



(19) 대한민국특허청(KR)  
 (12) 등록특허공보(B1)

(45) 공고일자 2019년06월27일  
 (11) 등록번호 10-1993915  
 (24) 등록일자 2019년06월21일

- (51) 국제특허분류(Int. Cl.)  
*G06F 9/50* (2018.01) *G06F 3/06* (2006.01)  
*G06F 9/455* (2018.01)
- (52) CPC특허분류  
*G06F 9/5088* (2013.01)  
*G06F 3/0604* (2013.01)
- (21) 출원번호 10-2018-7026783
- (22) 출원일자(국제) 2016년12월02일  
 심사청구일자 2018년10월11일
- (85) 번역문제출일자 2018년09월14일
- (65) 공개번호 10-2018-0117641
- (43) 공개일자 2018년10월29일
- (86) 국제출원번호 PCT/US2016/064738
- (87) 국제공개번호 WO 2017/160359  
 국제공개일자 2017년09월21일
- (30) 우선권주장  
 15/071,852 2016년03월16일 미국(US)
- (56) 선행기술조사문현  
 US20140365738 A1  
 US20130254368 A1  
 US20150052282 A1  
 US8924675 B1

(73) 특허권자  
 구글 엘엘씨  
 미국 캘리포니아 마운틴 뷰 엠피시어터 파크웨이  
 1600 (우:94043)

(72) 발명자  
 샌더슨, 타일러  
 미국 94043 캘리포니아 마운틴 뷰 엠피시어터 파  
 크웨이 1600

(74) 대리인  
 특허법인 남엔남

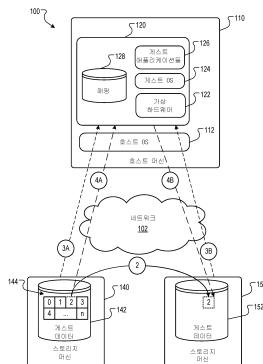
전체 청구항 수 : 총 21 항

심사관 : 유진태

(54) 발명의 명칭 원격으로 액세스되는 데이터의 효율적인 라이브-이송

**(57) 요약**

제1 스토리지 시스템으로부터 제2 스토리지 시스템으로 데이터를 이송하기 위한, 컴퓨터 스토리지 매체 상에 인코딩된 컴퓨터 프로그램들을 비롯한 방법들, 시스템들, 및 장치가 개시된다. 그 데이터는, 제1 스토리지 시스템 및 제2 스토리지 시스템과 별개인 컴퓨터 시스템 상에서 동작하는 프로세스(예컨대, 가상 머신, 애플리케이션, 또는 어떤 다른 프로세스)에 대한 것이다. 그 데이터는, 서로에 대해 각자 배타적인 데이터 서브셋트들에 따라 저장된다. 각각의 데이터 세트가 이송될 때, 프로세스에 의한 데이터 세트로의 액세스는 방지되지만, 나머지 데이터 세트들에 대한 액세스는 영향받지 않는다. 일단 데이터 이송이 완료되면, 프로세스에 의한 데이터 세트로의 액세스는 복원된다.

**대 표 도 - 도1**

(52) CPC특허분류

*G06F 3/0647* (2013.01)  
*G06F 3/0664* (2013.01)  
*G06F 3/0683* (2013.01)  
*G06F 9/45558* (2013.01)  
*G06F 2009/4557* (2013.01)  
*G06F 2009/45583* (2013.01)  
*G06F 2009/45595* (2013.01)

---

## 명세서

### 청구범위

#### 청구항 1

복수의 물리적 머신들을 포함하는 데이터 프로세싱 시스템에 의해 수행되는 방법으로서,

제1 물리적 머신과 별개인 제2 물리적 머신 상에서 실행되는 데이터 프로세싱 프로세스에 대한 데이터를 상기 제1 물리적 머신에 저장하는 단계 – 상기 데이터를 저장하는 단계는 서로에 대해 각자 배타적인 복수의 데이터 서브세트들에 따라 상기 데이터를 저장하는 단계를 포함함 –;

각각의 데이터 서브세트에 대해, 상기 데이터 프로세싱 프로세스에 의해, 매핑에서, 상기 제1 물리적 머신 상의 데이터 서브세트에 대한 어드레스 범위를 상기 데이터 프로세싱 프로세스에 대한 개개의 논리 어드레스 범위로 논리적으로 매핑하는 단계;

논리 어드레싱에 따라, 상기 데이터 프로세싱 프로세스에 의한 상기 데이터에 대한 판독 및 기록 액세스를 인에 이를 링하는 단계;

상기 데이터 프로세싱 프로세스에 대한, 상기 제1 물리적 머신 상에 저장된 상기 데이터가 상기 제1 물리적 머신 및 상기 제2 물리적 머신과 별개인 제3 물리적 머신에 이송(migrate)되어야 한다고 결정하고, 응답으로, 제1 데이터 서브세트들의 데이터를 별개로 이송하는 단계

를 포함하고,

각각의 제1 데이터 서브세트의 이송을,

상기 제1 물리적 머신 및 상기 제2 물리적 머신과 별개인 상기 제3 물리적 머신에 이송되어야 할, 상기 제1 물리적 머신 상에 저장된 데이터 서브세트를 제1 데이터 서브세트로서 선택하는 단계, 및 응답으로,

상기 데이터의 다른 데이터 서브세트들에 대한 상기 데이터 프로세싱 프로세스에 의한 판독 및 기록 액세스를 유지하면서, 상기 제1 데이터 서브세트에 대한 상기 데이터 프로세싱 프로세스에 의한 판독 및 기록 액세스를 차단하는 단계;

상기 데이터 서브세트를 상기 제3 물리적 머신 상에 저장하기 위해, 상기 제1 물리적 머신으로부터 상기 제3 물리적 머신으로 상기 제1 데이터 서브세트를 이송하는 단계;

상기 데이터 프로세싱 프로세스에 의해 상기 제3 물리적 머신 상의 상기 제1 데이터 서브세트에 대한 어드레스 범위를 상기 데이터 프로세싱 프로세스에 대한 개개의 논리 어드레스 범위로 논리적으로 매핑함으로써, 상기 매핑을 업데이트하는 단계

를 포함하는,

방법.

#### 청구항 2

제1항에 있어서,

상기 데이터 프로세싱 프로세스는 가상 머신에 의해 수행되는,

방법.

#### 청구항 3

제2항에 있어서,

상기 제1 물리적 머신 상에 저장된 상기 데이터가 상기 제3 물리적 머신에 이송되어야 한다는 결정은 상기 제1 물리적 머신에 의해 수행되며; 그리고

다른 데이터 서브세트들에 대한 상기 가상 머신에 의한 판독 및 기록 액세스를 유지하면서, 상기 제1 데이터 서

브세트에 대한 상기 가상 머신에 의한 판독 및 기록 액세스를 차단하는 단계는, 상기 제1 물리적 머신에 의해, 상기 제1 데이터 서브세트에 대한 상기 가상 머신에 의한 판독 및 기록 액세스를 차단하는 단계를 포함하는, 방법.

#### 청구항 4

제3항에 있어서,

상기 제1 물리적 머신에 의해, 상기 제3 물리적 머신으로의 상기 제1 데이터 서브세트의 이송(migration)의 통지를 상기 가상 머신에 제공하는 단계

를 더 포함하며; 그리고

상기 제3 물리적 머신 상의 상기 제1 데이터 서브세트에 대한 어드레스 범위를 상기 가상 머신에 대한 개개의 논리 어드레스 범위로 논리적으로 매핑함으로써 상기 매핑을 업데이트하는 단계는, 상기 가상 머신이 상기 제1 물리적 머신에 의해 제공된 상기 통지를 수신하는 것에 대한 응답으로 수행되는,

방법.

#### 청구항 5

제4항에 있어서,

상기 통지는, 상기 제1 데이터 서브세트에 액세스하지 않도록 상기 가상 머신에 명령하는,

방법.

#### 청구항 6

제4항에 있어서,

상기 데이터 서브세트를 상기 제3 물리적 머신 상에 저장하기 위해, 상기 제1 물리적 머신으로부터 상기 제3 물리적 머신으로 상기 제1 데이터 서브세트를 이송하는 단계는, 상기 제1 물리적 머신으로부터 바로 상기 제3 물리적 머신으로 상기 제1 데이터 서브세트를 전송하는 단계를 포함하는,

방법.

#### 청구항 7

제3항에 있어서,

상기 제3 물리적 머신에 의해, 상기 제3 물리적 머신으로의 상기 제1 데이터 서브세트의 이송의 통지를 상기 가상 머신에 제공하는 단계

를 더 포함하며; 그리고

상기 제3 물리적 머신 상의 상기 제1 데이터 서브세트에 대한 어드레스 범위를 상기 가상 머신에 대한 개개의 논리 어드레스 범위로 논리적으로 매핑함으로써 상기 매핑을 업데이트하는 단계는, 상기 가상 머신이 상기 통지를 수신하는 것에 대한 응답으로 수행되는,

방법.

#### 청구항 8

제2항에 있어서,

상기 제3 물리적 머신에 이송되어야 할, 상기 제1 물리적 머신 상에 저장된 상기 데이터 서브세트를 상기 제1 데이터 서브세트로서 선택하는 단계는 상기 가상 머신에 의해 수행되며; 그리고

상기 데이터 서브세트를 상기 제3 물리적 머신 상에 저장하기 위해, 상기 제1 물리적 머신으로부터 상기 제3 물리적 머신으로 상기 제1 데이터 서브세트를 이송하는 단계는:

상기 가상 머신에 의해, 상기 제1 데이터 서브세트를 상기 가상 머신에 전송하도록 상기 제1 물리적 머

신에 명령하는 단계; 및

상기 가상 머신에 의해, 상기 제1 물리적 머신으로부터 수신된 상기 제1 데이터 서브세트를 상기 제3 물리적 머신에 전송하는 단계

를 포함하는,

방법.

#### 청구항 9

제8항에 있어서,

상기 제1 물리적 머신 및 제3 물리적 머신들은 상기 가상 머신의 어드레스에 대한 참조 없이 상기 데이터 서브세트들을 저장하는,

방법.

#### 청구항 10

제2항에 있어서,

상기 제1 물리적 머신으로부터 상기 제3 물리적 머신으로의 상기 제1 데이터 서브세트의 이송 후에, 상기 가상 머신에 대해 상기 제1 데이터 서브세트에 대한 판독 및 기록 액세스를 복원시키는 단계

를 더 포함하는,

방법.

#### 청구항 11

제2항에 있어서,

상기 제3 물리적 머신에 이송되어야 할, 상기 제1 물리적 머신 상에 저장된 상기 데이터 서브세트를 상기 제1 데이터 서브세트로서 선택하는 단계는, 상기 제1 물리적 머신, 상기 가상 머신, 및 상기 제3 물리적 머신의 외부에 있는 프로세스에 의해 수행되며; 그리고

상기 데이터 서브세트를 상기 제3 물리적 머신 상에 저장하기 위해, 상기 제1 물리적 머신으로부터 상기 제3 물리적 머신으로 상기 제1 데이터 서브세트를 이송하는 단계는:

상기 가상 머신에 의해, 상기 제1 데이터 서브세트를 상기 가상 머신에 전송하도록 상기 제1 물리적 머신에 명령하는 단계; 및

상기 가상 머신에 의해, 상기 제1 물리적 머신으로부터 수신된 상기 제1 데이터 서브세트를 상기 제3 물리적 머신에 전송하는 단계

를 포함하는,

방법.

#### 청구항 12

제1항에 있어서,

상기 데이터 서브세트들은 물리 어드레스 범위에 논리적으로 매핑된 데이터 청크(chunk)들인,

방법.

#### 청구항 13

제1항에 있어서,

상기 데이터 서브세트들은 데이터 블록들인,

방법.

**청구항 14**

시스템으로서,

서로에 대해 각자 베타적인 복수의 데이터 서브세트들에 따라 데이터를 저장하는 제1 물리적 머신; 및

상기 제1 물리적 머신과 별개인 제2 물리적 머신

을 포함하며,

상기 제2 물리적 머신은 상기 제2 물리적 머신 상의 가상 머신과 별개이며, 상기 가상 머신은 상기 제1 물리적 머신 상에 저장된 데이터에 대한 관독 및 기록 액세스를 가지며, 상기 제2 물리적 머신은, 각각의 데이터 서브세트에 대해, 매핑에서, 상기 제1 물리적 머신 상의 데이터 서브세트에 대한 어드레스 범위를 상기 가상 머신에 대한 개개의 논리 어드레스 범위로 논리적으로 매핑하며;

상기 시스템은, 상기 가상 머신에 대한, 상기 제1 물리적 머신 상에 저장된 상기 데이터가 상기 제1 물리적 머신 및 상기 제2 물리적 머신과 별개인 제3 물리적 머신에 이송되어야 한다고 결정하고, 응답으로, 상기 데이터를 제1 데이터 서브세트별로 별개로 이송하고,

각각의 제1 데이터 서브세트의 이송은: 상기 제1 물리적 머신 및 상기 제2 물리적 머신과 별개인 제3 물리적 머신에 이송되어야 할, 상기 제1 물리적 머신 상에 저장된 데이터 서브세트를 제1 데이터 서브세트로서 선택하는 것을 포함하고,

상기 제1 데이터 서브세트는 상기 제1 물리적 머신으로부터 상기 제3 물리적 머신으로 이송되고;

상기 데이터의 다른 데이터 서브세트들에 대한 상기 가상 머신에 의한 관독 및 기록 액세스가 유지되면서, 상기 이송 동안, 상기 가상 머신에 대해 상기 제1 데이터 서브세트에 대한 관독 및 기록 액세스가 차단되며; 그리고

상기 제3 물리적 머신 상의 상기 제1 데이터 서브세트에 대한 어드레스 범위를 상기 가상 머신에 대한 개개의 논리 어드레스 범위로 논리적으로 매핑하도록, 상기 가상 머신에 의해 상기 매핑이 업데이트되며; 그리고

상기 제1 물리적 머신으로부터 상기 제3 물리적 머신으로의 상기 제1 데이터 서브세트의 이송 후에, 상기 가상 머신에 대해 상기 제1 데이터 서브세트에 대한 관독 및 기록 액세스가 복원되는,

시스템.

**청구항 15**

제14항에 있어서,

상기 제3 물리적 머신에 이송되어야 할, 상기 제1 물리적 머신 상에 저장된 상기 데이터 서브세트를 상기 제1 데이터 서브세트로서 선택하는 것은, 상기 제1 물리적 머신에 의해 수행되며; 그리고

상기 제1 데이터 서브세트에 대한 상기 가상 머신에 의한 관독 및 기록 액세스를 차단하는 것은, 상기 제1 물리적 머신에 의해 수행되는,

시스템.

**청구항 16**

제14항에 있어서,

상기 제1 물리적 머신은 상기 제3 물리적 머신으로의 상기 제1 데이터 서브세트의 이송의 통지를 상기 가상 머신에 제공하며; 그리고

상기 제3 물리적 머신 상의 상기 제1 데이터 서브세트에 대한 어드레스 범위를 상기 가상 머신에 대한 개개의 논리 어드레스 범위로 논리적으로 매핑하도록 상기 매핑을 업데이트하는 것은, 상기 가상 머신에 의해 상기 통지를 수신하는 것에 대한 응답으로 수행되는,

시스템.

**청구항 17**

제16항에 있어서,

상기 통지는, 상기 제1 데이터 서브세트에 액세스하지 않도록 상기 가상 머신에 명령하는, 시스템.

### 청구항 18

제14항에 있어서,

상기 제1 물리적 머신으로부터 상기 제3 물리적 머신으로 상기 제1 데이터 서브세트를 이송하는 것은, 상기 제1 물리적 머신으로부터 바로 상기 제3 물리적 머신으로 상기 제1 데이터 서브세트를 전송하는 것을 포함하는, 시스템.

### 청구항 19

제14항에 있어서,

상기 제3 물리적 머신은 상기 제3 물리적 머신으로의 상기 제1 데이터 서브세트의 이송의 통지를 상기 가상 머신에 제공하며; 그리고

상기 제3 물리적 머신 상의 상기 제1 데이터 서브세트에 대한 어드레스 범위를 상기 가상 머신에 대한 개개의 논리 어드레스 범위로 논리적으로 매핑하도록 상기 매핑을 업데이트하는 것은, 상기 가상 머신에 의해 상기 통지를 수신하는 것에 대한 응답으로 수행되는,

시스템.

### 청구항 20

제14항에 있어서,

상기 제3 물리적 머신에 이송되어야 할, 상기 제1 물리적 머신 상에 저장된 상기 데이터 서브세트를 상기 제1 데이터 서브세트로서 선택하는 것은, 상기 가상 머신에 의해 수행되며; 그리고

상기 제1 물리적 머신으로부터 상기 제3 물리적 머신으로 상기 제1 데이터 서브세트를 이송하는 것은:

상기 가상 머신에 의해, 상기 제1 데이터 서브세트를 상기 가상 머신에 전송하도록 상기 제1 물리적 머신에 명령하는 것; 및

상기 가상 머신에 의해, 상기 제1 물리적 머신으로부터 수신된 상기 제1 데이터 서브세트를 상기 제3 물리적 머신에 전송하는 것

을 포함하는,

시스템.

### 청구항 21

제19항에 있어서,

상기 제1 물리적 머신 및 제3 물리적 머신들은 상기 가상 머신의 어드레스에 대한 참조 없이 상기 데이터 서브세트들을 저장하는,

시스템.

### 청구항 22

삭제

## 발명의 설명

## 기술 분야

### 배경 기술

[0001]

[001] 클라우드 컴퓨팅은, 데이터 센터들 또는 "서버 팜들"에 하우징된 서버들의 집합들이 원격 최종 사용자들에게 필요할 때 계산 자원들 및 데이터 스토리지를 제공하는 네트워크-기반 컴퓨팅을 지칭한다. 일부 클라우드 컴퓨팅 서비스들은 소프트웨어 애플리케이션들, 이를테면 워드 프로세서들 및 다른 흔히 사용되는 애플리케이션들로의 액세스를 최종 사용자들에게 제공하며, 이러한 최종 사용자들은 웹 브라우저들 또는 다른 클라이언트-측 소프트웨어를 통해 애플리케이션들과 인터페이싱한다. 사용자들의 전자 데이터 파일들은 대개, 사용자들의 컴퓨팅 디바이스들 상에 저장되는 것이 아니라, 서버 팜에 저장된다. 소프트웨어 애플리케이션들 및 사용자 데이터를 서버 팜 상에 유지하는 것은 최종 사용자 컴퓨팅 디바이스들의 관리를 단순화한다. 일부 클라우드 컴퓨팅 서비스들은 최종 사용자들이 가상 머신들에서 소프트웨어 애플리케이션들을 실행할 수 있게 한다. 공중 클라우드 컴퓨팅 환경에서는, 다수의 사용자들이 가상 머신(VM; virtual machine)들을 론치(launch)할 수 있다.

[0002]

[002] 종종, 특정 가상 머신에 대한 데이터는, 이 가상 머신이 인스턴스화되어 있는 물리적 머신과 별개인 하나 이상의 물리적 머신들 상에 저장된다. 다양한 이유들로 –부하 공유, 서버 유지보수 등–, 특정 물리적 머신 상에 저장된 데이터의 일부 또는 전부가 다른 물리적 머신에 이송(migrate)될 수 있다. 데이터의 라이브-이송(live-migration)은, 가상 머신(또는 대안적으로, 비-가상 프로세싱 엔티티)이 데이터에 대한 임의적인 판독들 및 기록들을 수행하는 동안, 하나의 물리적 머신에서 다른 물리적 머신으로 이 데이터를 이동시키는 프로세스이다.

### 발명의 내용

[0003]

[003] 본 명세서는 데이터의 라이브 이송에 관한 것이다.

[0004]

[004] 본 문서는, 다른 물리적 머신으로부터 원격으로 데이터에 액세스하고 있는 제3자 프로세싱 디바이스(예컨대, 가상 머신 또는 애플리케이션)에 데이터로의 일관된 그리고 고성능의 액세스를 제공하면서, 스토리지 컴퓨터에서 다른 스토리지 컴퓨터로 이 데이터를 이동시키기 위한 체계적인 방법 및 시스템을 설명한다.

[0005]

[005] 일반적으로, 본 명세서에서 설명된 발명의 요지의 하나의 혁신적인 양상은 방법들에서 구현될 수 있으며, 이러한 방법들은, 제1 물리적 머신과 별개인 제2 물리적 머신 상에서 실행되는 데이터 프로세싱 프로세스에 대한 데이터를 제1 물리적 머신에 저장하는 동작 –데이터를 저장하는 동작은 서로에 대해 각자 배타적인 복수의 데이터 서브세트들에 따라 데이터를 저장하는 동작을 포함함–; 각각의 데이터 서브세트에 대해, 매핑에서, 데이터 프로세싱 프로세스가 제1 물리적 머신 상의 데이터 서브세트에 대한 어드레스 범위를 데이터 프로세싱 프로세스에 대한 개개의 논리 어드레스 범위로 논리적으로 매핑하는 동작; 논리 어드레싱에 따라, 데이터 프로세싱 프로세스에 의한 데이터에 대한 판독 및 기록 액세스를 인에이블링하는 동작; 제1 물리적 머신 상에 저장된 제1 데이터 서브세트가 제1 물리적 머신 및 제2 물리적 머신과 별개인 제3 물리적 머신에 이송되어야 한다고 결정하는 동작; 및 응답으로: 다른 데이터 서브세트들에 대한 데이터 프로세싱 프로세스에 의한 판독 및 기록 액세스를 유지하면서, 제1 데이터 서브세트에 대한 데이터 프로세싱 프로세스에 의한 판독 및 기록 액세스를 차단하는 동작, 데이터 서브세트를 제3 물리적 머신 상에 저장하기 위해, 제1 물리적 머신으로부터 제3 물리적 머신으로 제1 데이터 서브세트를 이송하는 동작, 및 데이터 프로세싱 프로세스가 제3 물리적 머신 상의 제1 데이터 서브세트에 대한 어드레스 범위를 데이터 프로세싱 프로세스에 대한 개개의 논리 어드레스 범위로 논리적으로 매핑함으로써, 매핑을 업데이트하는 동작을 포함한다. 이 양상의 다른 실시예들은 대응하는 시스템들, 장치, 및 컴퓨터 스토리지 디바이스들 상에 인코딩된, 방법들의 동작들을 수행하도록 구성된 컴퓨터 프로그램들을 포함한다.

[0006]

[006] 일반적으로, 본 명세서에서 설명된 발명의 요지의 다른 양상은 시스템에서 구현될 수 있으며, 이 시스템은, 서로에 대해 각자 배타적인 복수의 데이터 서브세트들에 따라 데이터를 저장하는 제1 물리적 머신; 및 제1 물리적 머신과 별개인 제2 물리적 머신을 포함하며, 제2 물리적 머신은 제2 물리적 머신 상의 가상 머신과 별개이며, 가상 머신은 제1 물리적 머신 상에 저장된 데이터에 대한 판독 및 기록 액세스를 가지며, 제2 물리적 머신은, 각각의 데이터 서브세트에 대해, 매핑에서, 제1 물리적 머신 상의 데이터 서브세트에 대한 어드레스 범위를 가상 머신에 대한 개개의 논리 어드레스 범위로 논리적으로 매핑하며; 제1 물리적 머신 상에 저장된 제1

데이터 서브세트가 제1 물리적 머신 및 제2 물리적 머신과 별개인 제3 물리적 머신에 이송되어야 한다는 결정에 대한 응답으로: 제1 물리적 머신으로부터 제3 물리적 머신으로 제1 데이터 서브세트가 이송되고; 다른 데이터 서브세트들에 대한 가상 머신에 의한 판독 및 기록 액세스가 유지되면서, 이송 동안, 가상 머신에 대해 제1 데이터 서브세트에 대한 판독 및 기록 액세스가 차단되며; 그리고 제3 물리적 머신 상의 제1 데이터 서브세트에 대한 어드레스 범위를 가상 머신에 대한 개개의 논리 어드레스 범위로 논리적으로 매핑하기 위해, 가상 머신에 의해 매핑이 업데이트되며; 그리고 제1 물리적 머신으로부터 제3 물리적 머신으로의 제1 데이터 서브세트의 이송 후에, 가상 머신에 대해 제1 데이터 서브세트에 대한 판독 및 기록 액세스가 복원된다. 이 양상의 다른 실시예들은 대응하는 방법들, 장치, 및 컴퓨터 스토리지 디바이스들 상에 인코딩된, 방법들의 동작들을 수행하도록 구성된 컴퓨터 프로그램들을 포함한다.

[0007] 본 명세서에서 설명된 발명의 요지의 특정 실시예들은, 다음의 장점들 중 하나 이상을 실현하기 위하여 구현될 수 있다. 방법은, 라이브-이송의 다른 방법들, 이를테면 사전-복사(pre-copy) 및 사후-복사(post-copy) 라이브 이송보다 더 우수한 성능 보증들을 가상 머신에 제공한다. 예컨대, 사전-복사 라이브-이송은 온전한 데이터 어드레스 공간을 온전한 이송 유닛으로서 간주하며, 따라서 전체 어드레스 공간이 이송의 발신 측 상에 완전히 존재하거나 또는 이송의 목적지 측 상에 완전히 존재한다. 이송이 진행중인 동안 데이터로의 "라이브" 액세스를 제공하기 위해, 가상 머신은 데이터를 판독/기록하는 것을 계속하도록 허용되지만, 변화들이 라이브 이송의 수신 물리적 머신(들)에 재전송될 수 있도록, 기록들은 추적된다. 이러한 재전송은 더 많은 판독 대역폭, 네트워크 대역폭, CPU 프로세싱, 및 시간을 취한다. 게다가, 바쁜 가상 머신은 통상적으로, 변화들이 네트워크를 통해 목적지에 전송될 수 있는 것보다 더 빨리 데이터를 변화시키고 있을 것이며; 그러한 상황들에서, 가상 머신 액세스 레이트가 늦춰져서, 성능 저하가 야기될 것임에 틀림없다.

[0008] 사후-복사 라이브-이송에 대해, 가상 머신은 목적지 물리적 머신에 대해 통보받으며, 데이터를 목적지 물리적 머신에 요청한다. 목적지 물리적 머신은, 데이터가 목적지 물리적 머신 상에 저장되어 있으면 데이터를 제공하며; 그렇지 않으면, 목적지 물리적 머신은 데이터가 이송되고 있는 발신 물리적 머신으로부터 데이터를 페치(fetch)하고, 그런 다음, 데이터를 제공한다. 데이터가 발신 물리적 머신으로부터 페치되어야 할 때, 가상 머신은 증가된 레이턴시를 경험한다. 목적지 물리적 머신으로부터 발신 물리적 머신으로의 많은 액세스들에 대해, 상당한 전체 대역폭 성능 저하가 있다.

[0009] 그러나, 아래에서 설명된 방법들 및 시스템들은 이를 동작 특성들의 일부 또는 전부를 극복하여서, 데이터 스토리지 및 관리의 기술 영역의 개선을 야기한다. X MB의 데이터의 청크(chunk) 또는 데이터의 페이지 등일 수 있는 데이터 서브세트들의 이송 데이터를 프로세싱함으로써, 이송은 사전-복사 라이브-이송보다 훨씬 더 과립성(granular)이다. 이송을 겪고 있는 데이터 서브세트는 가상 머신에 의해 액세스되는 것이 방지된다. 따라서, 기록들의 추적이 수행될 필요가 없다. 일단 데이터 서브세트가 이송되면, 이 데이터 서브세트는 재전송될 필요가 없는데, 그 이유는 모든 미래의 액세스들이 목적지 측으로 바로 가기 때문이다.

[0010] 가상 머신이 이송되고 있는 데이터 서브세트에 액세스하고 있지 않다면, 이 가상 머신은 성능에 대한 어떤 영향도 사실상 경험하지 않는다. 데이터 서브세트 크기가 감소함에 따라, 이송하기 위한 데이터 서브세트에 대해 대기하면서 가상 머신이 차단되어야 하는 시간의 양은 감소한다. 반대로, 데이터 서브세트 크기가 증가함에 따라, 스토리지 매핑을 위한 메타데이터의 양은 감소한다. 따라서, 대기 시간 대 매핑 유지보수의 트레이드-오프(trade-off)를 선택적으로 평가함으로써, 시스템 관리는 특정 애플리케이션에 데이터 서브세트 크기를 맞출 수 있으며, 이는 이 애플리케이션에 대한 개선된 이송 성능 동작을 야기한다.

[0011] 아래에서 설명된 이송 기법들이 오버헤드에 대해 일부 대역폭을 활용하지만, 활용되는 양은, 사전-복사 또는 사후-복사 이송에 의해 활용되는 대역폭과 비교하여 비교적 적다. 이는 여전히 데이터 이송의 기술 분야에 대한 또 다른 개선이다.

[0012] 이송을 겪고 있는 데이터 서브세트에 대한 판독 및 기록 액세스가 차단되기 때문에, 이 데이터 서브세트에 대한 변화들을 추적하기 위한 오버헤드 메커니즘이 필요하지도, 이송의 목적을 위한, 데이터 서브세트로의 액세스들을 가상 머신에 구체적으로 명령할 필요도 없다. 예컨대, 데이터 스토어가 디스크이고, 이송의 목적을 위한, 일 위치에 대한 미처리 판독이 있는 동시에 가상 머신이 동일한 위치로의 기록을 수행한다면, 판독 액세스의 결과는 정의되지 않는다. 본원의 시스템들 및 방법들은, 이송되고 있는 특정 구역에 가상 머신이 액세스하는 것을 차단함으로써, 동일한 위치로의 동시 액세스를 방지한다.

[0013] 아래에서 설명된 라이브 이송은, 진행을 잃지 않으면서 언제라도 정지될 수 있다. 이는, 정지되면 완전히 다시 시작되어야 하는 사전-복사 라이브 이송과 대조적이며, 그리고 또한, 가상 머신을 위한 액세스들이

목적지 측으로 스위칭된 후에는 정지될 수 없는 사후-복사 라이브 이송과 대조적이다.

[0014] 본 명세서에서 설명된 발명의 요지의 하나 이상의 실시예들의 세부사항들은 아래의 상세한 설명 및 첨부된 도면들에서 제시된다. 발명의 요지의 다른 특징들, 양상들, 및 장점들은 아래의 상세한 설명, 도면들, 및 청구항들로부터 자명해질 것이다.

### 도면의 간단한 설명

[0015] 도 1은 데이터가 라이브 이송을 겪을 수 있는 클라우드-기반 환경의 블록 다이어그램이다.

[0016] 도 2는 데이터의 라이브 이송을 위한 예시적인 프로세스의 흐름도이다.

[0017] 도 3은 제1 물리적 머신으로부터 제3 물리적 머신으로 바로 데이터 서브셋들이 이송되는, 데이터의 라이브 이송을 위한 예시적인 프로세스의 흐름도이다.

[0018] 도 4는 제1 물리적 머신으로부터 제3 물리적 머신으로 그리고 가상 머신이 구현되어 있는 물리적 머신을 통해 데이터 서브셋들이 이송되는, 데이터의 라이브 이송을 위한 예시적인 프로세스의 흐름도이다.

[0019] 다양한 도면들에서 유사한 참조 번호들 및 표기들은 유사한 엘리먼트들을 표시한다.

### 발명을 실시하기 위한 구체적인 내용

[0020] 개요

[0021] 제1 물리적 머신은, 서로에 대해 각자 배타적인 다수의 데이터 서브셋들에 따라 데이터를 저장한다. 제1 물리적 머신과 별개인 제2 물리적 머신 상에서 실행되는 데이터 프로세싱 프로세스, 이를테면 가상 머신에 대한 데이터가 저장된다. 데이터 프로세싱 프로세스는 제1 물리적 머신 상에 저장된 데이터에 대한 판독 및 기록 액세스를 가지며, 각각의 데이터 서브셋에 대해, 매핑에서, 제1 물리적 머신 상의 데이터 서브셋에 대한 어드레스 범위를 데이터 프로세싱 프로세스에 대한 개개의 논리 어드레스 범위로 논리적으로 매핑한다.

[0022] 제1 물리적 머신으로부터의 데이터가 제3 물리적 머신에 이송되어야 할 때, 다음의 단계들이 취해진다. 데이터는 데이터 서브셋별로(in data subsets) 이송된다. 이송을 겪고 있는(예컨대, 제1 물리적 머신으로부터 제3 물리적 머신으로 "비행 중(in flight)"인 과정에 있는) 각각의 데이터 서브셋에 대해, 데이터 서브셋에 대한 데이터 프로세싱 프로세스에 의한 판독 및 기록 액세스는 차단된다. 그러나, 데이터 이송을 겪고 있지 않은 다른 데이터 서브셋들에 대한 데이터 프로세싱 프로세스에 의한 판독 및 기록 액세스는 유지된다. 이러한 방식으로, 데이터 프로세싱 프로세스는, 제1 물리적 머신에 저장되는 데이터 중 많은 데이터에 계속해서 액세스할 수 있다. 제1 물리적 머신으로부터 제3 물리적 머신으로의 특정 데이터 서브셋의 이송에 대한 응답으로, 데이터 프로세싱 프로세스에 의해 매핑이 업데이트된다. 특히, 제3 물리적 머신 상의 특정 데이터 서브셋에 대한 어드레스 범위가 데이터 프로세싱 프로세스에 대한 개개의 논리 어드레스 범위에 매핑되며, 따라서 특정 데이터 서브셋의 이송-전 매핑은 업데이트된 매핑에 의해 대체된다. 제1 물리적 머신으로부터 제3 물리적 머신으로의 제1 데이터 서브셋의 이송 후에, 데이터 프로세싱 프로세스에 대해 제1 데이터 서브셋에 대한 판독 및 기록 액세스가 복원된다. 그후에, 이송된 데이터 서브셋에 저장된 데이터가 데이터 프로세싱 프로세스에 의해 요구될 때, 데이터 프로세싱 프로세스는 제3 물리적 머신에 액세스한다.

[0023] 위의 프로세스에 대한 몇몇 변형들이 시스템 요건들에 따라 유리할 수 있다. 예컨대, 데이터 프로세싱 프로세스는 데이터 서브셋의 이송에 대해 통지받지 않을 수 있다. 데이터 프로세싱 프로세스가 제1 물리적 머신으로부터의 데이터 서브셋에 저장된 데이터로의 액세스를 요청하면, 데이터 프로세싱 프로세스는, 데이터 서브셋이 이제 제3 물리적 머신 상에 저장되어 있다(또는 현재 제3 물리적 머신으로의 이송을 겪고 있다)는 것을 데이터 프로세싱 프로세스에 통보하는 응답을 수신할 것이다. 데이터 서브셋이 현재 이송을 겪고 있다면, 제3 물리적 머신은, 이송이 완료될 때까지 데이터 서브셋로의 액세스를 차단할 수 있다.

[0024] 대안적으로, 데이터 프로세싱 프로세스는 이송에 대해 사전에 통보받을 수 있으며, 이송의 예상 완료 시간을 계산할 수 있다. 그런 다음, 데이터 프로세싱 프로세스는, 예상 완료 시간까지, 데이터를 요청하려고 대기할 수 있다.

[0025] 제1 물리적 머신으로부터 제3 물리적 머신으로 바로 데이터 서브셋이 전송될 수 있으며, 그런 다음, 이송이 완료될 때, 제1 물리적 머신 또는 제3 물리적 머신 중 하나가 데이터 서브셋의 새로운 위치에 대해 데이터 프로세싱 프로세스에 통보할 것이다. 그러나, 대안적으로, 데이터 서브셋은 중개자로서의 역할을 하는

데이터 프로세싱 프로세스를 통해 전달될 수 있으며, 데이터 프로세싱 프로세스는 이송을 책임진다. 이는 데이터 프로세싱 프로세스가 거의 실시간으로, 데이터가 상주하는 곳에 관하여 최신이 될 수 있게 한다.

[0022] 또 다른 구현에서, 데이터 프로세싱 프로세스가 이송을 제어하지만, 데이터 서브세트는 하나의 스토리지 머신으로부터 다른 스토리지 머신으로 바로 전송된다. 예컨대, 가상 머신은 "전달 어드레스 범위" 메시지를 제1 스토리지 머신에 전송하며, 이 전달 어드레스 범위 메시지는, 전달 어드레스 범위에 의해 특정된 특정 어드레스 범위를 판독하여 그 데이터를 제2 스토리지 머신에 대한 다른 어드레스 범위에 기록하도록 제1 스토리지 머신에 명령한다. 스토리지 머신들은 무상태(stateless)이지만, 가상 머신으로부터 전송된 명령들에 의해, 데이터-서브세트는 제1 스토리지 머신으로부터 제2 스토리지 머신으로 바로 전송된다. 이는 VM을 통해 데이터를 전달하는 것보다 더 적은 데이터 전달을 야기하며, 많은 스토리지 머신들이 수반되면, 더욱 확장가능하다.

[0023] 마지막으로, 데이터 서브세트들은 다양한 상이한 데이터 관리 기법들에 의해 실현될 수 있다. 예컨대, 데이터 서브세트는 페이지 또는 블록 크기에 관련되지 않은, 메모리의 고정 크기, 예컨대 1 MB, 10 MB, 또는 심지어 1 GB일 수 있다. 대안적으로, 데이터 서브세트들은 블록 또는 페이지 레벨로 실현될 수 있으며, 그리고 X 아래의 모든 어드레스들이 제1 물리적 머신 상에 있고, X에 있거나 또는 그 위에 있는 모든 어드레스들이 제3 물리적 머신 상에 있도록, "워터마크"가 사용된다. X의 값은, 데이터가 이송될 때 데이터 프로세싱 프로세스에서 업데이트된다. 이는 데이터 서브세트로의 논리 어드레스의 데이터 프로세싱 프로세스 매핑을 제거할 수 있으며, 대신에, X의 워터마크 값에 따라, 데이터를 저장하는 물리적 머신들을 파티셔닝한다.

[0024] 이들 특징들 및 다른 특징들은 아래에서 더욱 상세히 설명된다.

#### [0025] 예시적인 동작 환경

[0030] 도 1은 데이터가 라이브 이송을 겪을 수 있는 클라우드-기반 환경(100)의 블록 다이어그램이다. 아래의 기록된 설명에서는, 가상 머신의 예시적인 구현이 설명된다. 그러나, 아래에서 설명된 시스템들 및 방법들에 의해, 데이터 프로세싱 프로세스의 일부, 이를테면 클라이언트-기반 애플리케이션에 대한 데이터의 데이터 이송이 클라우드-기반 스토리지를 갖거나, 또는 클라우드 기반 스토리지를 갖는 클라우드-기반 애플리케이션이 또한 용이하게 될 수 있다.

[0031] 클라우드-기반 환경(100)에서, 물리적 머신인 호스트 머신(110)은 하나 이상의 데이터 프로세싱 장치들, 이를테면 랙 장착(rack mounted) 서버들 또는 다른 컴퓨팅 디바이스들을 포함할 수 있다. 또한 물리적 머신들인 스토리지 머신(140 및 150)은, 호스트 머신(110) 상에서 실행되는 데이터 프로세싱 프로세스에 대한 데이터를 저장한다. 스토리지 머신들(140 및 150)은 또한, 하나 이상의 데이터 프로세싱 장치들, 이를테면 랙 장착 서버들 또는 다른 컴퓨팅 디바이스들일 수 있으며, 통상적으로, 네트워크(102)를 통해 통신하는 호스트 머신(110)에 의한 클라우드-기반 액세스를 위한 데이터의 저장을 용이하게 하도록 설계된다.

[0032] 호스트 머신(110)은, 호스트 머신 자원들을 관리하는 호스트 운영체제(112)를 실행한다. 이 예에서, 호스트 운영체제들(112)은, 기초적 호스트 머신 하드웨어를 가상화하고 하나 이상의 가상 머신들(120)의 동시 실행을 관리하는 소프트웨어를 실행한다. 도 1에서 예시된 바와 같이, 호스트 운영체제(112)는 하나의 가상 머신(120)을 관리한다. 일반적으로, 호스트 머신은 더 많은 수량들의 가상 머신들을 관리할 수 있지만, 수량은 호스트 머신의 물리적 자원들에 기반하여 제한될 수 있다. 단순성을 위해, 단 1개의 가상 머신(120)만이 도 1에서 도시된다.

[0033] 가상 머신(120)은 기초적 호스트 머신 하드웨어의 시뮬레이팅된 버전을 사용하며, 이 시뮬레이팅된 버전은 가상 하드웨어(122)로 지칭될 수 있다. 가상 하드웨어(122)에 의해 실행되는 소프트웨어는 게스트 소프트웨어, 예컨대 게스트 운영체제(124) 및 게스트 애플리케이션들(126)로 지칭될 수 있다. 일부 구현들에서, 게스트 소프트웨어는, 자신이 가상 하드웨어에 의해 실행되고 있는지 또는 물리적 호스트 머신에 의해 실행되고 있는지를 결정할 수 없다. 호스트 머신의 마이크로프로세서(들)는, 코드-재작성(code-rewriting), 재컴파일(recompilation), 또는 명령 애플리케이션을 요구하지 않으면서, 게스트 소프트웨어 명령들이 호스트 머신의 마이크로프로세서 상에서 바로 실행될 수 있게 함으로써, 가상 하드웨어가 소프트웨어 애플리케이션들을 효율적으로 실행하는 것을 가능하게 하는 프로세서-레벨 메커니즘들을 포함할 수 있다.

[0034] 호스트 머신(120)에는, 기초적 호스트 운영체제(112)의 가상 메모리로부터의 가상 메모리 페이지들의 세트가 할당되고, 가상 머신 상에서 실행되는 게스트 소프트웨어에 의한 사용을 위한 하나 이상의 가상 디스크 드라이브들로부터의 가상 디스크 블록들이 할당된다. 실제 물리적 스토리지가 호스트 머신(110) 상에 있을 필요는 없으며, 도시된 예에서, 스토리지는 스토리지 머신(140)에 의해 실현된다.

- [0031] [0035] 일부 구현들에서, 가상 디스크 블록들은, 스토리지 머신에 의해 관리되고 네트워크(102)를 통해 호스트 머신(110)과 통신하는 물리적 디스크 드라이브들 상에 할당된다. 가상 머신(120)에는 네트워크 어드레스들이 할당될 수 있으며, 이 네트워크 어드레스들을 통해, 그들의 개개의 프로세스들이 네트워크(102)를 통해 다른 프로세스들과 통신할 수 있다.
- [0032] [0036] 초기에, 가상 머신(120)에 대한 데이터 전부가 스토리지 머신(140) 상에 저장되어 있다고 가정하라. 가상 머신(120)에 대한 이 데이터는 게스트 데이터(142)로 지칭된다. 게스트 데이터(142)는 초기에 단일 물리적 머신 상에 저장될 필요가 없으며, 대신에, 초기에 다수의 스토리지 머신들에 걸쳐 저장될 수 있다. 그러나, 설명의 단순성을 위해, 이 예에 대한 시작 지점은 단일 스토리지 머신이다.
- [0033] [0037] 게스트 데이터(142)는, 서로에 대해 각자 배타적인 다수의 데이터 서브세트들에 따라 저장된다. 도 1에서 도시된 바와 같이, 게스트 데이터(142)는 데이터 서브세트들(144)에 저장되며, 각각의 데이터 서브세트는 예시적으로, 인덱스들(0...n) 중 하나에 의해 인덱싱된다. 매핑 데이터(128)는, 스토리지 머신(140) 상의 데이터 서브세트에 대한 어드레스 범위를 가상 머신(120)에 대한 개개의 논리 어드레스 범위로 논리적으로 매핑한다. 따라서, 매핑 데이터(128)의 사용에 의해, 가상 머신(120)은 논리 어드레스 공간을 특정 물리적 머신 상에 저장된 특정 데이터 서브세트에 매핑할 수 있다. 마지막으로, 매핑 데이터(128)가 가상 머신(120) 내에 있는 것으로서 예시되지만, 매핑 데이터(128)는 또한, 호스트 운영체제(112)에 의해 유지될 수 있다.
- [0034] [0038] 게스트 데이터(142)의 일부 또는 전부가 하나 이상의 다른 스토리지 머신들에 이송되게 할 수 있는 이벤트들이 일어날 수 있다. 그러한 이벤트들은, 스토리지 머신(140)에 저장된 데이터 전부의 이송을 요구하는, 스토리지 머신이 서비스에 대해 오프라인이 되려고 준비하는 것; 스토리지 머신에 저장된 데이터의 적어도 일부분의 이송을 요구하는 부하 밸런싱; 또는 스토리지 머신에 저장된 데이터의 적어도 일부분의 이송을 요구할 수 있는, 서비스 품질 요건들이 충족되지 않은 것을 포함할 수 있다. 아래에서 설명될 바와 같이, 일부 상황들에서는, 이송이 필요할 때를 스토리지 머신이 결정할 수 있으며, 다른 상황들에서는, 이송이 필요할 때를 가상 머신(또는 호스트 머신)이 결정할 수 있다. 위에서 일반적으로 설명된 바와 같이 그리고 아래에서 더욱 상세히 설명될 바와 같이, 또 다른 상황들에서는, 가상 머신, 호스트 머신 및 스토리지 머신들의 외부에 있는 프로세스 또는 엔티티가 또한, 이송이 필요할 때를 결정할 수 있으며, 가상 머신이 이송을 제어할 수 있거나 또는 스토리지 머신들이 이송을 제어할 수 있다.
- [0035] [0039] 도 1에서 참조 콜아웃을 갖는 화살표(2)에 의해 표시되는, 데이터의 라이브 이송을 위한 일반화된 프로세스가 도 2를 참조하여 설명된다. 물리적 스토리지 머신들이 이송을 부분적으로(또는 완전히) 제어하는 하나의 예시적인 프로세스가 도 3을 참조하여 설명되며, 도 1에서 참조 콜아웃들을 갖는 화살표들(3A 및 3B)과 결합하여, 참조 콜아웃을 갖는 화살표(2)에 의해 표시된다. 마지막으로, 가상 머신(또는 호스트 머신)이 이송을 부분적으로(또는 완전히) 제어하는 예시적인 프로세스가 도 4을 참조하여 설명되며, 도 1에서 참조 콜아웃들을 갖는 화살표들(4A 및 4B)과 결합하여, 참조 콜아웃을 갖는 화살표(2)에 의해 표시된다.
- [0036] [0040] 단순성을 위해, 아래에서 설명된 이송 예는 하나의 다른 물리적 머신인 스토리지 머신(150)으로의 데이터의 이송을 상세화할 것이다. 그러나, 게스트 데이터가 다수의 스토리지 머신들 상에 저장되어 있다면, 게스트 데이터는 스토리지 머신들 중 하나로부터, 게스트 데이터의 일부를 현재 저장하고 있는 다른 스토리지 머신 또는 가상 머신(120)에 대한 게스트 데이터를 현재 저장하고 있지 않은 새로운 스토리지 머신으로 이송될 수 있다.
- [0037] [0041] 제1 물리적 머신으로부터 제2 물리적 머신으로의 라이브 이송
- [0038] [0042] 도 2는 데이터의 라이브 이송을 위한 예시적인 프로세스(200)의 흐름도이다. 프로세스(200)는 도 1의 물리적 머신들(110, 140 및 150)에서 구현될 수 있다.
- [0039] [0043] 프로세스(200)는, 제1 물리적 머신과 별개인 제2 물리적 머신 상에서 실행되는 가상 머신에 대한 데이터를 제1 물리적 머신에 저장한다(202). 예컨대, 도 1에서 도시된 바와 같이, 데이터(142)는, 서로에 대해 각자 배타적인 데이터 서브세트들에 따라 저장된다. 데이터(142)의 "데이터 서브세트"는 미리 정의된 데이터 구성, 이를테면 블록, 섹터 또는 페이지일 수 있거나, 또는 임의적으로 정의된 데이터 유닛, 이를테면 1KB, 1MB, 10MB, 또는 심지어 1GB의 데이터의 양일 수 있다. 전자의 경우, 블록 또는 페이지는 가상 머신에 대해 가상적으로 실현되는 크기를 가질 수 있거나, 또는 대안적으로, 사용되는 물리적 하드웨어에 의해 결정되는 물리적 크기를 가질 수 있다.
- [0040] [0044] 프로세스(200)는, 각각의 데이터 서브세트에 대해, 가상 머신에 의해, 제1 물리적 머신 상의 데이터 서

브세트에 대한 어드레스 범위를 가상 머신에 대한 개개의 논리 어드레스 범위로 논리적으로 매핑한다(204). 예컨대, 가상 머신(또는 대안적으로, 호스트 머신)은, 가상 머신의 관점으로 데이터가 상주하는 것처럼 보이는 어드레스를, 데이터가 실제 상주하는 물리적 어드레스로 논리적으로 매핑한다. 제1 머신 상의 데이터 프로세싱 프로세스에서의 논리 어드레스를, 제1 머신과 별개인 제2 머신 상의 물리 어드레스로 매핑할 수 있는 임의의 적절한 어드레스 변환 프로세스가 사용될 수 있다.

- [0041] [0045] 데이터 서브세트가 임의적인 크기를 갖는 후자의 구현의 경우, 크기를 결정할 때 관리자에 의해 몇몇 팩터들이 고려될 수 있다. 데이터 서브세트의 크기가 작을수록, 매핑 데이터(128)가 더 많이 요구될 것이다. 그러나, 데이터 서브세트의 크기가 감소함에 따라, 이송 동안 데이터 서브세트가 비행 중인 시간의 양이 감소하기 때문에, 더 작은 데이터 서브세트들은, 가상 머신(120)이 이송을 겪고 있는 데이터 서브세트에 액세스하려고 시도할 때 일어날 수 있는 판독 및 기록 지연들을 더 적게 야기하는 경향이 있다.
- [0042] [0046] 일부 구현들에서, 가상 머신(120) 또는 호스트 머신(110)은 최대 블록 레이트 임계치와, 이송들에 기인하는 데이터 액세스 블록들의 레이트를 비교할 수 있다. 이 레이트가 최대 블록 레이트 임계치를 초과하면, 데이터 서브세트 크기를 감소시키기 위해, 가상 머신에 의해(또는 대안적으로, 데이터를 저장하는 스토리지 머신들에 의해) 메모리 관리 프로세스가 호출된다. 그후에, 데이터 액세스 블록들의 새로운 레이트가 결정된다. 이 레이트가 최대 블록 레이트 임계치 아래가 될 때까지, 이 프로세스는 계속될 수 있다.
- [0043] [0047] 다른 구현들에서, 가상 머신(120) 또는 호스트 머신(110)은 최대 크기 임계치와, 매핑 데이터(128)의 크기로부터 도출된 크기 메트릭 값을 비교할 수 있다. 이 크기 메트릭 값이 최대 크기 임계치를 초과하면, 논리-물리 매핑을 위해 요구되는 메타데이터의 양이 감소되도록, 가상 머신에 의해(또는 대안적으로, 데이터를 저장하는 스토리지 머신들에 의해) 호출된 메모리 관리 프로세스는 데이터 서브세트 크기를 증가시킬 수 있다.
- [0044] [0048] 또 다른 구현들에서, 데이터 서브세트들의 크기를 관리하기 위해, 데이터 액세스 블록들의 레이트 및 매핑 데이터(128)의 크기로부터 도출된 크기 메트릭 값 둘 모두가 사용될 수 있다. 2개의 성능 고려사항들의 상대적 중요도를 표시하는 가중들에 기반하여, 트레이드-오프들이 결정될 수 있다.
- [0045] [0049] 프로세스(200)는, 논리 어드레싱에 따라, 가상 머신에 의한 데이터에 대한 판독 및 기록 액세스를 인에 이블링한다(206). 예컨대, 어떤 데이터 서브세트들도 이송되고 있지 않을 때, 가상 머신(120)은 게스트 데이터(142)의 모든 데이터 서브세트들로의 액세스를 갖는다.
- [0046] [0050] 프로세스(200)는, 제1 물리적 머신 상에 저장된 제1 데이터 서브세트가 제1 물리적 머신 및 제2 물리적 머신과 별개인 제3 물리적 머신에 이송되어야 한다고 결정한다(208). 예컨대, 스토리지 머신(140) 상에 저장된 데이터의 일부 또는 전부가 이송될 필요가 있을 수 있다. 다시, 다양한 이벤트들이 데이터의 일부 또는 데이터의 전부의 이송을 요구할 수 있다. 이 예에서, 도 1에서 팬텀으로 도시된 데이터 서브세트 인덱스(2)에 의해 표시되는 하나의 데이터 서브세트가 스토리지 머신(140)으로부터 스토리지 머신(150)으로 이송되어야 한다고 가정하라.
- [0047] [0051] 프로세스(200)는, 제1 데이터 서브세트에 대한 가상 머신에 의한 판독 및 기록 액세스를 차단하고, 다른 데이터 서브세트들에 대한 가상 머신에 의한 판독 및 기록 액세스를 유지한다(210). 일부 구현들에서, 차단은 스토리지 머신(140)에 의해 수행될 수 있다. 스토리지 머신(140)은, 가상 머신(120)이 이송을 겪고 있는 데이터 서브세트에 저장된 데이터로의 액세스를 갖지 않도록 통지를 가상 머신(120)에 전송할 수 있으며, 그런 다음, 가상 머신(120)은, 화살표들(3A 및 3B)에 의해 각각 표시된 바와 같이, 스토리지 머신(140) 또는 스토리지 머신(150)으로부터 이송이 완료되어 있다는 통지를 자신이 수신할 때까지, 임의의 판독 또는 기록 동작들을 홀딩할 수 있다.
- [0048] [0052] 대안적으로, 스토리지 머신(140)은 이송에 대해 가상 머신(120)에 통지하지 않을 수 있으며, 이송이 완료될 때에만 가상 머신(120)은 통지받는다. 데이터 서브세트가 비행 중일 때 가상 머신(120)이 데이터 서브세트로부터 데이터를 요청하면, 이 가상 머신(120)은 이송에 대해 통지받고, 그리고/또는 제2 스토리지 머신(150)으로 재지향될 수 있다.
- [0049] [0053] 위의 예시적인 구현들에서, 데이터 서브세트는, 호스트 머신(110)을 통해서가 아니라, 스토리지 머신(140)으로부터 스토리지 머신(150)으로 바로 이송된다. 그러나, 다른 구현들에서, 데이터 서브세트는 호스트 머신(110)을 통해 전달될 수 있다. 예컨대, 가상 머신(120)이 데이터 서브세트들의 이송을 처리하는 구현들에서, 데이터 서브세트들은 가상 머신에 전달되고, 그런 다음, 가상 머신(120)으로부터 스토리지 머신(150)으로 전송된다. 이는 도 1에서 화살표들(4A 및 4B)에 의해 도시되며, 이 화살표들(4A 및 4B)은 화살표(2)에 의해 표

시된 이송의 실제 데이터 경로를 표시한다.

[0050] 가상 머신(120)(또는 호스트 머신(110))은 이용가능한 다수의 상이한 스토리지 머신들 중 하나로부터 제2 스토리지 머신(150)을 선택할 수 있다. 이러한 후자의 구현은, 가상 머신 또는 임의의 다른 스토리지 머신의 어드레스에 대한 참조 없이, 그리고 이송의 상태를 추적하고 그 자신을 "이송 상태"로 있는 것으로서 식별할 필요 없이, 데이터 서브세트들을 저장하는 "무상태" 스토리지 머신들을 용이하게 한다. 대신에, 데이터 스토리지의 관리가 가상 머신(120)에 의해 처리된다.

[0051] 프로세스(200)는, 데이터 서브세트를 제3 물리적 머신 상에 저장하기 위해, 제1 물리적 머신으로부터 제3 물리적 머신으로 제1 데이터 서브세트를 이송한다(212). 위에서 설명된 바와 같이, 데이터 서브세트는 스토리지 머신(140)으로부터 스토리지 머신(150)으로 바로 전송될 수 있거나, 또는 대안적으로, 가상 머신(120)에 의해, 제1 스토리지 머신(140)으로부터 폐지되고, 그런 다음, 제2 스토리지 머신(150)에 전송될 수 있다.

[0052] 프로세스(200)는, 제3 물리적 머신 상의 제1 데이터 서브세트에 대한 어드레스 범위를 가상 머신에 대한 개개의 논리 어드레스 범위로 논리적으로 매핑함으로써, 매핑을 업데이트한다(214). 매핑을 업데이트하기 위해 사용되는 데이터는 사용되는 구현에 따라 좌우된다. 예컨대, 가상 또는 호스트 머신이 이송을 제어하는 구현에서, 가상 또는 호스트 머신은, 가상 머신이 데이터 서브세트를 전송한 스토리지 머신의 어드레스에 기반하여 매핑 데이터를 업데이트할 수 있다. 스토리지 머신들이 데이터 서브세트의 이송을 제어하는 구현들에서, 가상 또는 호스트 머신은, 데이터 서브세트를 수신한 스토리지 머신의 어드레스를 표시하는, 가상 머신에 의해 수신된 통지에 기반하여 매핑 데이터를 업데이트할 수 있다.

#### [0053] 스토리지 머신 제어를 겪는 라이브 이송

[0054] 도 3은 제1 물리적 머신으로부터 제3 물리적 머신으로 바로 데이터 서브세트들이 이송되는, 데이터의 라이브 이송을 위한 예시적인 프로세스(300)의 흐름도이다. 프로세스(300)는 스토리지 머신들(140 및 150) 중 하나 또는 둘 모두에서 구현될 수 있다.

[0055] 프로세스(300)는, 제1 물리적 머신 상에 저장된 제1 데이터 서브세트가 제3 물리적 머신에 이송되어야 한다고 결정한다(302). 예컨대, 스토리지 머신(140)은, 스토리지 머신(140)이 유지보수를 위해 오프라인이 되어야 하고 스토리지 머신에 저장된 데이터 전부를 이송할 필요가 있다고 결정하거나, 또는 스토리지 머신(140)이 스토리지 용량 한계치에 도달했고 스토리지 머신에 저장된 데이터의 일부분을 이송할 필요가 있다고 결정할 수 있다.

[0056] 프로세스(300)는, 제1 데이터 서브세트에 대한 가상 머신에 의한 판독 및 기록 액세스를 차단하고, 다른 데이터 서브세트들에 대한 가상 머신에 의한 판독 및 기록 액세스를 유지한다(304). 예컨대, 스토리지 머신(140)은 가상 머신에 통지를 전송하며, 이 통지는, 이송되고 있는 데이터 서브세트를 식별하며, 그리고 가상 머신이 성공적인 이송에 대해 통지받을 때까지, 이 데이터 서브세트에 기록하거나 또는 이 데이터 서브세트를 판독하려고 시도하지 않도록 가상 머신에 명령한다. 이는 도 1의 화살표(3A)에 의해 표시된다.

[0057] 프로세스(300)는, 제1 물리적 머신으로부터 바로 제3 물리적 머신으로 제1 데이터 서브세트를 이송한다(306). 예컨대, 스토리지 머신(140)은, 중개자로서 호스트 머신(110)을 수반하지 않으면서, 데이터 서브세트를 제2 스토리지 머신(150)에 전송한다.

[0058] 프로세스(300)는, 이송이 완료되어 있다는 통지를 가상 머신에 제공하고, 제1 데이터 서브세트에 대한 판독 및 기록 액세스를 인애이블링한다(308). 예컨대, 제1 스토리지 머신(140)은 데이터 서브세트의 성공적인 수신에 대한 확인응답을 제2 스토리지 머신(150)으로부터 수신할 수 있으며, 차례로, 제2 스토리지 머신(150)의 어드레스 및 데이터 서브세트의 이송의 통지를 가상 머신(120)에 전송할 수 있다. 그런 다음, 가상 머신(120)은 자신의 매핑 데이터(128)를 업데이트하고, 스토리지 머신(150) 상의 새로운 위치에 있는 데이터 서브세트로의 액세스를 재개할 수 있다. 대안적으로, 제2 스토리지 머신(150)이 데이터 서브세트를 성공적으로 수신한 후에, 제2 스토리지 머신은 제2 스토리지 머신(150)의 어드레스 및 데이터 서브세트의 이송의 통지를 가상 머신(120)에 전송할 수 있다. 그런 다음, 가상 머신(120)은 자신의 매핑 데이터(128)를 업데이트하고, 스토리지 머신(150) 상의 새로운 위치에 있는 데이터 서브세트로의 액세스를 재개할 수 있다.

#### [0059] 가상 머신 또는 호스트 머신 제어를 겪는 라이브 이송

[0060] 도 4는 제1 물리적 머신으로부터 제3 물리적 머신으로 그리고 가상 머신이 구현되어 있는 물리적 머신을 통해 데이터 서브세트들이 이송되는, 데이터의 라이브 이송을 위한 예시적인 프로세스(400)의 흐름도이다.

프로세스는 가상 머신(120)(또는 호스트 머신(110))에서 구현될 수 있다.

[0061] 프로세스(400)는, 제1 물리적 머신 상에 저장된 제1 데이터 서브세트가 제3 물리적 머신에 이송되어야 한다고 결정한다(402). 예컨대, 가상 머신(120)은, 스토리지 머신(140)이 높은 레이턴시를 갖는다고 결정할 수 있거나; 또는 부하 밸런싱 동작이 필요하다고 결정할 수 있거나; 또는 심지어, 제1 스토리지 머신(140)이 유지 보수를 위해 오프라인이 되고 있으며 가상 머신(120)에 대해 저장된 데이터를 이송할 필요가 있다는 통지를 제1 스토리지 머신(140)으로부터 수신할 수 있다.

[0062] 프로세스(400)는, 제1 데이터 서브세트를 제3 물리적 머신에 이송하도록 제1 물리적 머신에 명령한다(404). 예컨대, 가상 머신(120)은, 데이터 서브세트를 스토리지 머신(150)에 이송하도록 스토리지 머신에 명령한다. 가상 머신(120)은 또한, 이송이 완료될 때까지, 데이터 서브세트에 액세스하지 않을 것이다.

[0063] [0067] 일부 구현들에서, 가상 머신(120)은, 화살표들(3A 및 3B)에 의해 표시된 바와 같이, 제1 스토리지 머신(140)으로부터 데이터 서브세트를 수신하고, 이 데이터 서브세트를 스토리지 머신(150)에 전송할 수 있다. 다른 구현들에서, 가상 머신(120)은, 데이터 서브세트를 바로 스토리지 머신(150)에 전송하도록 스토리지 머신(140)에 명령할 수 있다.

[0064] [0068] 프로세스(400)는, 제3 물리적 머신 상의 제1 데이터 서브세트에 대한 어드레스 범위를 가상 머신에 대한 개개의 논리 어드레스 범위로 논리적으로 매핑함으로써, 매핑을 업데이트한다(406). 예컨대, 스토리지 머신(150)으로부터의 확인응답 메시지로부터 예컨대 성공적인 이송에 대한 통지를 수신하자마자, 가상 머신(120)은 매핑 데이터(128)를 업데이트하고, 데이터 서브세트로의 액세스를 복원시킨다.

#### [0069] 추가적인 구현 세부사항들

[0070] [0066] 추가적인 특징들을 실현하기 위해, 위에서 설명된 예시적인 시스템 및 프로세스들에 대한 변형들이 구현될 수 있다. 예컨대, 성공적인 이송에 대한 통지에 대해 대기하는 대신에, 가상 머신은 이송의 예상 완료 시간을 계산할 수 있다. 예상 시간이 경과한 후에, 가상 머신은 데이터 서브세트에 액세스하려고 시도할 수 있다. 성공적이지 않으면, 가상 머신은 다른 시간 기간 동안 대기할 수 있거나, 또는 대신에, 메모리 오류 이벤트를 호출할 수 있다.

[0071] [0067] 다른 구현들에서, 데이터 서브세트들은 블록 또는 페이지 레벨로 실현될 수 있으며, 그리고 X 아래의 모든 어드레스들이 제1 물리적 머신 상에 있고, X에 있거나 또는 그 위의 모든 어드레스들이 제3 물리적 머신 상에 있도록, "워터마크"가 사용된다(또는 데이터가 3 이상의 물리적 머신들에 저장될 때, 각각의 물리적 머신에 대해, 인접한 어드레스 범위들이 사용될 수 있다). X의 값은, 데이터가 이송될 때 데이터 프로세싱 프로세스에서 업데이트된다. 이는 데이터 서브세트로의 논리 어드레스의 데이터 프로세싱 프로세스 매핑을 제거할 수 있으며, 대신에, X의 워터마크 값에 따라, 데이터를 저장하는 물리적 머신들의 파티션들을 매핑한다. 그에 따라서, 매핑 데이터(128)를 실현하기 위한 메타데이터 요건들이 감소된다.

[0072] [0068] 다른 구현들에서, 사전-복사 및 사후-복사 이송 기법들은 데이터 서브세트 단위로 사용될 수 있다. 이 구현은, 시스템 복잡성을 초래하면서, 이송 동안 가상 머신의 대기 기간을 감소시키거나 또는 제거한다.

[0073] [0069] 위의 예들은 클라우드-기반 시스템의 맥락에서 또는 데이터 센터들에서 설명된다. 그러나, 본원에서 설명된 시스템들 및 방법들은, 데이터에 액세스하는 애플리케이션 또는 가상 머신이 실행되고 있는 컴퓨터로부터 원격으로 저장 데이터를 관리하는 임의의 시스템에서 활용될 수 있다.

[0074] [0070] 본 명세서에서 설명된 발명의 요지 및 동작들의 실시예들은, 본 명세서에서 개시된 구조들 및 그들의 구조적 등가물들을 포함하여 컴퓨터 소프트웨어, 펌웨어, 또는 하드웨어에서, 또는 디지털 전자 회로소자에서 또는 그들 중 하나 이상의 결합들로 구현될 수 있다. 본 명세서에서 설명된 발명의 요지의 실시예들은, 데이터 프로세싱 장치에 의한 실행을 위해 또는 데이터 프로세싱 장치의 동작을 제어하기 위해 컴퓨터 스토리지 매체 상에 인코딩된 하나 이상의 컴퓨터 프로그램들, 즉, 컴퓨터 프로그램 명령들의 하나 이상의 모듈들로서 구현될 수 있다.

[0075] [0071] 컴퓨터 스토리지 매체는, 컴퓨터-관독가능 스토리지 디바이스, 컴퓨터-관독가능 스토리지 기판, 랜덤 또는 시리얼 액세스 메모리 어레이 또는 디바이스, 또는 그들 중 하나 이상의 결합일 수 있거나 또는 이들에 포함될 수 있다. 게다가, 컴퓨터 스토리지 매체가 전파 신호는 아니지만, 컴퓨터 스토리지 매체는 인공적으로-생성된 전파 신호에 인코딩된 컴퓨터 프로그램 명령들의 소스 또는 목적지일 수 있다. 컴퓨터 스토리지 매체는 또한, 하나 이상의 별개의 물리적 컴포넌트들 또는 매체(예컨대, 다수의 CD들, 디스크들, 또는 다른 스토리지

디바이스들)일 수 있거나 또는 이들에 포함될 수 있다.

[0072] [0076] 본 명세서에서 설명된 동작들은, 하나 이상의 컴퓨터-판독가능 스토리지 디바이스들 상에 저장되거나 또는 다른 소스들로부터 수신된 데이터에 대해 데이터 프로세싱 장치에 의해 수행되는 동작들로서 구현될 수 있다.

[0073] [0077] "데이터 프로세싱 장치"란 용어는, 예로서 프로그램가능 프로세서, 컴퓨터, 단일 칩 시스템(system on a chip), 또는 이들 중 다수의 것들 또는 결합들을 포함하여, 데이터를 프로세싱하기 위한 모든 종류들의 장치, 디바이스들, 및 머신들을 포괄한다. 장치는 특수 목적 논리 회로소자, 예컨대 FPGA(field programmable gate array) 또는 ASIC(application specific integrated circuit)를 포함할 수 있다. 장치는 또한, 하드웨어 외에도, 문제의 컴퓨터 프로그램을 위한 실행 환경을 생성하는 코드, 예컨대 프로세서 펌웨어, 프로토콜 스택, 데이터베이스 관리 시스템, 운영체제, 크로스-플랫폼 런타임 환경, 가상 머신, 또는 그들 중 하나 이상의 결합을 구성하는 코드를 포함할 수 있다. 장치 및 실행 환경은 다양한 상이한 컴퓨팅 모델 인프라구조들, 이를테면 웹 서비스들, 분산 컴퓨팅 및 그리드 컴퓨팅 인프라구조들을 실현할 수 있다.

[0074] [0078] 컴퓨터 프로그램(프로그램, 소프트웨어, 소프트웨어 애플리케이션, 스크립트, 또는 코드로서 또한 알려짐)은, 컴파일링되거나 또는 인터프리팅되는 언어들, 선언적 또는 절차적 언어들을 포함하여 임의의 형태의 프로그래밍 언어로 작성될 수 있으며, 이러한 컴퓨터 프로그램은, 독립형 프로그램으로서 또는 컴퓨팅 환경에서 사용하기에 적절한 모듈, 컴포넌트, 서브루틴, 오브젝트, 또는 다른 유닛을 포함하는 임의의 형태로 배치될 수 있다. 컴퓨터 프로그램이 파일 시스템 내의 파일에 대응할 수는 있지만, 그럴 필요는 없다. 프로그램은 다른 프로그램들 또는 데이터(예컨대, 마크업 언어 문서에 저장된 하나 이상의 스크립트들)를 홀딩하는 파일의 일부분에, 문제의 프로그램에 전용되는 단일 파일에, 또는 다수의 조정된 파일들(예컨대, 하나 이상의 모듈들, 서브프로그램들, 또는 코드의 일부분들을 저장하는 파일들)에 저장될 수 있다. 컴퓨터 프로그램은 하나의 컴퓨터 상에서, 또는 하나의 장소에 위치되거나 또는 다수의 장소들에 걸쳐 분산되고 통신 네트워크에 의해 상호연결되는 다수의 컴퓨터들 상에서 실행되도록 배치될 수 있다.

[0075] [0079] 본 명세서에서 설명된 프로세스들 및 논리 흐름들은, 입력 데이터에 대해 동작하고 출력을 생성함으로써 동작들을 수행하기 위해 하나 이상의 컴퓨터 프로그램들을 실행하는 하나 이상의 프로그램가능 프로세서들에 의해 수행될 수 있다. 프로세스들 및 논리 흐름들은 또한 특수 목적 논리 회로소자, 예컨대 FPGA(field programmable gate array) 또는 ASIC(application specific integrated circuit)에 의해 수행될 수 있으며, 장치가 또한 이들로서 구현될 수 있다.

[0076] [0080] 컴퓨터 프로그램의 실행에 적절한 프로세서들은, 예로서 범용 마이크로프로세서 및 특수 목적 마이크로프로세서 둘 모두, 그리고 임의의 종류의 디지털 컴퓨터의 임의의 하나 이상의 프로세서들을 포함한다. 일반적으로, 프로세서는 판독 전용 메모리 또는 랜덤 액세스 메모리 또는 둘 모두로부터 명령들 및 데이터를 수신할 것이다. 컴퓨터의 필수적인 엘리먼트들은 명령들에 따라 동작들을 수행하기 위한 프로세서, 그리고 명령들 및 데이터를 저장하기 위한 하나 이상의 메모리 디바이스들이다. 일반적으로, 컴퓨터는 또한, 데이터를 저장하기 위한 하나 이상의 대용량 스토리지 디바이스들, 예컨대 자기, 마그네토 광학 디스크들 또는 광학 디스크들을 포함하거나, 또는 이들로부터 데이터를 수신하거나 또는 이들에 데이터를 전달하거나 또는 둘 모두를 위해 동작 가능하게 커플링될 것이다. 그러나, 컴퓨터가 그러한 디바이스들을 가질 필요는 없다. 게다가, 컴퓨터는 다른 디바이스, 예컨대, 단지 몇몇을 말하자면, 모바일 전화, PDA(personal digital assistant), 모바일 오디오 또는 비디오 플레이어, 게임 콘솔, GPS(Global Positioning System) 수신기, 또는 휴대용 스토리지 디바이스(예컨대, USB(universal serial bus) 플래시 드라이브)에 내장될 수 있다. 컴퓨터 프로그램 명령들 및 데이터를 저장하기에 적절한 디바이스들은, 예로서 반도체 메모리 디바이스들, 예컨대 EEPROM, EEPROM, 및 플래시 메모리 디바이스들; 자기 디스크들, 예컨대 내부 하드 디스크들 또는 착탈형 디스크들; 마그네토 광학 디스크들; 그리고 CD-ROM 및 DVD-ROM 디스크들을 포함하는 모든 형태들의 비-휘발성 메모리, 미디어 및 메모리 디바이스들을 포함한다. 프로세서 및 메모리는 특수 목적 논리 회로소자에 의해 보충되거나 또는 특수 목적 논리 회로소자에 통합될 수 있다.

[0077] [0081] 사용자와의 상호작용을 제공하기 위해, 본 명세서에서 설명된 발명의 요지의 실시예들은, 사용자에게 정보를 디스플레이하기 위한 디스플레이 디바이스, 예컨대 CRT(cathode ray tube) 또는 LCD(liquid crystal display) 모니터, 및 사용자가 입력을 컴퓨터에 제공할 수 있게 하는 키보드 및 포인팅 디바이스, 예컨대 마우스 또는 트랙볼을 갖는 컴퓨터 상에서 구현될 수 있다. 다른 종류들의 디바이스들도 또한, 사용자와의 상호작용을 제공하기 위해 사용될 수 있는데; 예컨대, 사용자에게 제공되는 피드백은 임의의 형태의 감지 피드백, 예

컨대, 시각 피드백, 가정 피드백, 또는 촉각 피드백일 수 있고; 사용자로부터의 입력은 음향, 스피치, 또는 촉각 입력을 포함하는 임의의 형태로 수신될 수 있다. 그 외에도, 컴퓨터는, 사용자에 의해 사용되는 디바이스에 문서들을 전송하고 이 디바이스로부터 문서들을 수신함으로써; 예컨대, 사용자의 사용자 디바이스 상의 웹 브라우저로부터 수신된 요청들에 대한 응답으로, 이 웹 브라우저에 웹 페이지들을 전송함으로써, 사용자와 상호작용할 수 있다.

[0078] 본 명세서에서 설명된 발명의 요지의 실시예들은, 예컨대 데이터 서버로서 후단 컴퓨포넌트를 포함하거나, 또는 미들웨어 컴퓨포넌트, 예컨대 애플리케이션 서버를 포함하거나, 또는 전단 컴퓨포넌트, 예컨대 사용자가 본 명세서에서 설명된 발명의 요지의 구현과 상호작용할 수 있게 하는 그래픽 사용자 인터페이스 또는 웹 브라우저를 갖는 사용자 컴퓨터를 포함하거나, 또는 하나 이상의 그러한 후단, 미들웨어, 또는 전단 컴퓨포넌트들의 임의의 결합을 포함하는 컴퓨팅 시스템에서 구현될 수 있다. 시스템의 컴퓨포넌트들은 임의의 형태 또는 매체의 디지털 데이터 통신, 예컨대 통신 네트워크에 의해 상호연결될 수 있다. 통신 네트워크들의 예들은 로컬 영역 네트워크("LAN(local area network)") 및 광역 네트워크("WAN(wide area network)"), 인터-네트워크(예컨대, 인터넷), 및 피어-투-피어 네트워크들(예컨대, 애드 혹 피어-투-피어 네트워크들)을 포함한다.

[0079] 컴퓨팅 시스템은 사용자들 및 서버들을 포함할 수 있다. 사용자 및 서버는 일반적으로 서로 원격이며, 통상적으로 통신 네트워크를 통해 상호작용한다. 사용자와 서버의 관계는, 개개의 컴퓨터들 상에서 실행되고 서로에 대해 사용자-서버 관계를 갖는 컴퓨터 프로그램들에 의해 생긴다. 일부 실시예들에서, 서버는 (예컨대, 사용자 디바이스와 상호작용하여, 사용자에게 데이터를 디스플레이하고 이 사용자로부터 사용자 입력을 수신하는 목적들을 위해) 사용자 디바이스에 데이터(예컨대, HTML 페이지)를 송신한다. 사용자 디바이스에서 생성된 데이터(예컨대, 사용자 상호작용의 결과)는 사용자 디바이스로부터 서버에서 수신될 수 있다.

[0080] 본 명세서가 많은 특정 구현 세부사항들을 포함하지만, 이들은 청구될 수 있는 것의 범위 또는 임의의 특징들의 범위에 대한 제한들로서 이해되는 것이 아니라, 특정 실시예들에 특정한 특징들의 설명들로서 이해되어야 한다. 별개의 실시예들의 맥락에서 본 명세서에서 설명되는 소정의 특징들은 또한, 단일 실시예로의 결합으로 구현될 수 있다. 반대로, 단일 실시예의 맥락에서 설명되는 다양한 특징들은 또한, 별개로 다수의 실시예들에서 또는 임의의 적절한 서브결합으로 구현될 수 있다. 게다가, 특징들이 소정의 결합들에서 동작하는 것으로서 위에서 설명되고 심지어 초기에는 그와 같이 청구될 수 있지만, 청구되는 결합으로부터의 하나 이상의 특징들은, 일부 경우들에서, 이 결합으로부터 삭제될 수 있으며, 청구되는 결합은 서브결합 또는 서브결합의 변형으로 지향될 수 있다.

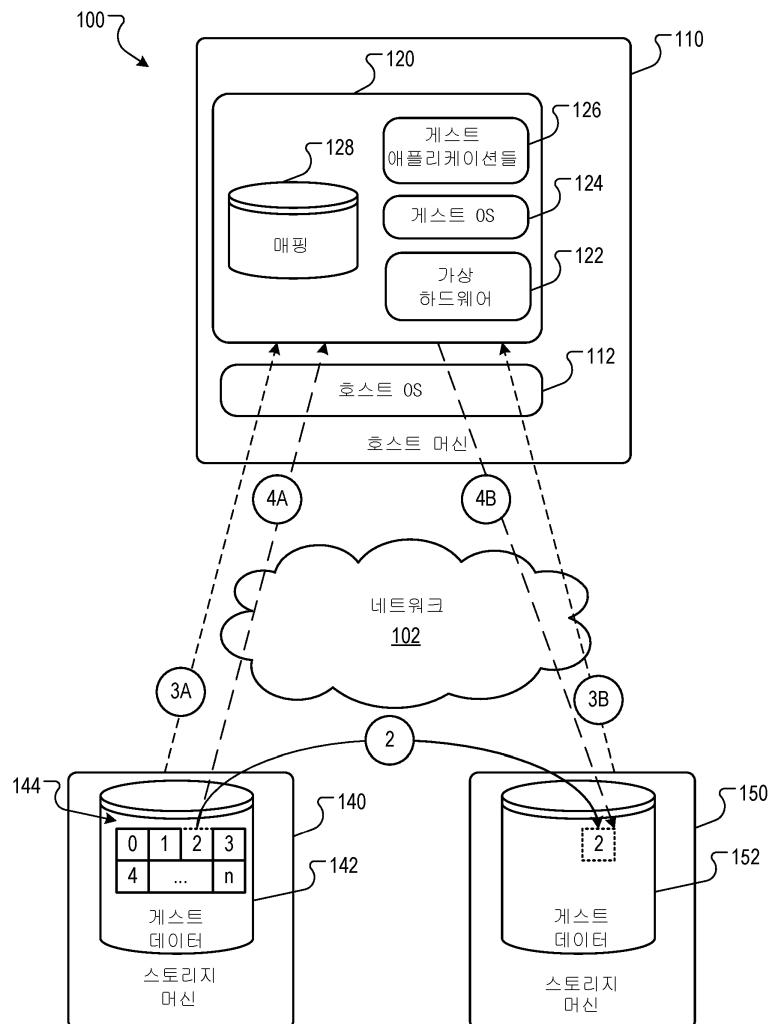
[0081] 유사하게, 동작들이 특정 순서로 도면들에서 도시되지만, 이것은, 바람직한 결과들을 달성하기 위해, 그러한 동작들이 도시된 특정 순서로 또는 순차적인 순서로 수행되거나 또는 모든 예시된 동작들이 수행될 것을 요구하는 것으로서 이해되지 않아야 한다. 소정의 상황들에서, 멀티태스킹 및 병렬 프로세싱이 유리할 수 있다. 게다가, 위에서 설명된 실시예들에서의 다양한 시스템 컴퓨포넌트들의 분리는 모든 실시예들에서 그러한 분리를 요구하는 것으로서 이해되지 않아야 하며, 설명된 프로그램 컴퓨포넌트들 및 시스템들이 일반적으로, 단일 소프트웨어 제품에 함께 통합되거나 또는 다수의 소프트웨어 제품들로 패키징될 수 있다는 것이 이해되어야 한다.

[0082] 예시적인 실시예들에 따라, 제1 스토리지 시스템으로부터 제2 스토리지 시스템으로 데이터를 이송하기 위한, 컴퓨터 스토리지 매체 상에 인코딩된 컴퓨터 프로그램들을 비롯한 방법들, 시스템들, 및 장치가 개시된다. 그 데이터는, 제1 스토리지 시스템 및 제2 스토리지 시스템과 별개인 컴퓨터 시스템 상에서 동작하는 프로세스(예컨대, 가상 머신, 애플리케이션, 또는 어떤 다른 프로세스)에 대한 것이다. 그 데이터는, 서로에 대해 각자 베타적인 데이터 서브셋들에 따라 저장된다. 각각의 데이터 세트가 이송될 때, 프로세스에 의한 데이터 세트로의 액세스는 방지되지만, 나머지 데이터 세트들에 대한 액세스는 영향받지 않는다. 일단 데이터 이송이 완료되면, 프로세스에 의한 데이터 세트로의 액세스는 복원된다.

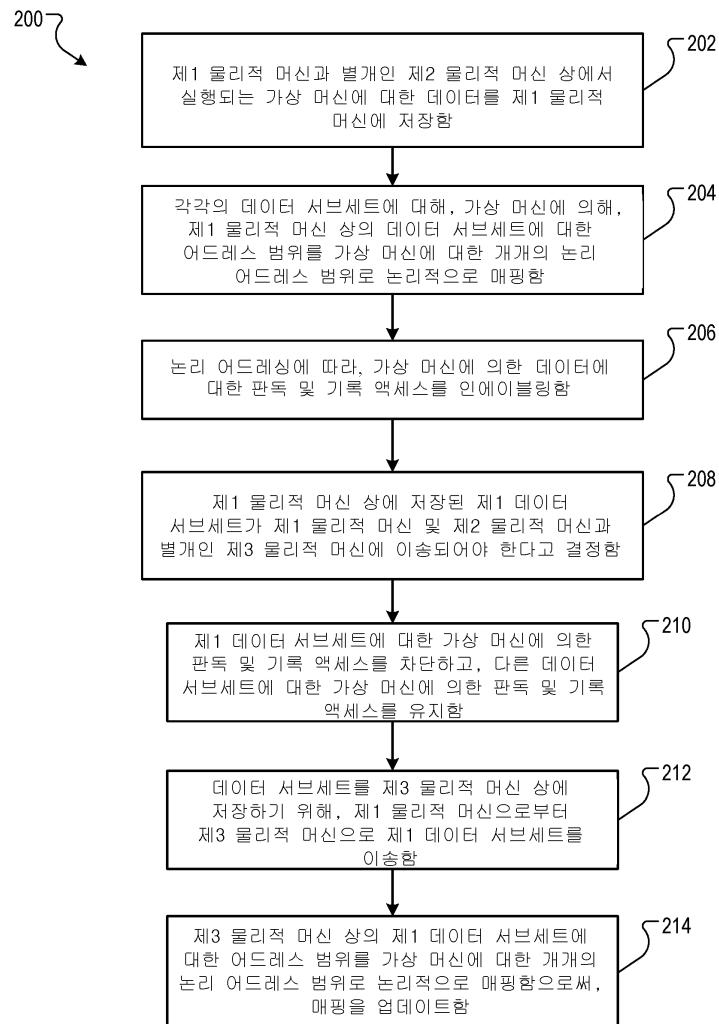
[0083] 따라서, 발명의 요지의 특정 실시예들이 설명되었다. 다른 실시예들은 다음의 청구항들의 범위 내에 있다. 일부 경우들에서, 청구항들에서 언급된 동작들은, 상이한 순서로 수행되며, 여전히 바람직한 결과들을 달성할 수 있다. 그 외에도, 첨부된 도면들에서 도시된 프로세스들이, 바람직한 결과들을 달성하기 위해, 도시된 특정 순서 또는 순차적인 순서를 반드시 요구하지는 않는다. 소정의 구현들에서, 멀티태스킹 및 병렬 프로세싱이 유리할 수 있다.

## 도면

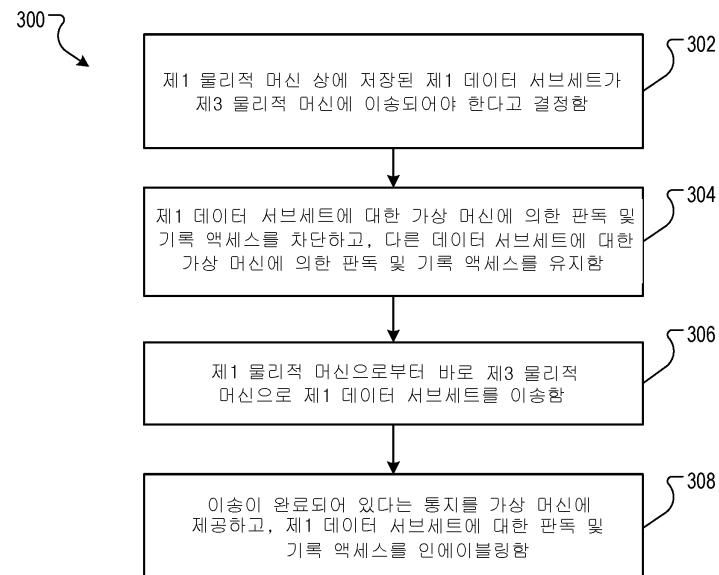
## 도면1



## 도면2



## 도면3



## 도면4

