경우, 데이터 저장 장치는 키값 유형 분산형 데이터베이스의 하나의 테이블에 복수의 데이터 객체를 저장한다. 데이터 저장 장치는, 동일한 컬럼 속성에 기초하여, 키값 유형 분산형 데이터베이스에서, 각 데이터 객체의 로우 식별자(예를 들어, RowKey)의 포맷을 결정하며, 여기서 로우 식별자 포맷은 동일한 컬럼 속성과 데이터 객체 식별자를 포함하고, 적어도 두 개의 데이터 객체에서 상이한 데이터 객체의 데이터 객체 식별자는 로우 식별자 포맷에서 동일한 위치에 있으며 동일한 컬럼 속성 뒤에 위치하게 된다. 즉, 데이터 객체 모두의 로우 식별자 포맷은 쿼리를 가능하게 하기 위해 동일한 컬럼 속성을 포함하며, 상이한 데이터 객체를 구별하기 위해 데이터 객체에 대응하는 데이터 객체 식별자를 더 포함한다. 상이한 데이터 객체의 로우 식별자 포맷이 일관된 구조를 가지고록, 상이한 데이터 객체의 데이터 객체 식별자가 로우 식별자 포맷에서 동일한 위치에 있으며 동일한 컬럼 속성 뒤에 위치된다. 그 후, 데이터 저장 장치는 결정된 로우 식별자 포맷에 따라서 각 데이터 객체에서 각 데이터 기록의 로우 식별자 값을 결정하고, 각 데이터 기록의 로우 식별자 값과 각 데이터 기록을 키값 유형 분산형 데이터베이스에 저장하며, 여기서 각 데이터 기록의 로우 식별자 값은 키값 유형 분산형 데이터베이스에서 각 데이터 기록의 1차 인덱스로 사용된다. 이와 같이, 일관된 로우 식별자 포맷이 복수의 상관된 데이터 객체에 대해 사용되기 때문에, 데이터 쿼리 중에, 쿼리 정보가 로우 식별자 포맷에 따라서 구성되어, 상관된 쿼리가 하나의 데이터 테이블에서 복수의 데이터 객체에 대해 구현될 수 있으므로, 데이터 쿼리 효율을 향상시킬 수가 있다.
- [0032] 따라서, 본 발명의 실시예에 따른 데이터 저장 방법에서, 상관된 데이터 객체가 동일한 컬럼 속성을 포함하는 로우 식별자 포맷을 사용하여 키값 유형 분산형 데이터베이스에 저장되어, 상관된 쿼리가 복수의 데이터 객체에 대해 구현될 수 있으므로, 데이터 쿼리 효율을 향상시킬 수 있다.
- [0033] 본 발명의 실시예에서, 로우 식별자가 데이터의 각 로우의 식별자라는 것이 이해되어져야 한다. 로우 식별자는 RowKey 또는 Key로서 표현될 수 있으며, 이는 본 발명에서 한정되지 않는다. 설명의 편의를 위해, 이하의 실시예에서, RowKey를 예로 사용하여 설명된다.
- [0034] 이하, 구체적인 예를 사용하여 본 발명의 실시예가 설명된다. 이러한 예는 단순히 통상의 기술자가 본 발명의 실시예를 더 잘 이해할 수 있도록 하고자 하는 것으로, 본 발명의 실시예의 범위를 한정하고자 하는 것이 아니다.
- [0035] 단계 S110에서, 데이터 저장 장치가 키값 유형 분산형 데이터베이스에 저장될 적어도 두 개의 데이터 객체의 동일한 컬럼 속성을 결정한다.
- [0036] 복수의 데이터 객체가 동일한 컬럼 속성을 갖는다는 것은 복수의 데이터 객체가 상관된다는 것을 의미한다.
- [0037] 예를 들어, 학생 정보를 기록하고 있는 테이블인 [표 1]이 예로서 사용되는 경우, 데이터 객체 A에 대해, 속성 정보가 이름(Name), 학생 번호(ID), 성별(Sex), 및 반(department, Dept)과 같은 컬럼(Column) 속성을 포함한

다.

[0038]

[표 1]

RowKey	Columns				
ID	Name	ID	Sex	Dept	...
0001	Zhang San	0001	Male	Class 3	...
0002	Li Si	0002	Male	Class 3	...
0003	Wang Wu	0003	Female	Class 3	...

[0039]

[0040] 학생 교과-변동(curricula-variable) 정보를 기록하고 있는 테이블인 [표 2]가 예로서 사용되는 경우, 데이터 객체 B에 대해, 속성 정보가 일련번호(No), 학생 번호(ID), 및 코스(course)와 같은 컬럼(Column) 속성을 포함한다.

[0041]

[표 2]

RowKey	Columns				
ID + No	No	ID	Course	...	
00011	1	0001	English	...	
00012	2	0001	Chinese	...	
00013	3	0001	Math	...	
00024	4	0002	English		
00025	5	0002	Math		
00036	6	0003	English		

[0042]

[0043] [표 1]과 [표 2] 둘 다 ID 컬럼을 가지고 있다. 즉, [표 1]과 [표 2]는 동일한 컬럼 속성 ID를 가지고 있으며; 따라서, [표 1]과 [표 2]는 상관된 데이터 객체이다.

[0044]

단계 S120에서, 데이터 저장 장치가 키값 유형 분산형 데이터베이스에서, 적어도 두 개의 데이터 객체에서 각 데이터 객체의 로우 식별자의 포맷을 결정한다.

[0045]

로우 식별자 포맷은 동일한 컬럼 속성과 데이터 객체 식별자를 포함하고, 적어도 두 개의 데이터 객체에서 상이한 데이터 객체의 데이터 객체 식별자는 로우 식별자 포맷에서 동일한 위치에 있으며 동일한 컬럼 속성 뒤에 위치된다.

[0046]

구체적으로, 본 발명의 실시예에서, 데이터 쿼리 효율을 향상시키기 위해, 각 데이터 객체가 독립된 테이블에 저장되지 않지만, 대신에, 복수의 데이터 객체가 통일된 포맷에 따라서 하나의 테이블에 저장되므로, 상관된 쿼리를 구현할 수 있다. 복수의 데이터 객체를 하나의 테이블에 저장하기 위해, 로우 식별자 포맷이 미리 설정되어야 하고, RowKey 포맷이 미리 설계되는 것이 예로서 사용된다. 즉, RowKey가 구성 요소들(composition elements)을 포함하고, 구성 요소들이 어떻게 결합되는지가 미리 규정된다. 본 실시예에서, 로우 식별자의 구성 요소는, 적어도 두 개의 객체의 동일한 컬럼 속성과 데이터 객체 식별자를 포함한다. 동일한 컬럼 속성은 적어도 두 개의 데이터 객체를 상관 방식으로 저장하는 데 사용되고, 데이터 객체 식별자는 데이터의 각 로우를 제공하는 데이터 객체를 지시하는 데 사용된다. 데이터 객체 식별자는 동일한 컬럼 속성 뒤에 위치되고, 데이터 객체 모두의 데이터 객체 식별자는 로우 식별자 포맷에서 동일한 위치에 있게 된다. 구체적으로, 상이한 데이터 객체의 데이터 객체 식별자는 상이한 데이터 객체를 구별하기 위해 상이한 상수일 수 있다.

[0047]

도 2에 도시된 바와 같이, 본 발명의 실시예에서, 선택적으로, 단계 S120은 다음의 단계를 포함한다.

[0048]

단계 S121: 적어도 두 개의 데이터 객체에 대해 키값 유형 분산형 데이터베이스에서, 상수인 로우 식별자의 제1 프리픽스(prefix)를 특정한다.

[0049]

단계 S122: 적어도 두 개의 데이터 객체에서 제1 데이터 객체의 로우 식별자 포맷이 제1 프리픽스, 동일한 컬럼 속성, 및 제1 데이터 객체의 데이터 객체 식별자를 포함하는 것으로 결정한다. 이와 달리,

- [0050] 단계 S123: 적어도 두 개의 데이터 객체에서 제2 데이터 객체의 로우 식별자 포맷이 제1 프리픽스, 동일한 컬럼 속성, 제2 데이터 객체의 데이터 식별자, 및 제2 데이터 객체의 컬럼 속성에 있으면서 제2 데이터 객체의 데이터 기록 모두를 구별할 수 있는 컬럼 속성을 포함하는 것으로 결정한다.
- [0051] 구체적으로, 본 실시예에서, 로우 식별자는 동일한 프리픽스(제1 프리픽스로 표현됨)를 포함한다. RowKey를 예로 사용하면, RowKey는 다음의 포맷에 해당된다:
- [0052]  $\text{RowKey} = \text{제1 프리픽스} + \text{동일한 컬럼 속성} + \text{데이터 객체 식별자 (+ 기타 속성)}$
- [0053] 본 실시예에서, 상이한 데이터 객체에 대해, RowKey 포맷은 동일한 제1 프리픽스와 동일한 속성을 포함한다. 이와 같이, 데이터 기록이 RowKey 값의 순서로 저장되는 경우, 상이한 데이터 객체의 데이터 기록은 인접해서 저장될 수 있다(클러스터(cluster) 저장으로서 명명됨). 따라서, 상이한 데이터 객체들은 각 데이터 객체의 데이터 객체 식별자를 사용하여 구별된다. 또한, 동일한 컬럼 속성이 하나의 데이터 객체(제1 데이터 객체로서 표현됨)의 데이터 기록 모두를 구별할 수 있는 경우, 예를 들어, [표 1]에서의 ID 속성이 [표 1]의 데이터 기록 모두를 구별할 수 있는 경우, 제1 데이터 객체의 RowKey 포맷은 다른 속성을 포함하지 않는다. 동일한 속성이 하나의 데이터 객체(제2 데이터 객체로 표현됨)의 데이터 기록 모두를 구별할 수 없는 경우, 예를 들어, [표 2]에서의 ID 속성이 [표 2]의 데이터 기록 모두를 구별할 수 없는 경우, 제2 데이터 객체의 RowKey 포맷은, 예를 들어 [표 2]에서의 No 속성과 같이, 제2 데이터 객체의 다른 컬럼 속성에 있으면서 제2 데이터 객체의 데이터 기록 모두를 구별할 수 있는 속성을 포함해야 한다.
- [0054] [표 1]과 [표 2]를 예로서 사용하면, 동일한 컬럼 속성은 ID이다. [표 1]의 경우, ID는 [표 1]의 데이터 기록 모두를 구별할 수 있으며; 따라서, [표 1]의 RowKey 포맷은 다음과 같을 수 있다.
- [0055]  $A(\text{제1 프리픽스}) + ID + M(M은 [표 1]에 대응하는 데이터 객체 식별자임})$
- [0056] [표 2]의 경우, ID는 [표 2]의 데이터 기록 모두를 구별할 수 없고, No가 [표 2]의 데이터 기록 모두를 구별할 수 있으며; 따라서, [표 2]의 RowKey 포맷은 다음과 같을 수 있다.
- [0057]  $A(\text{제1 프리픽스}) + ID + N(N은 [표 2]에 대응하는 데이터 객체 식별자임}) + No$
- [0058] 본 발명의 실시예에서, 선택적으로, 로우 식별자 포맷 내의 필드(field)들 사이에 분리자(separator)가 설정되거나, 또는 필드들이 고정 길이를 갖는다.
- [0059] 즉, 전술한 RowKey 포맷에서 "+"가 분리자일 수 있다. 이와 달리, RowKey 포맷에서 필드들이 고정 길이를 갖도록 설정될 수 있으며; 이 경우, "+"는 실제로 존재하지 않는다.
- [0060] 단계 S130에서, 각 데이터 객체에 대해 결정된 로우 식별자 포맷에 따라서 각 데이터 객체의 각 데이터 기록의 로우 식별자 값을 결정한다.
- [0061] 구체적으로, RowKey를 예로 사용하면, 데이터 저장 장치가 결정된 RowKey 포맷에 따라서 각 데이터 객체에서 각 데이터 기록의 RowKey 값을 결정한다.
- [0062] 예를 들어, 데이터 객체 A에서 제1 데이터 기록의 경우, RowKey 값은 A0001M이고; 데이터 객체 B에서 제1 데이터 기록의 경우, RowKey 값은 A0001N1이며, 이하 마찬가지로 적용된다.
- [0063] 단계 S140에서, 데이터 저장 장치가 각 데이터 기록의 로우 식별자 값과 각 데이터 기록을 키값 유형 분산형 데이터베이스에 저장한다.
- [0064] 각 데이터 기록의 로우 식별자 값이, 키값 유형 분산형 데이터베이스에서, 각 데이터 기록의 1차 인덱스로 사용된다.
- [0065] 본 발명의 실시예에서, 선택적으로, 도 3에 도시된 바와 같이, 본 방법(100)은 다음의 단계를 더 포함한다.
- [0066] 단계 S150: 적어도 두 개의 데이터 객체의 데이터 기록 모두가 클러스터 테이블을 형성하도록, 데이터 기록 모두의 로우 식별자 값에 대해, 제1 순서로 데이터 기록 모두를 배열한다.
- [0067] 본 발명의 실시예에서, 복수의 데이터 객체의 데이터 기록이 클러스터 테이블을 형성하도록, 데이터 기록이 로우 식별자 값의 순서로 키값 유형 분산형 데이터베이스에 저장된다. 이러한 클러스터-테이블 저장 방식에서, 로우 식별자 값의 범위는 파티션(partition)에 대응할 수 있다. 즉, 데이터 기록의 로우 식별자 값이 범위에 속하는 경우, 그 데이터 기록은 이 범위에 해당하는 파티션에 있으며; 따라서, 범위에 따른 쿼리가 지원될 수 있다.

[0068] 제1 순서는 사전식 순서(lexicographic order)일 수 있으며, 예를 들어, 데이터 기록 모두가 RowKey 값의 사전식 순서로 키값 유형 분산형 데이터베이스에서 배열될 수 있다. 제1 순서는 또한 다른 순서, 예를 들어, 크기 순서일 수 있다. 즉, 데이터 기록이 RowKey 값의 크기 순서로 순차적으로 저장될 수 있으며, 여기서 작은 RowKey값이 큰 RowKey 값 이전에 저장되거나, 또는 큰 RowKey 값이 작은 RowKey 값 이전에 저장된다. 제1 순서는 본 발명의 실시예에서 한정되지 않는다. 설명의 편의를 위해, 이하의 실시예에서는, RowKey 값의 사전식 순서를 예로 사용하여 설명된다.

[0069] [표 1]과 [표 1]를 예로 사용하는 경우, RowKey 값의 사전식 순서로 키값 유형 분산형 데이터베이스에서 저장이 수행되는 경우 테이블 [표 3]이 얻어진다.

[0070] [표 3]

RowKey	Columns					
	Name	ID	Sex	Dept	No	Course
RowKey = A + ID + ... A0001M	Zhang San	0001	Male	Class 3		
A0001N1		0001			1	English
A0001N2		0001			2	Chinese
A0001N3		0001			3	Math
A0002M	Li Si	0002	Male	Class 3		
A0002N4		0002			4	English
A0002N5		0002			5	Math
A0003M	Wang Wu	0003	Female	Class 3		
A0003N6		0003			6	English

[0071] [0072] (대각선을 포함하는 컬럼은 현재의 로우 데이터가 컬럼을 갖고 있지 않는 것을 지시하고, 이것은 저장시 어떠한 공간도 차지하지 않으며, 이하 동일하다)

[0073] [0074] 본 발명의 실시예에 따른 데이터 저장 방법을 사용함으로써, 상이한 상관 데이터 객체의 데이터 기록이 인접하여 저장됨으로써 클러스터 테이블을 형성할 수 있다. 예를 들어, [표 3]에서 첫 번째 기록 내지 네 번째 기록은 원래의 [표 2]에서 첫 번째 기록 내지 세 번째 기록이고, 이들 데이터 기록은 그들이 동일한 ID를 가지기 때문에 키값 유형 분산형 데이터베이스에서 인접하여 저장된다. 인접 저장에 의해, 데이터 액세스 속도가 향상될 수 있고, 쿼리 효율이 더욱 향상될 수 있다.

[0075] [0076] 이하, 전술한 저장 방법에 따라서 획득되는 데이터 테이블을 쿼리하는 구현 방식에 대해 설명한다.

본 발명의 실시예에서, 선택적으로, 도 4에 도시된 바와 같이, 본 방법(100)은 다음의 단계를 더 포함한다.

단계 S160: 적어도 두 개의 데이터 객체에서 데이터 기록을 쿼리하거나 또는 획득하도록 지시하는 데이터 요청을 수신한다.

단계 S170: 로우 식별자 포맷에 따라서, 키값 유형 분산형 데이터베이스를 쿼리하기 위한 쿼리 정보를 구성하며, 여기서 쿼리 정보는 동일한 컬럼 속성을 포함한다.

단계 S180: 쿼리 정보에 따라서, 1차 인덱스가 쿼리 정보를 포함하는 데이터 기록을 키값 유형 분산형 데이터베이스로부터 내보낸다(expert).

[0077] [0078] 구체적으로, 데이터 기록을 쿼리하거나 또는 획득하기 위한 데이터 요청이 수신되는 경우, 데이터 저장 장치가 로우 식별자 포맷에 따라서, 동일한 컬럼 속성을 포함해야 하는 쿼리 정보를 구성하며, 그 후 데이터 저장 장치가 구성된 쿼리 정보를 사용하여 쿼리를 수행하여, 1차 인덱스가 키값 유형 분산형 데이터베이스에서 쿼리 정보를 포함하는 데이터 기록을 획득한다.

[0079] 선택적으로, 로우 식별자 포맷이 제1 프리픽스를 포함하는 경우, 쿼리 정보가 제1 프리픽스 및 동일한 컬럼 속성을 포함한다.

[0081] 예를 들어, 학생 정보와 학생 교과-변동 정보가 학생의 학생 번호(ID-0001)를 사용하여 획득되는 경우, 쿼리 정보: A0001이 구성될 수 있으며, 학생 번호(ID=0001)를 갖는 학생의 학생 정보와 학생 교과-변동 정보가 범위 쿼리에 의해 [표 3]에서 획득될 수 있다. 획득된 쿼리 결과가 테이블 [표 4]에 도시되어 있다. 쿼리 과정에서, 하나의 테이블만이 쿼리되어야 하고, 사용자 데이터 로케이팅(locating)이 한번에 완성되어야 한다. 학생 정보와 학생 교과-변동 정보가 인접하여 저장되기 때문에 고속의 리턴(return)이 구현될 수 있다.

[0082] [표 4]

A0001M	Zhang San	0001	Male	Class 3		
A0001N1		0001			1	English
A0001N2		0001			2	Chinese
A0001N3		0001			3	Math

[0083]

[0084] 따라서, 본 발명의 실시예에 따른 데이터 저장 방법에서, 상관 데이터 객체가 하나의 데이터 테이블에서 클래스터로서 저장될 수 있고, 상관 쿼리가 하나의 데이터 테이블에서 구현될 수 있도록, 상관 데이터 객체가 동일한 컬럼 속성을 포함하는 로우 식별자 포맷을 사용하여 키값 유형 분산형 데이터베이스에 저장됨으로써, 데이터 쿼리 효율이 향상될 수 있다.

[0085] 본 발명의 실시예에서, 선택적으로, 본 방법(100)이 다음의 단계를 더 포함한다.

[0086] 키값 유형 분산형 데이터베이스에서, 적어도 두 개의 데이터 객체 중 임의의 데이터 객체의 2차 인덱스(secondary index)의 포맷이 제2 프리픽스, 제2 속성, 및 동일한 컬럼 속성인 것으로 결정하며, 여기서 제2 속성은 임의의 데이터 객체의 컬럼 속성에 있는 컬럼 속성이고 동일한 속성과 상이하며, 제2 프리픽스는 제1 프리픽스와 상이한 상수이다.

[0087] 임의의 데이터 객체의 2차 인덱스의 포맷에 따라서 임의의 데이터 객체의 2차 인덱스를 결정한다.

[0088] 임의의 데이터 객체의 2차 인덱스를 키값 유형 분산형 데이터베이스에 저장한다. 구체적으로, 복잡한 쿼리의 시나리오를 만족시키기 위해, 2차-인덱스 정보가 데이터 객체를 위해 구성될 수 있다. 1차 인덱스의 데이터와 동일한 데이터 객체의 2차 인덱스가 동일한 테이블에 저장되지만, 상이한 로우 식별자 포맷이 사용되어, 동일한 테이블에서 데이터 분리를 구현할 수 있다. 또한, 상관된 쿼리가 2차-인덱스 정보를 사용하여 완성될 수 있다. 2차 인덱스의 로우 식별자 포맷은 제2 프리픽스, 제2 속성, 및 동일한 속성이다. 제2 프리픽스는 데이터 분리를 구현하기 위해 제1 프리픽스와 상이하고; 제2 속성은 제2 속성에 따라서 쿼리 정보의 구축이 가능하도록 동일한 속성과 상이한 속성이며; 동일한 속성은 1차 인덱스와 연관되어 사용된다.

[0089] 예를 들어, 데이터 객체 [표 1]과 데이터 객체 [표 2]가 키값 유형 분산형 데이터베이스에 저장되는 경우, [표 1]의 1차-인덱스 로우키 포맷은 A + ID + M일 수 있고; [표 2]의 1차-인덱스 로우키 포맷은 A + ID + N + No일 수 있으며; [표 1]의 2차-인덱스 로우키 포맷은 B + Name + ID일 수 있다. 이렇게 하여, 로우키 값의 사전식 순서로 키값 유형 분산형 데이터베이스에서 저장이 수행되는 경우 테이블 [표 5]가 얻어진다.

[0090]

[표 5]

RowKey	Columns					
RowKey = A + ID + ...	Name	ID	Sex	Dept	No	Course
A0001M	Zhang San	0001	Male	Class 3		
A0001N1		0001			1	English
A0001N2		0001			2	Chinese
A0001N3		0001			3	Math
A0002M	Li Si	0002	Male	Class 3		
A0002N4		0002			4	English
A0002N5		0002			5	Math
A0003M	Wang Wu	0003	Female	Class 3		
A0003N6		0003			6	English

(이하의 데이터는 다른 데이터 파티션에 속하고, 이 때의 로우(row)는 상기한 파티션으로부터의 분리를  
지시하는데 사용되며, 실제 데이터를 갖지 않음)

BZhang San0001						
BLi Si0002						
BWang Wu0003						

[0091]

[0092] 2차-인덱스 로우키에 대응하는 컬럼에서, 특정의 중요한 컬럼 정보 조각과 같은 컬럼 정보가 또한 저장될 수 있으며, 이것은 본 발명의 실시예에서 한정되지 않는다.

[0093]

학생 정보와 학생 교과-변동 정보가 학생 이름(Name = Zhang San)을 사용하여 획득되는 경우, 쿼리 정보: BZhang San이 구축될 수 있으며, 완전한 2차-인덱스 로우키 정보: Bzhang San0001이 범위 쿼리에 의해 [표 5]에서 획득될 수 있다. 2차-인덱스 로우키가 형성되어 ID = 0001이 획득된다. 그 후, 쿼리 정보: A0001가 추가로 구성되고, 학생 정보와 학생 교과-변동 정보가 추가 쿼리를 수행하여 [표 5]에서 획득될 수 있다. 이로 인해 사용자가 이름만 알고 ID를 모르는 경우에도 쿼리를 수행할 수가 있다.

[0094]

따라서, 본 발명의 실시예에 따른 데이터 저장 방법에서, 데이터 객체의 2차 인덱스가 키값 유형 분산형 데이터 베이스에 저장되어, 보다 다양한 쿼리 시나리오가 사용자에게 제공될 수 있으며, 사용자에 의한 쿼리가 가능하게 되어, 데이터 쿼리 효율을 향상시킬 수 있다.

[0095]

본 발명의 실시예에서, 복수의 데이터 객체의 로우 식별자 포맷이 또한 다른 방식으로 구현될 수 있다. 예를 들어, 상이한 데이터 객체들의 로우 식별자 포맷이 상이한 프리픽스들을 포함할 수 있거나, 또는 프리픽스를 포함하지 않을 수 있다.

[0096]

예를 들어, 프리픽스가 사용되지 않는 경우, 로우키는 다음의 포맷을 갖는다:

[0097]

RowKey = 동일한 컬럼 속성 + 데이터 객체 식별자(+ 기타 속성)

[0098]

상이한 데이터 객체의 데이터 기록은 동일한 컬럼 속성을 사용하여 인접하게 저장될 수 있으며, 상이한 데이터 객체는 각 데이터 객체에 대응하는 데이터 객체 식별자를 사용하여 구별된다. 또한, 동일한 컬럼 속성이 데이터 객체(제1 데이터 객체로 표현됨)의 데이터 기록 모두를 구별할 수 있는 경우, 제1 데이터 객체의 로우키 포맷은 다른 속성을 포함하지 않으며; 동일한 컬럼 속성이 데이터 객체(제2 데이터 객체로 표현됨)의 데이터 기록 모두를 구별할 수 없는 경우, 제2 데이터 객체의 로우키 포맷은 제2 데이터 객체의 다른 컬럼 속성이면서 제2 데이터 객체의 데이터 기록 모두를 구별할 수 있는 속성을 더 포함해야 한다.

[0099]

[표 1]과 [표 2]를 예로서 사용하면, 그들의 동일한 컬럼 속성은 ID이다. [표 1]의 경우, ID가 [표 1]의 데이터 기록 모두를 구별할 수 있으므로; 따라서, [표 1]의 로우키 포맷은

[0100]

ID + M일 수 있다.

[0101]

[표 2]의 경우, ID가 [표 2]의 데이터 기록 모두를 구별할 수 없고, No가 [표 2]의 데이터 기록 모두를 구별할

수 있으므로; 따라서, [표 2]의 로우키 포맷은

[0102] ID + N + No일 수 있다.

[0103] 이렇게 하여, 로우키 값의 사전식 순서로 키값 유형 분산형 데이터베이스에서 저장이 수행되는 경우 테이블 [표 6]이 얻어진다.

[0104] [표 6]

RowKey	Columns					
	Name	ID	Sex	Dept	No	Course
RowKey = ID + ... 0001M	Zhang San	0001	Male	Class 3		
0001N1		0001			1	English
0001N2		0001			2	Chinese
0001N3		0001			3	Math
0002M	Li Si	0002	Male	Class 3		
0002N4		0002			4	English
0002N5		0002			5	Math
0003M	Wang Wu	0003	Female	Class 3		
0003N6		0003			6	English

[0105]

[0106] 학생 정보와 학생 교과-변동 정보가 학생의 학생 번호(ID = 0001)를 사용하여 획득되는 경우, 쿼리 정보: 0001이 구축될 수 있으며, 학생 번호(ID = 0001)을 갖는 학생의 학생 정보와 학생 교과-변동 정보가 범위 쿼리에 의해 획득될 수 있다. 쿼리 과정 중에, 하나의 테이블만이 쿼리되어야 하며, 사용자 데이터 로케이팅은 단지 한 번에 완료되어야 한다. 학생 정보와 학생 교과-변동 정보가 인접하여 저장되기 때문에 빠른 리턴(return)이 구현될 수 있다.

[0107] 상이한 프리픽스가 상이한 데이터 객체에 대해 사용되는 경우, 로우키는 다음의 포맷을 갖는다:

[0108] RowKey = (데이터 객체에 대응하는 프리픽스) + 동일한 컬럼 속성 (+ 기타 속성)

[0109] 상이한 프리픽스가 상이한 데이터 객체에 대해 사용되는 경우, 상이한 데이터 객체가 상이한 파티션에 저장되도록 구현된다. 또한, 동일한 컬럼 속성이 데이터 객체(제1 데이터 객체로 표현됨)의 데이터 기록 모두를 구별할 수 있는 경우, 제1 데이터 객체의 로우키 포맷은 다른 속성을 포함하지 않으며; 동일한 컬럼 속성이 데이터 객체(제2 데이터 객체로 표현됨)의 데이터 기록 모두를 구별할 수 없는 경우, 제2 데이터 객체의 로우키 포맷은 제2 데이터 객체의 다른 컬럼 속성이면서 제2 데이터 객체의 데이터 기록 모두를 구별할 수 있는 속성을 더 포함해야 한다.

[0110] [표 1]과 [표 2]를 예로서 사용하면, 그들의 동일한 컬럼 속성은 ID이다. [표 1]의 경우, ID가 [표 1]의 데이터 기록 모두를 구별할 수 있으므로; 따라서, [표 1]의 로우키 포맷은

[0111] A + ID일 수 있다.

[0112] [표 2]의 경우, ID가 [표 2]의 데이터 기록 모두를 구별할 수 없고, No가 [표 2]의 데이터 기록 모두를 확인할 수 있으므로; 따라서, [표 2]의 로우키 포맷은

[0113] B + ID + No일 수 있다.

[0114] 이렇게 하여, 로우키 값의 사전식 순서로 키값 유형 분산형 데이터베이스에서 저장이 수행되는 경우 테이블 [표 7]이 얻어진다.

[0115]

[표 7]

RowKey	Columns					
	Name	ID	Sex	Dept	No	Course
RowKey = A (B) + ID + ...						
A0001	Zhang San	0001	Male	Class 3		
A0002	Li Si	0002	Male	Class 3		
A0003	Wang Wu	0003	Female	Class 3		

(이하의 데이터는 다른 데이터 파티션에 속하고, 이 때의 로우(row)는 상기한 데이터 파티션으로부터의 분리를 지시하는데 사용되며, 실제 데이터를 갖지 않음)

B00011		0001			1	English
B00012		0001			2	Chinese
B00013		0001			3	Math
B00024		0002			4	English
B00025		0002			5	Math
B00036		0003			6	English

[0116]

[0117] 상이한 데이터 객체가 키값 유형 분산형 데이터베이스에서 상이한 파티션에 저장될 수 있도록, 상이한 데이터 객체의 로우키 포맷이 상이한 프리픽스를 포함한다.

[0118] 전술한 과정의 일련 번호는 본 발명의 다양한 실시예에서의 실행 순서를 의미하는 것이 아니라는 점이 이해되어야 한다. 상기 과정의 실행 순서는 해당 과정의 기능과 내부 로직에 따라서 결정되어야 하고, 본 발명의 실시예의 구현 과정에 대해 어떠한 제한사항으로 작용해서도 안된다.

[0119] 본 발명의 실시예에 따른 데이터 저장 방법에서, 복수의 데이터 객체가 동일한 로우 식별자 포맷에 따라서 키값 유형 분산형 데이터 베이스 내의 하나의 데이터 테이블에 저장되므로, 데이터베이스에 대한 관리가 용이해지며, 데이터 쿼리 효율을 향상시킬 수 있다.

[0120] 본 발명의 실시예에 따른 데이터 저장 방법이 상기에서 상세하게 설명되었으며, 본 발명의 실시예에 따른 데이터 저장 장치가 하기에서 설명된다.

[0121] 도 5는 본 발명의 실시예에 따른 데이터 저장 장치(500)의 개략적인 블록 구성도이다. 도 5에 도시된 바와 같이, 장치(500)는,

[0122] 키값 유형 분산형 데이터베이스에 저장될 적어도 두 개의 데이터 객체의 동일한 컬럼 속성을 결정하고; 키값 유형 분산형 데이터베이스에서, 적어도 두 개의 데이터 객체 각각의 로우 식별자 포맷을 결정하며; 각 데이터 객체에 대해 결정된 로우 식별자 포맷에 따라서 각 데이터 객체의 각 데이터 기록의 로우 식별자 값을 결정하도록 구성되는 결정 모듈(510) - 여기서 로우 식별자 포맷은 동일한 컬럼 속성 및 데이터 객체 식별자를 포함하고, 적어도 두 개의 데이터 객체에서 상이한 데이터 객체의 데이터 객체 식별자는 로우 식별자 포맷에서 동일한 위치에 있으며 동일한 컬럼 속성 뒤에 위치함-; 및

[0123] 키값 유형 분산형 데이터베이스에 각 데이터 기록의 로우 식별자 값과 각 데이터 기록을 저장하도록 구성되는 저장 모듈(520) - 여기서 각 데이터 기록의 로우 식별자 값은 키값 유형 분산형 데이터베이스에서 각 데이터 기록의 1차 인덱스로 사용됨 -을 포함한다.

[0124] 복수의(즉, 적어도 두 개의) 데이터 객체가 상관되는 경우, 즉, 복수의 데이터 객체가 복수의 데이터 객체에 대해 상관된 쿼리를 이용할 수 있도록 동일한 컬럼 속성을 갖는 경우, 데이터 저장 장치(500)는 키값 유형 분산형 데이터베이스의 하나의 테이블에 복수의 데이터 객체를 저장한다. 결정 모듈(510)은, 동일한 컬럼 속성에 기초하여, 키값 유형 분산형 데이터베이스에서, 각 데이터 객체의 로우 식별자의 포맷(예를 들어, 로우키)을 결정하며, 여기서 로우 식별자 포맷은 동일한 컬럼 속성과 데이터 객체 식별자를 포함하고, 적어도 두 개의 데이터 객체에서 상이한 데이터 객체의 데이터 객체 식별자는 로우 식별자 포맷에서 동일한 위치에 있으며 동일한 컬럼 속성 뒤에 위치한다. 즉, 데이터 객체 모두의 로우 식별자 포맷은 쿼리를 가능하게 하기 위해 동일한 컬럼 속성을 포함하며, 상이한 데이터 객체를 구별하기 위해 데이터 객체에 대응하는 데이터 객체 식별자를 더 포함한다.

상이한 데이터 객체의 로우 식별자 포맷이 일관된 구조를 가지도록, 상이한 데이터 객체의 데이터 객체 식별자가 로우 식별자 포맷에서 동일한 위치에 있으며 동일한 컬럼 속성 뒤에 위치된다. 그 후, 결정 모듈(510)은 결정된 로우 식별자 포맷에 따라서 각 데이터 객체에서 각 데이터 기록의 로우 식별자 값을 결정한다. 저장 모듈(520)은 각 데이터 기록의 로우 식별자 값과 각 데이터 기록을 키값 유형 분산형 데이터베이스에 저장하며, 여기서 각 데이터 기록의 로우 식별자 값은 키값 유형 분산형 데이터베이스에서 각 데이터 기록의 1차 인덱스로 사용된다. 일관된 로우 식별자 포맷이 복수의 상관된 데이터 객체에 대해 사용되기 때문에, 데이터 쿼리 중에, 쿼리 정보가 로우 식별자 포맷에 따라서 구축되어, 상관된 쿼리가 하나의 데이터 테이블에서 복수의 데이터 객체에 대해 구현될 수 있으므로, 데이터 쿼리 효율을 향상시킬 수가 있다.

[0125] 따라서, 본 발명의 실시예에 따른 데이터 저장 장치에서, 상관된 데이터 객체가 동일한 컬럼 속성을 포함하는 로우 식별자 포맷을 사용하여 키값 유형 분산형 데이터베이스에 저장되어, 상관된 쿼리가 복수의 데이터 객체에 대해 구현될 수 있으므로, 데이터 쿼리 효율을 향상시킬 수 있다.

[0126] 본 발명의 실시예에서, 선택적으로, 적어도 두 개의 데이터 객체의 데이터 기록 모두가 클러스터 테이블을 형성하도록, 저장 모듈(520)이 데이터 기록 모두의 로우 식별자에 대해, 제1 순서로 데이터 기록 모두를 배열하도록 추가로 구성된다.

[0127] 본 발명의 실시예에서, 선택적으로, 제1 순서는 사전식 순서이다.

[0128] 본 발명의 실시예에서, 선택적으로, 결정 모듈(510)은 적어도 두 개의 데이터 객체에 대해 키값 유형 분산형 데이터베이스에서, 상수인 로우 식별자의 제1 프리픽스를 특정하고; 적어도 두 개의 데이터 객체에서 제1 데이터 객체의 로우 식별자 포맷이 제1 프리픽스, 동일한 컬럼 속성, 및 제1 데이터 객체의 데이터 객체 식별자를 포함하는 것으로 결정하거나; 또는, 적어도 두 개의 데이터 객체에서 제2 데이터 객체의 로우 식별자 포맷이 제1 프리픽스, 동일한 컬럼 속성, 제2 데이터 객체의 데이터 객체 식별자, 및 제2 데이터 객체의 컬럼 속성에 있으면서 제2 데이터 객체의 데이터 기록 모두를 구별할 수 있는 컬럼 속성을 포함하는 것으로 결정하도록 구체적으로 구성된다.

[0129] 본 발명의 실시예에서, 선택적으로, 도 6에 도시된 바와 같이, 장치(500)는,

[0130] 적어도 두 개의 객체에서 데이터 기록을 쿼리하거나 또는 획득하도록 지시하는 데이터 요청을 수신하도록 구성되는 수신 모듈(530); 및

[0131] 로우 식별자 포맷에 따라서, 키값 유형 분산형 데이터베이스를 쿼리하기 위한 쿼리 정보를 구축하고; 쿼리 정보에 따라서, 1차 인덱스가 쿼리 정보를 포함하는 데이터 기록을 키값 유형 분산형 데이터베이스로부터 내보내도록 구성되는 쿼리 모듈(540) - 여기서 쿼리 정보는 동일한 컬럼 속성을 포함함 -을 더 포함한다.

[0132] 선택적으로, 로우 식별자 포맷이 제1 프리픽스를 포함하는 경우, 쿼리 정보는 제1 프리픽스 및 동일한 컬럼 속성을 포함한다.

[0133] 본 발명의 실시예에서, 선택적으로, 결정 모듈(510)은 키값 유형 분산형 데이터베이스에서, 적어도 두 개의 데이터 객체 중 임의의 데이터 객체의 2차 인덱스의 포맷이 제2 프리픽스, 제2 속성, 및 동일한 컬럼 속성인 것으로 결정하고; 임의의 데이터 객체의 2차 인덱스의 포맷에 따라서 임의의 데이터 객체의 2차 인덱스를 결정하도록 추가로 구성되며, 여기서 제2 속성은 임의의 데이터 객체의 컬럼 속성에 있는 컬럼 속성이고 동일한 속성과 상이하며, 제2 프리픽스가 제1 프리픽스와 상이하고,

[0134] 저장 모듈(520)은 키값 유형 분산형 데이터베이스에서 임의의 데이터 객체의 2차 인덱스를 저장하도록 추가로 구성된다.

[0135] 본 발명의 실시예에 따른 데이터 저장 장치에서, 데이터 객체의 2차 인덱스가 키값 유형 분산형 데이터베이스에 저장되어, 보다 다양한 쿼리 시나리오가 사용자에게 제공될 수 있으며, 사용자에 의한 쿼리가 가능하게 되어, 데이터 쿼리 효율을 향상시킬 수 있다.

[0136] 본 발명의 실시예에서, 선택적으로, 로우 식별자 포맷 내의 필드들 사이에 분리자가 설정되거나, 또는 필드들이 고정 길이를 갖는다.

[0137] 본 발명의 실시예에 따른 데이터 저장 장치(500)는 전술한 본 발명의 실시예에 따른 방법에서의 데이터 저장 장치에 대응될 수 있으며, 데이터 저장 장치(500)의 모듈들의 전술한 동작 미 기타 동작들 및/또는 기능들이 도 1 내지 도 4에서의 방법의 대응되는 절차를 구현하기 위해 별개로 수행된다. 간략히 하기 위해 상세한 설명은 여

기에서 중복되지 않는다.

[0138] 본 발명의 실시예에 따른 데이터 저장 장치서, 복수의 데이터 객체가 동일한 로우 식별자 포맷에 따라서 키값 유형 분산형 데이터 베이스 내에서 하나의 데이터 테이블에 저장되므로, 데이터베이스에 대한 관리가 용이해지며, 데이터 쿼리 효율을 향상시킬 수 있다.

[0139] 도 7은 본 발명의 다른 실시예에 따른 데이터 저장 장치의 구조를 도시하며, 본 발명의 다른 실시예에 따른 데이터 저장 장치는, 적어도 하나의 프로세서(702)(예를 들어, CPU), 적어도 하나의 네트워크 인터페이스(705) 또는 기타 통신 인터페이스, 메모리(706), 및 이들 장치들 사이의 연결 및 통신을 구현하도록 구성되는 적어도 하나의 통신 버스(703)를 포함한다. 프로세서(702)는 메모리(706)에 저장된, 컴퓨터 프로그램과 같은 실행 모듈을 실행하도록 구성된다. 메모리(706)는 고속의 랜덤 액세스 메모리(Random Access Memory, RAM)를 포함하며, 예를 들어, 적어도 하나의 자기 디스크 스토리지와 같은, 비휘발성 메모리(non-volatile memory)를 더 포함할 수 있다. 적어도 다른 네트워크 엘리먼트와의 통신 및 연결은 (유선 또는 무선일 수 있는) 적어도 하나의 네트워크 인터페이스(705)를 사용하여 구현된다.

[0140] 일부 구현 방식으로, 메모리(706)가 프로세서(702)에 의해 실행될 수 있는 프로그램(7061)을 저장하고, 이 프로그램이,

[0141] 키값 유형 분산형 데이터베이스에 저장될 적어도 두 개의 데이터 객체의 동일한 컬럼 속성을 결정하는 과정;

[0142] 키값 유형 분산형 데이터베이스에서, 적어도 두 개의 데이터 객체 내에서 각 데이터 객체의 로우 식별자의 포맷을 결정하는 과정 - 여기서 로우 식별자 포맷은 동일한 컬럼 속성과 데이터 객체 식별자를 포함하고, 적어도 두 개의 데이터 객체에서 상이한 데이터 객체의 데이터 객체 식별자는 로우 식별자 포맷에서 동일한 위치에 있으며 동일한 컬럼 속성 뒤에 위치함 -;

[0143] 각 데이터 객체에 대해 결정된 로우 식별자 포맷에 따라서 각 데이터 객체의 각 데이터 기록의 로우 식별자 값을 결정하는 과정; 및

[0144] 키값 유형 분산형 데이터베이스에 각 데이터 기록의 로우 식별자 값과 각 데이터 기록을 저장하는 과정을 포함하며,

[0145] 각 데이터 기록의 로우 식별자 값은 키값 유형 분산형 데이터베이스에서 각 데이터 기록의 1차 인덱스로서 사용된다.

[0146] 선택적으로, 프로그램은,

[0147] 적어도 두 개의 데이터 객체의 데이터 기록 모두가 클러스터 테이블을 형성하도록, 데이터 기록 모두의 로우 식별자 값에 대해, 제1 순서로 데이터 기록 모두를 저장하는 과정을 추가로 포함한다.

[0148] 선택적으로, 제1 순서는 사전식 순서이다.

[0149] 선택적으로, 프로그램은,

[0150] 적어도 두 개의 객체에서 데이터 기록을 쿼리하거나 또는 획득하도록 지시하는 데이터 요청을 수신하는 과정;

[0151] 로우 식별자 포맷에 따라서, 키값 유형 분산형 데이터베이스를 쿼리하기 위한, 동일한 컬럼 속성을 포함하는 쿼리 정보를 구축하는 과정; 및

[0152] 쿼리 정보에 따라서, 1차 인덱스가 쿼리 정보를 포함하는 데이터 기록을 키값 유형 분산형 데이터베이스로부터 내보내는 과정을 추가로 포함한다.

[0153] 선택적으로, 키값 유형 분산형 데이터베이스에서, 적어도 두 개의 데이터 객체 내에서 각 데이터 객체의 로우 식별자의 포맷을 결정하는 과정이,

[0154] 키값 유형 분산형 데이터베이스에서 적어도 두 개의 데이터 객체에 대해 상수인 로우 식별자의 제1 프리픽스를 특정하는 과정; 및

[0155] 적어도 두 개의 데이터 객체에서 제1 데이터 객체의 로우 식별자 포맷이 제1 프리픽스, 동일한 컬럼 속성, 및 제1 데이터 객체의 데이터 객체 식별자를 포함하는 것으로 결정하거나, 또는,

[0156] 적어도 두 개의 데이터 객체에서 제2 데이터 객체의 로우 식별자 포맷이 제1 프리픽스, 동일한 컬럼 속성, 제2 데이터 객체의 데이터 객체 식별자, 및 제2 데이터 객체의 컬럼 속성에 있으면서 제2 데이터 객체의 데이터 기

록 모두를 구별할 수 있는 컬럼 속성을 포함하는 것으로 결정하는 과정을 포함한다.

[0157] 선택적으로, 쿼리 정보는 제1 프리픽스 및 동일한 컬럼 속성을 포함한다.

[0158] 선택적으로, 프로그램은,

[0159] 키값 유형 분산형 데이터베이스에서, 적어도 두 개의 데이터 객체 중 임의의 데이터 객체의 2차 인덱스의 포맷이 제2 프리픽스, 제2 속성, 및 동일한 컬럼 속성인 것으로 결정하는 과정 - 여기서 제2 속성은 임의의 데이터 객체의 컬럼 속성에 있는 컬럼 속성이고 동일한 속성과 상이하며, 제2 프리픽스가 제1 프리픽스와 상이함 -;

[0160] 임의의 데이터 객체의 2차 인덱스의 포맷에 따라서 임의의 데이터 객체의 2차 인덱스를 결정하는 과정; 및

[0161] 키값 유형 분산형 데이터베이스에서 임의의 데이터 객체의 2차 인덱스를 저장하는 과정을 추가로 포함한다.

[0162] 선택적으로, 로우 식별자 포맷 내의 필드들 사이에 분리자가 설정되거나, 또는 필드들이 고정 길이를 갖는다.

[0163] 본 발명의 실시예에서 제공된 전술한 기술적 해결수단으로부터, 본 발명의 실시예에서, 상관된 데이터 객체가 동일한 컬럼 속성을 포함하는 로우 식별자 포맷을 사용하여 키값 유형 분산형 데이터베이스에 저장되어, 상관된 쿼리가 복수의 데이터 객체에 대해 구현될 수 있으므로, 데이터 쿼리 효율을 향상시킬 수 있다.

[0164] 본 발명의 실시예에서 용어 "및/또는"은 상관된 객체의 관계만을 설명하며 세 가지 관계가 존재하는 것을 나타내는 것이 이해되어야 한다. A 및/또는 B는 A만이 존재하거나, A와 B 둘 다 존재하거나, B만이 존재하는, 세 개의 경우를 나타낸다. 또한, 본 명세서에서 문자 "/"는 통상적으로 상관된 객체들 사이의 "또는"의 관계를 지시한다.

[0165] 통상의 기술자라면, 본 명세서에서 개시된 실시예에 기술된 예시들을 조합에 있어서, 유닛과 알고리즘 단계가 전자 하드웨어 또는 컴퓨터 소프트웨어, 또는 이들의 조합에 의해 구현될 수 있다는 것을 잘 알 것이다. 하드웨어와 소프트웨어 사이의 교체 가능성을 명확하게 설명하기 위해, 전술한 내용이 통상적으로 기능에 따라서 설명된 조합과 각 예의 단계를 갖는다. 기능들이 하드웨어에 의해 수행되는지 소프트웨어에 의해 수행되는지는 기술적 해결수단의 구체적인 적용 및 설계 제약 조건에 따라 결정된다. 통상의 기술자라면, 상술한 기능을 각각의 구체적인 적용을 위해 상이한 방법을 사용하여 구현할 수 있지만, 그 또한 본 발명의 보호 범위를 벗어난 것으로 취급되지 않는다.

[0166] 통상의 기술자라면, 간편하게 기술하기 위한 목적으로, 전술한 시스템, 장치 및 유닛의 구체적인 작동 프로세스에 대해, 전술한 방법 실시예에서의 대응하는 프로세스를 참조하고 상세한 반복 설명은 생략되었다는 것을 이해할 것이다.

[0167] 본 출원에서 제공된 여러 실시예에서, 개시된 시스템, 장치 및 방법은 다른 방식으로 구현될 수 있다. 예컨대, 기술된 장치 실시예는 단지 예시적이다. 예컨대, 유닛 분할은 단지 논리적 기능 분할이며 실제 구현에서는 다르게 나누어질 수 있다. 예컨대, 복수의 유닛 또는 컴포넌트는 다른 하나의 시스템으로 결합되거나 통합될 수 있고, 또는 일부 특징은 무시되거나 또는 수행되지 않을 수도 있다. 또한, 표시되거나 설명된 상호 간의 연결 또는 직접 연결 또는 통신 연결은 임의의 인터페이스를 통해 구현될 수 있다. 장치 또는 유닛 간의 간접적인 연결 또는 통신 연결은 전자, 기계 또는 다른 형태로 구현될 수 있다.

[0168] 별개의 부분으로 설명된 유닛은 물리적으로 독립된 것이거나, 물리적으로 독립되지 않은 것일 수 있고, 유닛으로 도시된 부분이 물리적 유닛일 수 있거나 물리적 유닛이 아닐 수 있거나, 또는 한 곳에 위치될 수 있거나, 복수의 네트워크 유닛 상에 분배되어 있을 수 있다. 일부 또는 모든 유닛은, 실시예의 해결수단의 목적을 달성하기 위해 실제 요건에 따라 선택될 수 있다.

[0169] 또한, 본 발명의 실시예에서의 기능적 유닛은 하나의 처리 유닛으로 통합될 수도 있고, 또는 각 유닛이 물리적으로 독립하여 존재할 수도 있으며, 또는 둘 이상의 유닛이 하나의 유닛으로 통합될 수도 있다. 통합된 유닛은 하드웨어의 형태로 구현될 수 있거나, 또는 소프트웨어 기능 유닛의 형태로 구현될 수 있다.

[0170] 통합된 유닛이 소프트웨어 기능 유닛의 형태로 구현되고 독립된 제품으로 판매되거나 사용되는 경우, 그 통합된 유닛은 컴퓨터에 의해 판독 가능한 저장 매체에 저장될 수 있다. 그러한 사상에 기초하는 경우, 본 발명의 기술적 해결수단은 필수적으로, 또는 종래 기술에 기여하는 부분 또는 그 기술적 해결수단 모두 또는 일부는 소프트웨어 제품의 형태로 구현될 수 있다. 소프트웨어 제품은 저장 매체에 저장되고, 컴퓨터 장치(개인용 컴퓨터, 서버, 또는 네트워크 장치)로 하여금 본 발명의 실시예에서 기술된 방법의 일부 또는 모든 단계를 수행하도록 하는 명령어를 포함한다. 전술한 저장 매체는, USB 플래시 드라이브, 탈착 가능한 하드 디스크, ROM(Read-Only

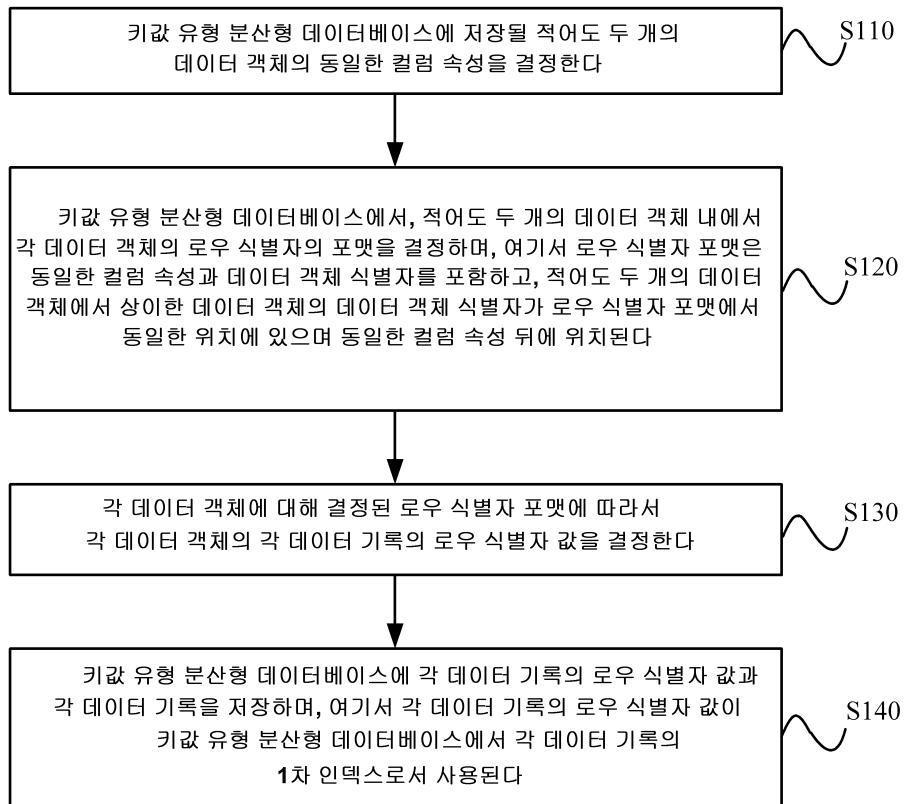
Memory), RAM(Random Access Memory), 자기 디스크 또는 광 디스크와 같은, 프로그램 코드를 저장할 수 있으면 어떤 매체라도 좋다.

[0171] 전술한 설명은 단지 본 발명의 구체적인 실시예일 뿐이며, 본 발명의 보호 범위를 제한하기 위한 것이 아니다. 통상의 기술자가 본 발명에서 개시된 기술적 범위 내에서 생각해 낼 수 있는 변형 또는 대체는 모두 본 발명의 보호 범위 내에 속한다. 따라서, 본 발명의 보호 범위는 청구범위의 보호 범위에 의해 결정되어야 한다.

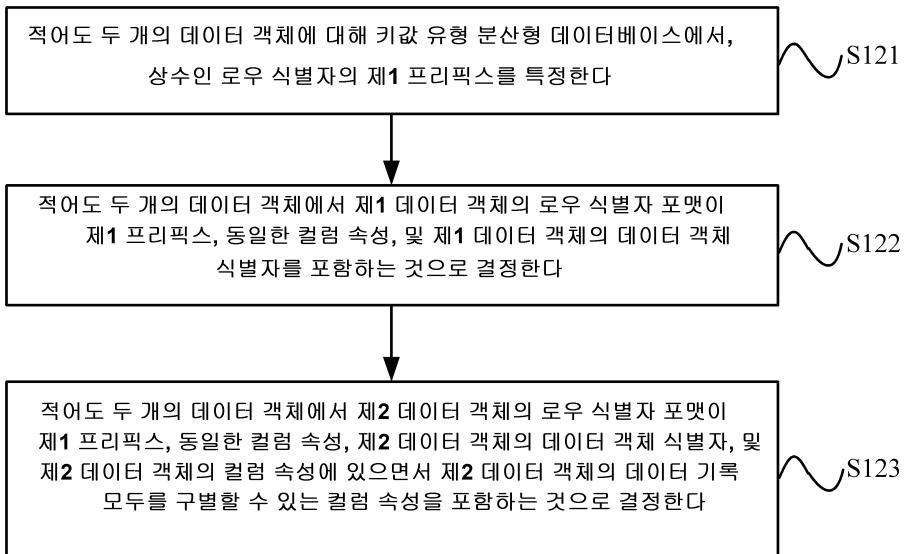
## 도면

### 도면1

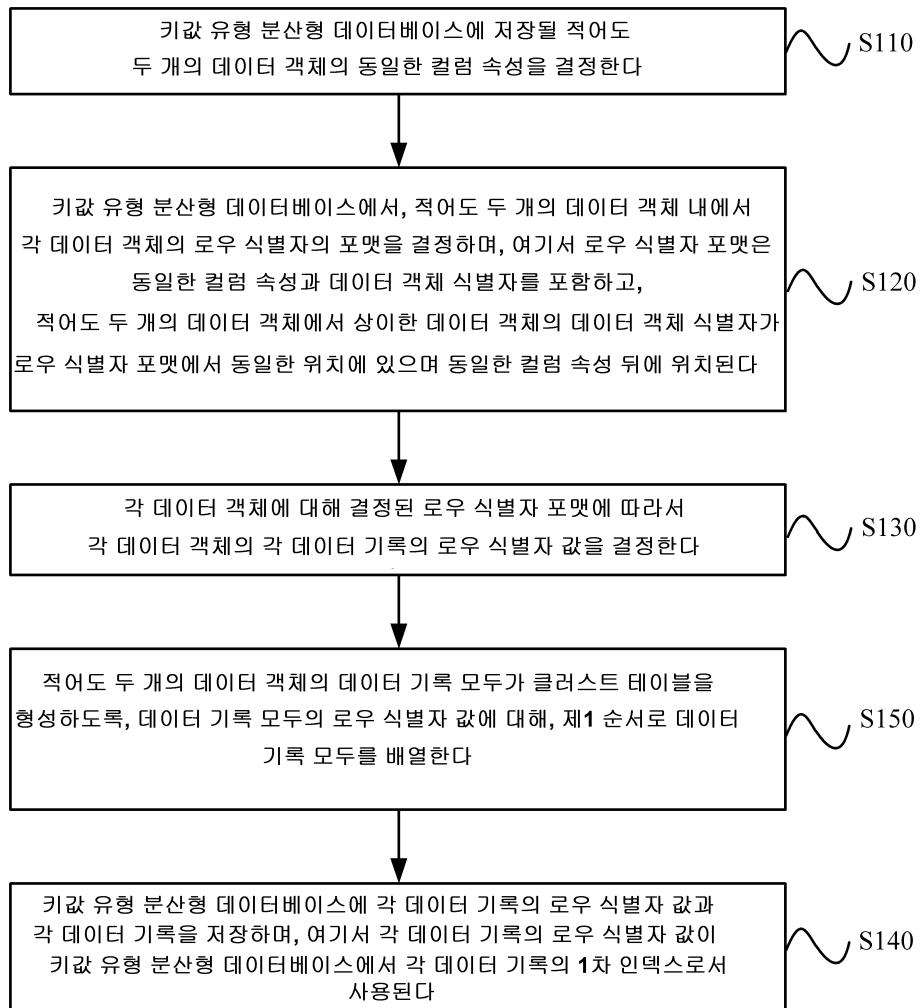
100



## 도면2

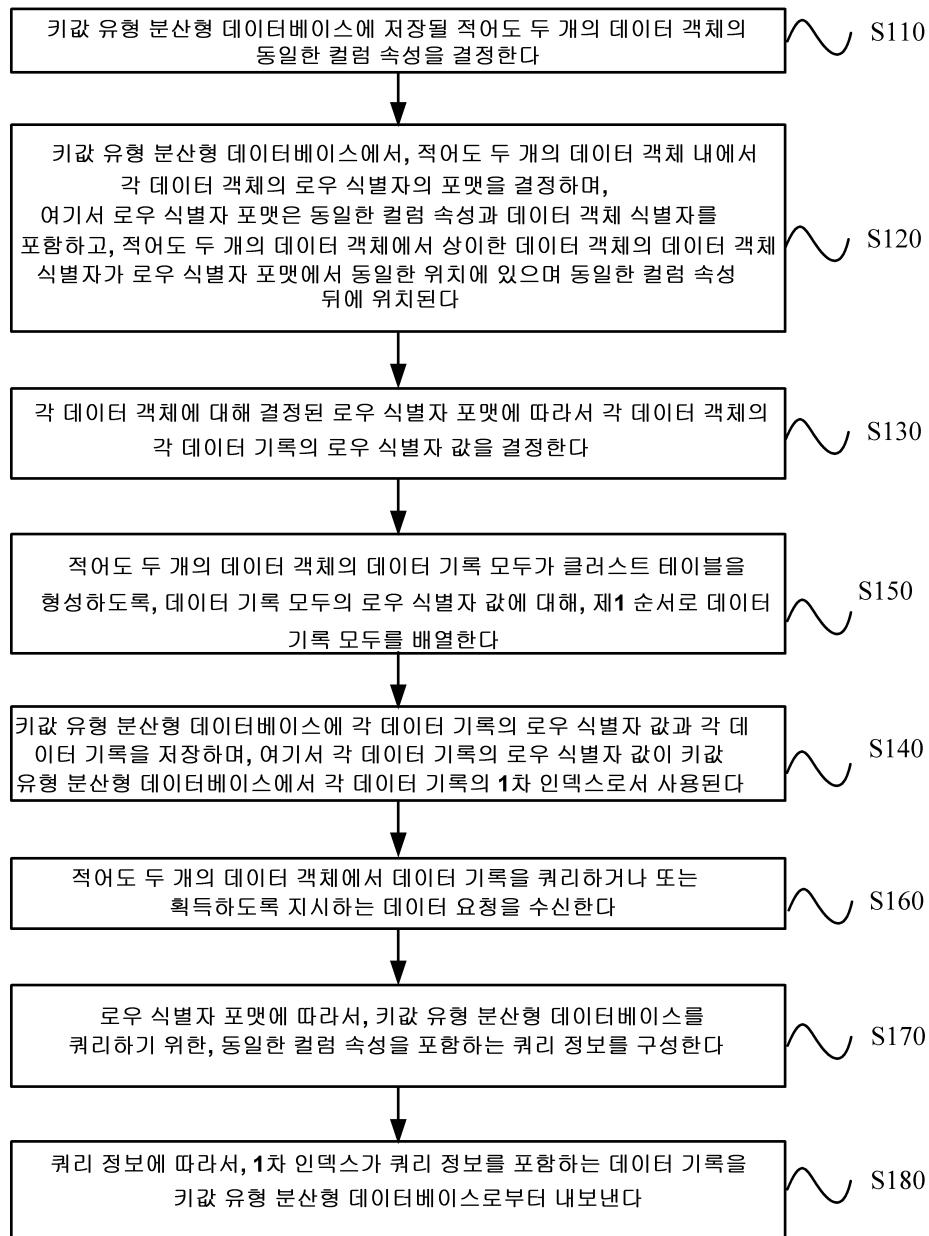
S120

## 도면3

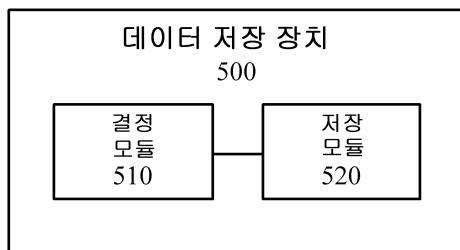
100

## 도면4

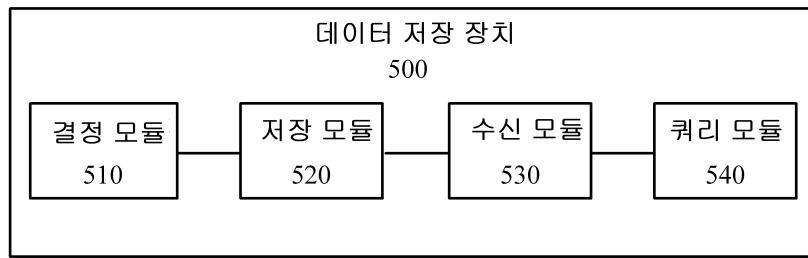
100



## 도면5



도면6



도면7

