



(19) 대한민국특허청(KR)  
(12) 공개특허공보(A)

(11) 공개번호 10-2017-0084400  
(43) 공개일자 2017년07월20일

(51) 국제특허분류(Int. Cl.)  
H04L 29/08 (2006.01) G06F 17/30 (2006.01)  
(52) CPC특허분류  
H04L 67/1031 (2013.01)  
G06F 17/30575 (2013.01)  
(21) 출원번호 10-2016-0003268  
(22) 출원일자 2016년01월11일  
심사청구일자 2016년01월11일

(71) 출원인  
충북대학교 산학협력단  
충청북도 청주시 서원구 충대로 1 (개신동)  
(72) 발명자  
유재수  
충청북도 청주시 서원구 월평로 24 805동 1001호  
(분평동, 현대대우아파트)  
복경수  
충청북도 청주시 상당구 중흥로 196 207동 402호  
(뒷면에 계속)  
(74) 대리인  
추혁, 김진동

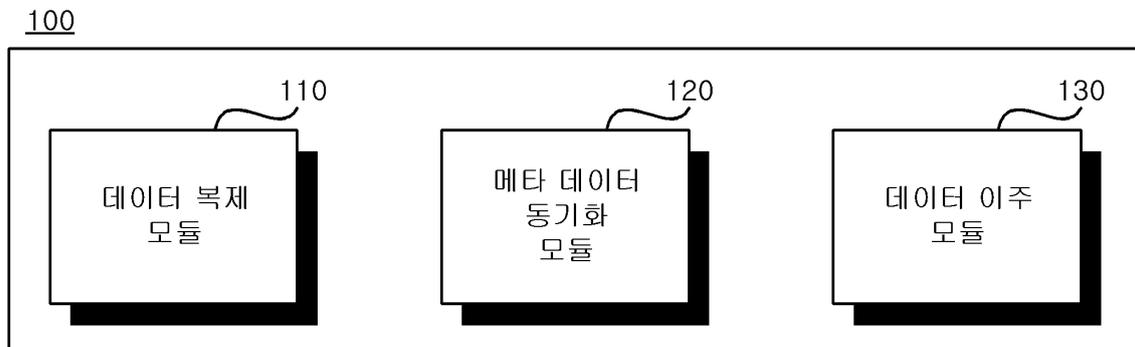
전체 청구항 수 : 총 9 항

(54) 발명의 명칭 분산 인-메모리 환경에서 데이터 복제 및 이주를 이용한 부하 분산 시스템

(57) 요약

본 발명은 분산 인-메모리 환경에서 데이터 복제 및 이주를 이용한 부하 분산 시스템에 관한 것으로서, 링 기반의 해시 기법을 사용하여 사용량이 많은 핫 데이터를 다른 노드에 복제하고, 이때 노드의 부하 상태를 고려하여 균등한 해시 범위로 핫 데이터를 복제하는 데이터 복제 모듈, 핫 데이터의 메타 데이터를 각 노드로부터 지속적으로 전송 받아 로드 밸런서에 전달하고, 로드 밸런서에 유지되는 핫 데이터의 메타 데이터를 주기적으로 클라이언트에 전송하는 메타 데이터 동기화 모듈 및 링 기반의 해시 기법을 사용하여 노드를 추가 또는 제거하고, 이때 모든 데이터를 재분배하는 것이 아니라 노드의 부하 상태를 고려하여 인접한 다른 노드에서 관리해야 할 해시 범위를 조정하여 일부 데이터만을 이주시키는 데이터 이주 모듈을 포함함으로써, 특정 노드에 집중될 수 있는 부하를 효율적으로 관리할 수 있다.

대표도 - 도2



(52) CPC특허분류

*G06F 17/30722* (2013.01)

*H04L 67/1029* (2013.01)

*H04L 67/1095* (2013.01)

(72) 발명자

**최기태**

충청북도 청주시 청원구 새터로176번길 62 202동  
801호

**임중태**

충청북도 청주시 서원구 창직로31번길 5 203호 (사  
창동, 스위트빌)

이 발명을 지원한 국가연구개발사업

과제고유번호 IITP-2015-H8501-15-10-13

부처명 미래창조과학부

연구관리전문기관 정보통신기술진흥센터

연구사업명 대학ICT연구센터육성지원사업

연구과제명 스마트 공간을 위한 빅데이터 기반의 스마트 라이프 케어 서비스

기여율 34/100

주관기관 충북대학교

연구기간 2013.06.01 ~ 2016.12.31이 발명을 지원한 국가연구개발사업

과제고유번호 2013H1B8A2032298

부처명 교육부

연구관리전문기관 한국연구재단

연구사업명 지역혁신인력양성사업

연구과제명 공공기관 대민 서비스를 위한 빅 데이터 처리 및 분석 시스템 개발

기여율 33/100

주관기관 충북대학교

연구기간 2013.05.01 ~ 2016.03.31이 발명을 지원한 국가연구개발사업

과제고유번호 B0101-15-0266

부처명 미래창조과학부

연구관리전문기관 정보통신기술진흥센터

연구사업명 정보통신·방송 연구개발사업

연구과제명 실시간 대규모 영상 데이터 이해·예측을 위한 고성능 비주얼 디스커버리 플랫폼 개발

기여율 33/100

주관기관 한국전자통신연구원

연구기간 2014.04.01 ~ 2024.02.29

## 명세서

### 청구범위

#### 청구항 1

링 기반의 해시 기법을 사용하여 사용량이 많은 핫 데이터를 다른 노드에 복제하고, 이때 노드의 부하 상태를 고려하여 균등한 해시 범위로 핫 데이터를 각각 복제하는 데이터 복제 모듈,

핫 데이터의 메타 데이터를 각 노드로부터 지속적으로 전송 받아 로드 밸런서에 전달하고, 로드 밸런서에 유지되는 핫 데이터의 메타 데이터를 주기적으로 클라이언트에 전송하는 메타 데이터 동기화 모듈 및

링 기반의 해시 기법을 사용하여 노드를 추가 또는 제거하고, 이때 모든 데이터를 재분배하는 것이 아니라 노드의 부하 상태를 고려하여 인접한 다른 노드에서 관리해야 할 해시 범위를 조정하여 일부 데이터만을 이주시키는 데이터 이주 모듈

을 포함하는 부하 분산 시스템.

#### 청구항 2

제1항에서,

상기 데이터 복제 모듈은,

상기 핫 데이터를 복제할 때 복제되는 핫 데이터의 수에 따라서 전체 해시 범위를 균등하게 분할하는 단계 및

각각의 분할 해시 범위 안에서 작은 해시 값부터 순차적으로 검사하여 과부하 상태가 아닌 노드에 상기 핫 데이터를 복제하는 단계

를 수행하는

부하 분산 시스템.

#### 청구항 3

제2항에서,

상기 전체 해시 범위를 분할하는 단계는,

$Hash_{max} \times \frac{i}{numofreplica} + Hash_{hotdata}$  을 이용하여 해시 값( $Hash_{replica(i)}$ )을 계산하는 단계로서, 여기서

$Hash_{replica(i)}$ 는 분할 해시 범위를 구하기 위해 사용되는 해시 값,  $Hash_{max}$ 는 전체 해시 범위,  $i$ 는 1, 2, ...,

복제본의 수,  $\frac{i}{numofreplica}$ 는 계산하는 해시 값의 수,  $Hash_{hotdata}$ 는 핫 데이터의 해시 값을 나타내는, 상기 해시 값을 계산하는 단계 및

$Hash_{replica(i)} \sim Hash_{replica(i+1)}$ 를 이용하여 분할 해시 범위( $Hash_{range(i)}$ )를 산출하는 단계로서, 여기서

$Hash_{replica(i)} \sim Hash_{replica(i+1)}$ 는 상기 해시 값을 이용하여 전체 해시 범위를 분할하는, 상기 분할 해시 범위를 산출하는 단계

를 포함하는

부하 분산 시스템.

#### 청구항 4

제1항에서,

상기 메타 데이터 동기화 모듈은,

로드 밸런서에 대한 접근을 감소시키기 위해 상기 핫 데이터의 메타 데이터를 각 노드로부터 지속적으로 전송 받아 상기 로드 밸런서에 전달함으로써 상기 로드 밸런서가 시스템 전체의 상기 핫 데이터의 메타 데이터를 관리하게 하고, 상기 로드 밸런서에 유지되는 상기 핫 데이터의 메타 데이터를 주기적으로 상기 클라이언트에 전송하여 상기 클라이언트가 상기 로드 밸런서를 거치지 않고 상기 핫 데이터의 메타 데이터를 이용하여 직접 핫 데이터를 보유한 노드에 접근하게 하는

부하 분산 시스템.

**청구항 5**

제1항에서,

상기 데이터 이주 모듈은,

새로운 노드 추가 시 전체 노드들 중 부하가 가장 큰 노드와 반시계 방향으로 이웃한 노드 사이에 상기 새로운 노드를 추가하고 상기 부하가 가장 큰 노드의 해시 범위의 일부를 상기 새로운 노드로 이주시키는

부하 분산 시스템.

**청구항 6**

제5항에서,

상기 새로운 노드의 해시 값은  $S_k = \frac{|S_j|}{n}$  이고, 여기서  $|S_j|$ 는 상기 부하가 가장 큰 기존 노드의 해시 범위, n은 가변 상수인 부하 분산 시스템.

**청구항 7**

제6항에서,

상기 n은 2인 부하 분산 시스템.

**청구항 8**

제1항에서,

상기 데이터 이주 모듈은, 노드 제거 시 제거되는 노드의 양 이웃 노드들의 부하 상태를 고려하여 상기 제거되는 노드의 해시 범위를 일정한 비율로 양 이웃 노드들로 이주시키는 부하 분산 시스템.

**청구항 9**

제8항에서,

상기 제거되는 양 이웃 노드 중 시계 방향 이웃 노드( $S_i$ )의 해시 값은

$S_i = \frac{Node\_Load_i}{Node\_Load_i + Node\_Load_j} \times |S_k|$  이고, 상기 제거되는 양 이웃 노드 중 반시계 방향 이웃 노드( $S_j$ )의 해시

값은  $S_j = \frac{Node\_Load_j}{Node\_Load_i + Node\_Load_j} \times |S_k|$  이며, 여기서  $\{Node\_Load_i\}$ 는 상기 노드  $S_i$ 의 부하,  $Node\_Load_j$ 는 상기 노드  $S_j$ 의 부하,  $|S_k|$ 는 상기 제거되는 노드의 해시 범위인 부하 분산 시스템.

**발명의 설명**

**기술 분야**

본 발명은 부하 분산 시스템에 관한 것으로, 더욱 상세하게는 분산 메모리 환경에서 링 기반의 해시 기법을 사

[0001]

용하여 데이터 복제 및 이주를 수행함으로써 특정 노드에 집중될 수 있는 부하를 효율적으로 관리하는 부하 분산 시스템에 관한 것이다.

### 배경 기술

[0002] 최근 Twitter나 Facebook과 같은 소셜 미디어의 급격한 성장과 스마트폰 같은 디지털 기기 사용이 증가하면서 데이터양이 기하급수적으로 증가하였다. 기존의 저장 및 분석 시스템의 처리 한계를 넘어서는 데이터량의 증가로 인해 하둡(hadoop)이나 병렬 DBMS(Database Management System)와 같은 분산 저장 및 관리 기술이 활용되고 있다.

[0003] 하지만 디스크에 데이터를 저장하고 처리할 경우 I/O 속도로 인해 병목 현상이 발생하여 전체적인 처리 속도가 저하되는 문제가 발생한다. 이러한 문제점을 해결하기 위해 디스크에 비해 I/O 속도가 빠른 메모리에 데이터를 저장하고 처리하는 인-메모리 기술이 중요하게 부각되고 있다. 인-메모리 기술은 데이터를 하드 디스크가 아닌 메모리에 적재하여 사용하기 때문에 데이터를 빠르게 접근하고 처리할 수 있다. 이러한 분산 인-메모리 기술은 Facebook과 Twitter와 같이 방대한 양의 데이터를 실시간으로 처리하는 기업에서 많이 활용되고 있다. 대표적인 인-메모리 처리 기술로는 맴캐시(memcached)가 있다. 맴캐시는 Facebook, Twitter, Reddit, YouTube와 같이 클라우드 및 웹 서비스 제공 회사에서 사용하는 키-값(key-value) 기반의 메모리 캐시이다. 이러한 맴캐시는 분산 환경에서 각각의 메모리를 하나의 저장소처럼 관리하여 사용하기 때문에 백-엔드 시스템에 연결된 저장소에 대한 접근 비용과 시간을 감소시킬 수 있다.

[0004] 분산 메모리 환경에서 데이터를 처리할 때 특정 노드에 과부하가 발생하면 처리 성능이 저하되는 문제가 발생한다. 따라서 노드의 부하 분산을 처리하기 위한 연구들이 진행되고 있다. 부하 분산을 위한 기법으로 데이터 이주나 복제, 이주와 복제를 혼합한 기법을 많이 사용한다. 예를 들어, APA(Adaptive Performance-Aware distributed memory caching)는 노드의 부하를 계산하기 위해 데이터 적중률과 사용률을 기반으로 노드의 비용을 계산하고, 계산된 비용을 이용하여 노드의 해시 공간을 조정하고 데이터를 이주하여 노드의 부하를 분산하는 기법을 제안하였다. 노드의 데이터 중에서 접근이 많은 핫 데이터는 노드의 부하에서 많은 양을 차지하고 있다. 하지만 핫 데이터를 고려하지 않고 노드의 부하만을 이용하여 해시범위를 조정할 경우 각 노드의 부하가 차이가 클수록 조정해야 하는 해시 범위가 많아지고 이로 인해 많은 데이터가 이주하여 이주비용이 증가한다. 과부하 노드의 이웃노드 또한 과부하 상태일 경우 해시범위를 조정하여 부하분산을 처리할 수 없다. 또한, ECMS(Efficient Cache Management Scheme)는 핫 데이터를 다른 노드로 이주하여 부하분산을 관리한다. 하지만 핫 데이터를 이주하게 되면 데이터가 이주된 노드의 부하를 상승시켜 다시 다른 노드의 과부하를 발생시킬 수 있다. 그로 인하여 기존 과부하 노드는 부하가 줄어들지만, 핫 데이터를 받은 노드는 부하가 크게 증가하여 과부하를 발생시키는 문제점이 있다.

### 선행기술문헌

#### 특허문헌

[0005] (특허문헌 0001) 대한민국 공개특허공보 제10-2011-0070772호(공개일 2011.06.24.)

(특허문헌 0002) 대한민국 등록특허공보 제10-1419379호(공고일 2014.07.15.)

### 발명의 내용

#### 해결하려는 과제

[0006] 따라서, 본 발명은 상기한 종래 기술의 문제점을 해결하기 위해 이루어진 것으로서, 본 발명의 목적은 핫 데이터 복제 및 노드의 추가/제거 시에 노드의 부하를 고려하여 해시 공간을 조정하고 데이터를 복제 및 이주하여 부하를 분산시킴으로써 특정 노드에 집중될 수 있는 부하를 효율적으로 관리하는 부하 분산 시스템을 제공하는 데 있다.

#### 과제의 해결 수단

[0007] 상기와 같은 목적을 달성하기 위한 본 발명의 부하 분산 시스템은, 링 기반의 해시 기법을 사용하여 사용량이 많은 핫 데이터를 다른 노드에 복제하고, 이때 노드의 부하 상태를 고려하여 균등한 해시 범위로 핫 데이터를

각각 복제하는 데이터 복제 모듈, 핫 데이터의 메타 데이터를 각 노드로부터 지속적으로 전송 받아 로드 밸런서에 전달하고, 로드 밸런서에 유지되는 핫 데이터의 메타 데이터를 주기적으로 클라이언트에 전송하는 메타 데이터 동기화 모듈 및 링 기반의 해시 기법을 사용하여 노드를 추가 또는 제거하고, 이때 모든 데이터를 재분배하는 것이 아니라 노드의 부하 상태를 고려하여 인접한 다른 노드에서 관리해야 할 해시 범위를 조정하여 일부 데이터만을 이주시키는 데이터 이주 모듈을 포함한다.

**발명의 효과**

[0008] 상술한 바와 같이, 본 발명에 의한 분산 인-메모리 환경에서 데이터 복제 및 이주를 이용한 부하 분산 시스템에 따르면, 핫 데이터 복제 및 노드의 추가/제거 시에 노드의 부하를 고려하여 해시 공간을 조정하고 데이터를 복제 및 이주하여 부하를 분산시킴으로써 특정 노드에 집중될 수 있는 부하를 효율적으로 관리할 수 있다.

[0009] 또한, 클라이언트가 핫 데이터에 대한 실시간으로 동기화된 메타 데이터를 보유하게 하여 핫 데이터에 접근할 때 중앙 서버를 거치지 않고 직접 데이터에 접근하게 함으로써 중앙 서버의 부하를 감소시키고 노드에 대한 접근 속도가 빨라질 수 있다.

**도면의 간단한 설명**

- [0010] 도 1은 본 발명의 부하 분산 시스템 및 방법이 적용되는 분산 메모리 환경을 나타내는 전체 구성도이다.
- 도 2는 도 1과 같은 분산 메모리 환경에 적용되는 본 발명에 따른 부하 분산 시스템의 전체 구성을 개략적으로 나타낸 블록도이다.
- 도 3은 데이터 복제 모듈이 본 발명의 실시예에 따라 분산 메모리 환경에서 핫 데이터를 복제하는 과정을 나타낸다.
- 도 4는 메타 데이터 동기화 모듈이 본 발명의 실시예에 따라 분산 메모리 환경에서 핫 데이터의 메타 데이터를 동기화하는 과정을 나타낸다.
- 도 5는 데이터 이주 모듈이 본 발명의 실시예에 따라 분산 메모리 환경에 새로운 노드를 추가하는 과정을 나타낸다.
- 도 6은 본 발명의 실시예에 따라 분산 메모리 환경에서 노드를 제거하는 과정을 나타낸다.

**발명을 실시하기 위한 구체적인 내용**

[0011] 이하에서 제시된 본 발명에 의한 부하 분산 시스템은 로드 밸런서(load balancer)와 데이터를 분배하여 저장 및 처리하는 하나 이상의 노드로 구성되어, 특정 노드에 과부하가 발생할 경우 과부하가 발생한 노드의 핫 데이터를 다른 노드에 복제하고, 새로운 노드가 추가되는 경우 과부하가 가장 큰 노드의 데이터를 새로운 노드로 이주하여 부하를 분산하고, 클라이언트가 핫 데이터를 요청하는 경우 클라이언트로 하여금 자신이 보유한 핫 데이터의 메타 데이터를 이용하여 핫 데이터를 보유한 노드에 접근하게 하며, 클라이언트가 핫 데이터가 아닌 데이터를 요청하는 경우에는 로드 밸런서를 통해 데이터가 저장된 노드에 접근하게 하는 분산 메모리 환경을 바탕으로 하는 경우를 바람직한 실시예로서 제안한다.

- [0012] 도 1은 본 발명의 부하 분산 시스템이 적용되는 분산 메모리 환경을 나타내는 전체 구성도이다.
- [0013] 도 1을 참조하면, 본 발명의 부하 분산 시스템이 적용되는 분산 메모리 환경은 로드 밸런서와 노드1 내지 노드4로 구성되어, Node1에서 과부하가 발생할 경우 자주 사용되는 핫 데이터를 다른 노드인 Node2와 Node3에 복제하고, 새로운 노드인 Node4가 추가되는 경우 기존 노드인 Node3에서 Node4로 데이터를 이주하여 부하를 분산한다. 그리고, 클라이언트가 data1을 요청할 경우 클라이언트가 자신이 보유한 핫 데이터의 메타 데이터를 이용하여 핫 데이터인 data1을 보유한 Node1에 직접 접근하게 하는 한편, 클라이언트가 data10을 요청할 경우에는 data10이 핫 데이터가 아니므로 로드 밸런서를 통하여 data10이 저장된 node3에 접근하게 한다.
- [0014] 이하, 본 발명의 부하 분산 시스템 대하여 첨부된 도면을 참조하여 상세히 설명하기로 한다.
- [0015] 도 2는 도 1과 같은 분산 메모리 환경에 적용되는 본 발명에 따른 부하 분산 시스템의 전체 구성을 개략적으로 나타낸 블록도이다.
- [0016] 도 2에 도시된 바와 같이, 본 발명에 따른 부하 분산 시스템은 링 기반의 해시 기법을 사용하여 사용량이 많은 핫 데이터를 다른 노드에 복제하고, 이때 노드의 부하 상태를 고려하여 균등한 해시 범위로 핫 데이터를 각각

복제하는 데이터 복제 모듈(110), 핫 데이터의 메타 데이터를 각 노드로부터 지속적으로 전송 받아 로드 밸런서에 전달하고, 로드 밸런서에 유지되는 핫 데이터의 메타 데이터를 주기적으로 클라이언트에 전송하는 메타 데이터 동기화 모듈(120) 및 링 기반의 해시 기법을 사용하여 노드를 추가 또는 제거하고, 이때 모든 데이터를 재분배하는 것이 아니라 노드의 부하 상태를 고려하여 인접한 다른 노드에서 관리해야 할 해시 범위를 조정하여 일부 데이터만을 이주시키는 데이터 이주 모듈(130)을 포함하여 이루어진다.

[0017] 먼저, 본 발명의 데이터 복제 모듈(110)은 특정 노드에 집중되는 부하를 분산시키기 위해 링 기반의 해시 기법을 사용하여 사용량이 많은 핫 데이터를 다수의 노드에 복제하고, 이때 노드의 부하 상태를 고려하여 균등한 해시 범위로 각각 핫 데이터를 복제한다. 이와 같이, 본 발명에 따르면, 특정 노드에 집중될 수 있는 부하를 분배하는 것이 중요한 도 1과 같은 분산 메모리 환경에서, 특히 특정 노드에 접근이 많아 부하를 크게 발생시키는 핫 데이터가 발생하더라도, 핫 데이터를 적절하게 복제하여 분배함으로써 과부하 된 노드의 부하를 효율적으로 감소시킬 수 있다.

[0018] 도 3은 데이터 복제 모듈(110)이 본 발명의 실시예에 따라 분산 메모리 환경에서 핫 데이터를 복제하는 과정을 나타낸다.

[0019] 분산 메모리 환경에서 노드가 삭제될 경우 통상적으로 삭제되는 노드의 데이터를 이웃 노드로 이주하게 되는데, 이에 따라 만약 연속된 이웃 노드에 데이터가 복제되고 복제 데이터를 가진 노드가 제거될 경우에 하나의 노드에 핫 데이터가 중복해서 저장될 수 있다. 이렇게 되면, 같은 노드에 두 개의 핫 데이터 복제본이 저장되기 때문에 메모리 공간이 낭비되고 부하도 집중된다. 따라서, 본 발명의 데이터 복제 모듈(110)은 분산 메모리 환경에서 핫 데이터를 복제할 때 복제되는 핫 데이터의 수에 따라서 전체 해시 범위를 고르게 분할하고, 분할 범위 안에서 작은 해시 값부터 순차적으로 검사하여 과부하 상태가 아닌 노드에 핫 데이터를 복제하여, 핫 데이터의 복제본을 가진 노드가 제거되어도 하나의 노드에 데이터가 중복 저장되는 것을 방지할 수 있다. 도 3을 참조하면, 원본 데이터를 포함한 데이터의 복제 수가 3개일 때, 원본 데이터의 해시 값을 기준으로 전체 해시 범위를 균등하게 3개로 분할한다. 그 후, 원본 데이터는 기존 저장 노드에 유지한다. 첫 번째 복제본은 range2의 범위 내에서 순차적으로 부하를 고려하여 저장을 한다. range2의 첫 번째 노드인 N6이 과부하 상태에 있다면 다음 노드인 N7을 검사하여 N7이 과부하 상태가 아니면 N7에 두 번째 데이터를 복제한다. 두 번째 복제본은 range3의 범위에 첫 번째 노드인 N10이 과부하 상태가 아니면 N10에 복제한다.

[0020] 데이터 복제를 위해 노드를 선택하기 위한 해시 범위는 수학적 식 1 및 2를 사용하여 분할한다. 먼저 수학적 식 1은 데이터 복제를 위해 해시 범위를 분할하기 위해 사용되는 해시 값을 계산한다. 여기서,  $Hash_{replica(i)}$ 는 분할 범위를 구하기 위해 사용되는 해시 값,  $Hash_{max}$ 는 전체 해시 범위,  $\frac{i}{numofreplica}$ 는 계산하는 해시 값의 수,  $Hash_{hotdata}$ 는 핫 데이터의 해시 값을 나타낸다. 다음으로, 수학적 식 2는 수학적 식 1에서 계산한 해시 값을 사용하여 데이터 복제를 위해 노드를 선택하기 위한 분할 해시 범위를 산출한다.  $Hash_{range(i)}$ 는 해시 범위,  $Hash_{replica(i)} \sim Hash_{replica(i+1)}$ 는 수학적 식 1에서 계산한 해시 값을 이용하여 해시 범위를 분할한 것이다.

**수학적 식 1**

[0021] 
$$Hash_{replica(i)} = Hash_{max} \times \frac{i}{numofreplica} + Hash_{hotdata} \quad (i는 1, 2, \dots, 복제본의 수)$$

**수학적 식 2**

[0022] 
$$Hash_{range(i)} = Hash_{replica(i)} \sim Hash_{replica(i+1)}$$

- [0023] 다음으로, 메타 데이터 동기화 모듈(120)은 핫 데이터의 메타 데이터를 각 노드로부터 지속적으로 전송 받아 로드 밸런서에 전달하고, 로드 밸런서에 유지되는 핫 데이터의 메타 데이터를 주기적으로 클라이언트에 전송한다. 구체적으로, 본 발명의 메타 데이터 동기화 모듈(120)은 로드 밸런서에 대한 접근을 감소시키기 위해 핫 데이터의 메타 데이터를 각 노드로부터 지속적으로 전송 받아 로드 밸런서에 전달함으로써 로드 밸런서가 시스템 전체의 핫 데이터의 메타 데이터를 관리하게 하고, 로드 밸런서에 유지되는 핫 데이터의 메타 데이터를 주기적으로 클라이언트에 전송하여 클라이언트가 로드 밸런서를 거치지 않고 핫 데이터의 메타 데이터를 이용하여 직접 핫 데이터를 보유한 노드에 접근하게 한다. 이와 같이, 본 발명에 따르면, 클라이언트가 핫 데이터가 아닌 데이터를 요청하는 경우에만 로드 밸런서를 통해 데이터가 저장된 노드에 접근하게 하고 클라이언트가 핫 데이터를 요청하는 경우에는 클라이언트로 하여금 자신이 보유한 핫 데이터의 메타 데이터를 이용하여 핫 데이터를 보유한 노드에 접근하게 함으로써, 로드 밸런서에 대한 접근 및 그에 따른 부하를 감소시키고 전체 시스템의 성능을 향상시킬 수 있다.
- [0024] 도 4는 메타 데이터 동기화 모듈(120)이 본 발명의 실시예에 따라 분산 메모리 환경에서 핫 데이터의 메타 데이터를 동기화하는 과정을 나타낸다.
- [0025] 분산 메모리 환경에서 클라이언트가 노드에 접근하기 위해서는 통상적으로 로드 밸런서를 통해 데이터를 보유한 노드에 접근을 한다. 이렇게 해서 데이터 요청이 많아지게 되면 로드 밸런서의 부하가 많아지고 성능이 저하되기 때문에 전체적인 시스템 성능이 저하된다. 따라서, 본 발명의 메타 데이터 동기화 모듈(120)은 로드 밸런서가 노드들로부터 핫 데이터에 대한 정보를 받아 별도의 핫 데이터의 메타 데이터를 종합하여 관리하게 하여, 로드 밸런서의 접근을 감소시킨다. 도 4를 참조하여, data1 및 data12가 핫 데이터이고 데이터가 복제가 되었을 경우 로드 밸런서는 핫 데이터의 복제본의 메타 데이터를 유지하고, 시스템을 사용하는 클라이언트 #1, 클라이언트 #2, 클라이언트 #3과 같은 모든 클라이언트는 로드 밸런서에 업데이트된 핫 데이터의 메타 데이터를 주기적으로 전송 받아 최신의 메타정보를 유지하도록 한다. 핫 데이터를 동기화하지 않으면 클라이언트가 핫 데이터의 메타 데이터를 사용하여 노드에 접근할 때 잘못된 노드에 접근할 수 있기 때문에, 본 발명에 따르면 핫 데이터의 복제본에 대해 동기화된 메타 데이터를 주기적으로 클라이언트에게 전송함으로써 클라이언트가 노드에 잘못 접근하는 것을 방지할 수 있다.
- [0026] 데이터 이주 모듈(130)은 링 기반의 해시 기법을 사용하여 노드를 추가 또는 제거하고, 이때 모든 데이터를 재분배하는 것이 아니라 노드의 부하 상태를 고려하여 인접한 다른 노드에서 관리해야 할 해시 범위를 조정하여 일부 데이터만을 이주시킨다. 구체적으로, 데이터 이주 모듈(130)은 새로운 노드 추가 시 전체 노드들 중 부하가 가장 큰 노드와 반시계 방향으로 이웃한 노드 사이에 새로운 노드를 추가하고 상기 부하가 가장 큰 노드의 해시 범위의 일부를 새로운 노드로 이주시키고, 노드 제거 시에는 제거되는 노드의 양 이웃 노드들의 부하 상태를 고려하여 상기 제거되는 노드의 해시 범위를 분배하여 양 이웃 노드들로 이주시킨다. 이와 같이, 본 발명에 따르면, 분산 메모리 환경에서 노드가 추가 또는 제거되는 경우 분산된 노드에 모든 데이터를 재분배하지 않고 인접한 다른 노드에서 관리해야 할 해시 값을 조정하여 일부 데이터만을 재분배함으로써 시스템 전체에 부하가 발생하는 것을 방지하고 시스템 전체 부하를 감소시킬 수 있다.
- [0027] 도 5는 데이터 이주 모듈(130)이 본 발명의 실시예에 따라 분산 메모리 환경에 새로운 노드를 추가하는 과정을 나타낸다.
- [0028] 데이터 이주 모듈(130)은 본 발명의 분산 메모리 환경에 새로운 노드가 추가되는 경우 새로운 노드를 전체 노드들 중 부하가 가장 큰 노드와 반시계 방향으로 이웃한 노드 사이에 추가하고 상기 부하가 가장 큰 노드의 해시 범위의 일부, 바람직하게는 50%를 새로운 노드로 이주시켜, 부하가 가장 큰 노드의 부하를 감소시킨다. 도 5를 참조하면, 노드  $S_j$ 가 전체 노드들 중에서 부하가 가장 큰 노드이고 그러한 노드  $S_j$ 가 관리하는 해시 범위가 0 내지 1000인 경우, 새로운 노드( $S_k$ )는 부하가 가장 큰 노드( $S_j$ )의 반시계 방향의 이웃 노드로 추가되고 부하가 가장 큰 노드( $S_j$ )의 해시 범위의 일부, 바람직하게는 50%인 0 내지 500을 관리한다. 이와 같이, 본 발명에 따르면, 분산 메모리 환경에 새로운 노드 추가 시 부하가 가장 큰 노드의 해시 범위를 새로 추가되는 노드가 일부 관리하게 함으로써 기존 과부하 노드의 부하를 감소시킬 수 있다.
- [0029] 새로운 노드의 해시 값은 수학식 3과 같이 연산한다. 여기서,  $S_k$ 는 새롭게 추가되는 노드의 해시 값을 나타내고,  $|S_j|$ 는 부하가 가장 큰 기존 노드의 해시 범위를 나타내며, n은 가변 상수로서 바람직하게는 2이다.

수학식 3

$$S_k = \frac{|S_j|}{n}$$

[0030]

[0031] 도 6은 데이터 이주 모듈(130)이 본 발명의 실시예에 따라 분산 메모리 환경에서 노드를 제거하는 과정을 나타낸다.

[0032] 분산 메모리 환경에서 노드가 제거되면 그 시계 방향의 이웃 노드가 제거되는 노드의 해시 범위를 관리하게 된다. 그러면 기존 시계 방향의 노드에 부하가 집중되어 사용자의 요청을 처리하는데 지연이 발생할 수 있다. 따라서, 본 발명의 데이터 이주 모듈(130)은 이를 해결하기 위해 노드가 제거되는 경우 제거되는 노드를 기준으로 양 이웃 노드들의 부하 상태를 고려하여 제거되는 노드의 해시 범위를 일정한 비율로 분배하여 양 이웃 노드들로 이주시킴으로써, 제거되는 노드의 시계 방향 이웃 노드로 부하가 집중되는 것을 방지한다. 도 6을 참조하면, 기존 노드( $S_k$ )가 제거되는 경우, 기존 노드( $S_k$ )의 시계 방향 노드( $S_i$ )와 반시계 방향 노드( $S_j$ )가 기존 노드( $S_k$ )의 해시 범위를 분배하여 관리한다. 따라서, 기존 노드( $S_k$ )가 관리하던 해시범위가 500~1000이고 시계 방향 노드( $S_i$ )의 부하가 20, 반시계 방향 이웃 노드( $S_j$ )의 부하가 30일 경우, 시계 방향 노드( $S_i$ )는 해시범위 500~700을 추가로 관리하고 반시계 방향 이웃 노드( $S_j$ )는 해시범위 701~1000을 추가로 관리하도록 부하를 분배한다.

[0033] 제거되는 노드의 상기 양 이웃 노드들의 해시 값은 수학식 4와 같이 연산한다. 여기서,  $S_i$ 는 시계 방향 노드의 해시 값,  $S_j$ 는 반시계 방향 노드의 해시 값,  $\{Node\_Load\_i\}$ 는 노드  $S_i$ 의 부하,  $Node\_Load_j$ 는 노드  $S_j$ 의 부하,  $|S_k|$ 는 제거되는 노드의 해시 범위를 나타낸다.

수학식 4

$$S_i = \frac{Node\_Load_i}{Node\_Load_i + Node\_Load_j} \times |S_k|$$

$$S_j = \frac{Node\_Load_j}{Node\_Load_i + Node\_Load_j} \times |S_k|$$

[0034]

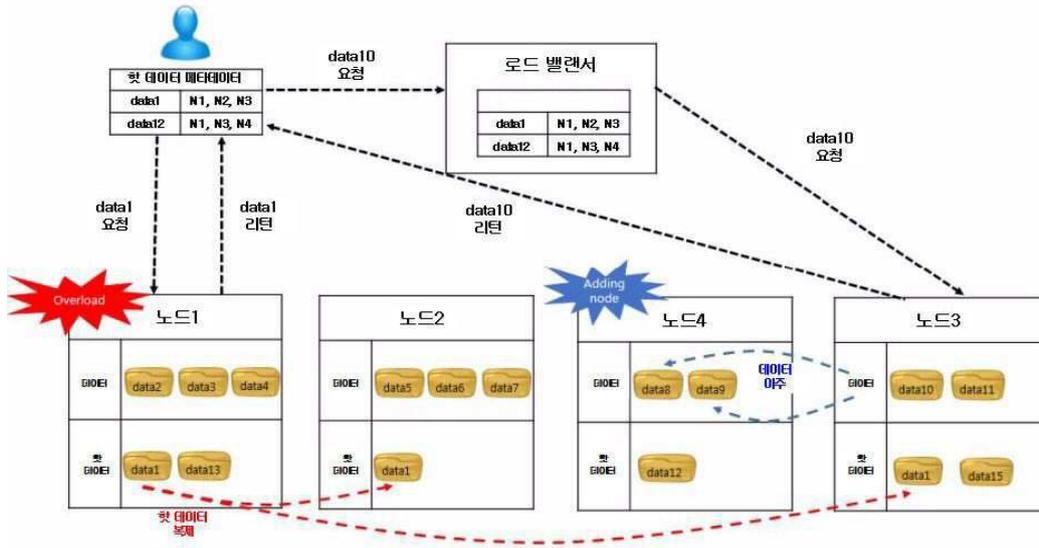
[0035] 이상에서 몇 가지 실시예를 들어 본 발명을 더욱 상세하게 설명하였으나, 본 발명은 반드시 이러한 실시예로 국한되는 것이 아니고 본 발명의 기술사상을 벗어나지 않는 범위 내에서 다양하게 변형 실시될 수 있다.

부호의 설명

- [0036] 100 : 부하 분산 시스템
- 110 : 데이터 복제 모듈
- 120 : 메타 데이터 동기화 모듈
- 130 : 데이터 이주 모듈

도면

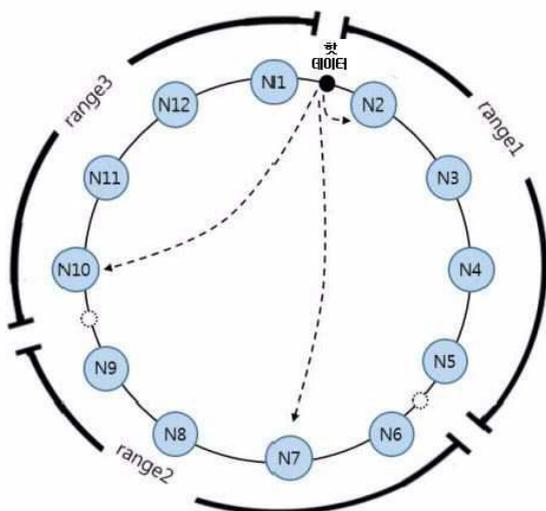
도면1



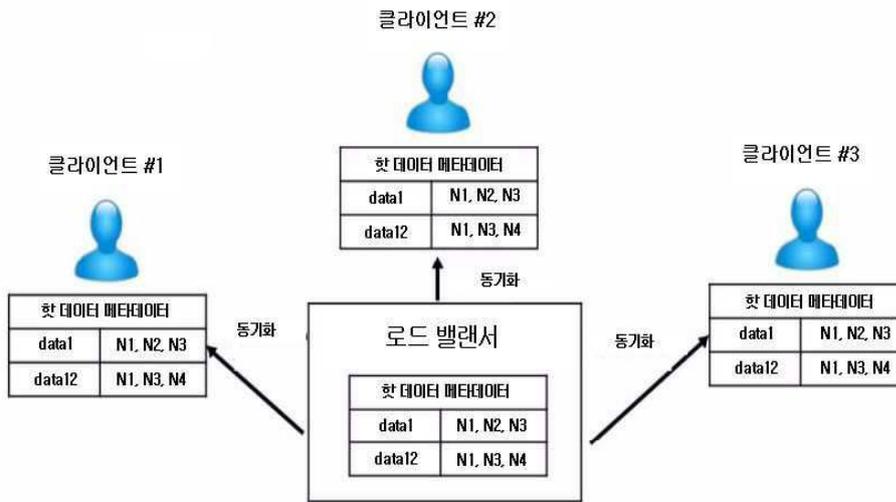
도면2



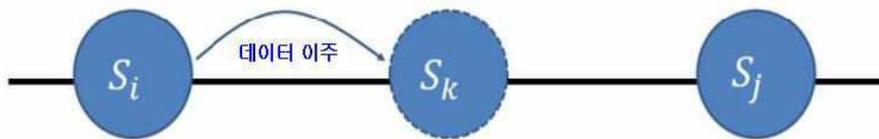
도면3



도면4



도면5



도면6

