



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2025년03월11일
(11) 등록번호 10-2777103
(24) 등록일자 2025년02월28일

(51) 국제특허분류(Int. Cl.)
G06F 3/06 (2006.01) G06F 16/17 (2019.01)
(52) CPC특허분류
G06F 3/0644 (2013.01)
G06F 16/1727 (2019.01)
(21) 출원번호 10-2023-7038560
(22) 출원일자(국제) 2021년07월21일
심사청구일자 2024년07월18일
(85) 번역문제출일자 2023년11월08일
(65) 공개번호 10-2023-0170024
(43) 공개일자 2023년12월18일
(86) 국제출원번호 PCT/US2021/042619
(87) 국제공개번호 WO 2022/220858
국제공개일자 2022년10월20일
(30) 우선권주장
17/229,709 2021년04월13일 미국(US)
(56) 선행기술조사문헌
US09418699 B1
KR1020200119542 A

(73) 특허권자
오라클 인터내셔널 코포레이션
미국, 캘리포니아 94065, 레드우드 쇼어스 엠에스 5오피7, 오라클 파크웨이 500
(72) 발명자
오토, 에리히
미국 94065 캘리포니아주 레드우드 쇼어스 엠/에스 5오피7 오라클 파크웨이 500 오라클 인터내셔널 코포레이션
커비, 마이클
미국 94065 캘리포니아주 레드우드 쇼어스 엠/에스 5오피7 오라클 파크웨이 500 오라클 인터내셔널 코포레이션
쿨카르니, 아진키아
미국 94065 캘리포니아주 레드우드 쇼어스 엠/에스 5오피7 오라클 파크웨이 500 오라클 인터내셔널 코포레이션
(74) 대리인
양영준, 김연송, 백만기

전체 청구항 수 : 총 20 항

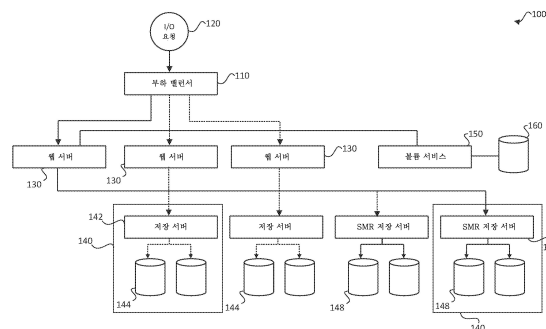
심사관 : 신현상

(54) 발명의 명칭 구역 세그먼트 드라이브 관리

(57) 요약

분산형 저장 시스템들에서 데이터 저장의 관리를 위한 기술들이 제공된다. 방법은, 컴퓨터 시스템에 의해, 불륨에 데이터를 기입하라는 요청을 수신하는 단계를 포함할 수 있다. 이 방법은, 컴퓨터 시스템에 의해, 불륨에 맵핑된 구역 세그먼트를 식별하는 단계를 포함할 수 있다. 구역 세그먼트는 복수의 구역을 포함할 수 있다. 이 방법은, 컴퓨터 시스템에 의해, 구역 세그먼트의 구역 내의 기입 위치를 나타내는 세그먼트 포인터를 식별하는 단계를 포함할 수 있다. 이 방법은, 컴퓨터 시스템에 의해, 그 기입 위치에서 시작하여 구역 세그먼트의 복수의 구역 중 하나 이상의 구역에 데이터를 기입하는 단계를 포함할 수 있다. 이 방법은 또한, 컴퓨터 시스템에 의해, 구역 세그먼트 내의 데이터의 종료점에 따라 세그먼트 포인터를 업데이트하는 단계를 포함할 수 있다.

대표도



(52) CPC특허분류

G06F 3/0604 (2013.01)
G06F 3/0608 (2013.01)
G06F 3/061 (2013.01)
G06F 3/0635 (2013.01)
G06F 3/064 (2013.01)
G06F 3/0653 (2013.01)
G06F 3/0665 (2013.01)
G06F 3/067 (2013.01)
G06F 3/0676 (2013.01)

명세서

청구범위

청구항 1

방법으로서,

컴퓨터 시스템에 의해, 볼륨에 데이터를 기입하라는 요청을 수신하는 단계;

상기 컴퓨터 시스템에 의해, 상기 볼륨에 맵핑된 구역 세그먼트를 식별하는 단계 - 상기 구역 세그먼트는 복수의 구역을 포함함 -;

상기 컴퓨터 시스템에 의해, 상기 구역 세그먼트 내의 기입 위치를 나타내는 세그먼트 포인터를 식별하는 단계;

상기 컴퓨터 시스템에 의해, 상기 기입 위치에서 시작하여, 상기 구역 세그먼트의 복수의 구역 중 하나 이상의 구역에 상기 데이터를 기입하는 단계 - 상기 하나 이상의 구역은 적어도 상기 구역 세그먼트 내의 구역 및 후속 구역을 포함함 -; 및

상기 컴퓨터 시스템에 의해, 상기 구역 세그먼트 내의 상기 데이터의 데이터 종료점에 따라 상기 세그먼트 포인터를 업데이트하는 단계

를 포함하는, 방법.

청구항 2

제1항에 있어서,

상기 요청에서 식별된 데이터의 크기를 확인하는 단계;

상기 구역의 용량을 확인하는 단계 - 상기 용량은 상기 기입 위치와 상기 구역의 구역 종료점 사이의 기입 경로에 대응함 -;

상기 구역의 용량과 상기 데이터의 크기를 비교하는 단계;

상기 구역의 용량을 초과하는 데이터의 크기에 따라, 상기 데이터를 제1 블록과 제2 블록으로 분할하는 (breaking) 단계 - 상기 제1 블록의 크기는 상기 구역의 용량에 대응함 -;

상기 기입 위치에서 시작하여 상기 제1 블록을 기입하는 단계; 및

상기 제2 블록을 상기 구역 세그먼트의 상기 후속 구역에 기입하는 단계

를 더 포함하는, 방법.

청구항 3

제1항에 있어서, 상기 세그먼트 포인터를 업데이트하는 단계는,

상기 구역 세그먼트의 용량과 일치하는 상기 구역 세그먼트 내의 상기 데이터의 상기 데이터 종료점에 따라, 상기 구역 세그먼트를 전체 구역 세그먼트로서 마킹하는 단계;

상기 볼륨에 맵핑된 새로운 구역 세그먼트를 식별하는 단계; 및

상기 세그먼트 포인터를 상기 새로운 구역 세그먼트의 시작점으로 업데이트하는 단계

를 포함하는, 방법.

청구항 4

제1항에 있어서, 상기 세그먼트 포인터를 식별하는 단계는 비어 있거나 개방된 구역들에 대해 상기 볼륨에 맵핑된 구역 세그먼트를 스캔하는 단계를 포함하는, 방법.

청구항 5

제4항에 있어서, 상기 세그먼트 포인터를 식별하는 단계는,

비어 있거나 개방된 구역을 포함하지 않는 볼륨에 맵핑된 구역 세그먼트에 따라, 새로운 구역 세그먼트를 상기 볼륨에 할당하는 단계

를 더 포함하고, 상기 세그먼트 포인터는 상기 새로운 구역 세그먼트의 제1 구역의 시작 위치인, 방법.

청구항 6

제1항에 있어서,

상기 구역 세그먼트의 이용 비율(usage fraction)을 확인하는 단계;

상기 이용 비율을 이용 임계값과 비교하는 단계;

상기 이용 비율이 상기 이용 임계값을 충족하지 않는 것에 따라:

상기 구역 세그먼트의 복수의 구역에서 저장된 데이터를 식별하는 단계;

상기 하나 이상의 소스 구역으로부터의 상기 저장된 데이터를 상기 복수의 구역 중 하나 이상의 비어 있거나 개방된 구역에 맵핑하는 단계;

상기 저장된 데이터를 상기 비어 있거나 개방된 구역에 기입하는 단계; 및

상기 소스 구역들을 개방 상태로 리셋하는 단계

를 더 포함하는, 방법.

청구항 7

제6항에 있어서, 상기 복수의 구역에서 저장된 데이터를 식별하는 단계는,

상기 볼륨에 대한 상기 구역 세그먼트의 맵핑을 기술하는 맵핑 메타데이터를 수신하는 단계; 및

상기 맵핑 메타데이터에 따라 상기 복수의 구역에서 상기 저장된 데이터를 위치파악하는(locating) 단계

를 포함하는, 방법.

청구항 8

제1항에 있어서, 상기 복수의 구역은 쉐들링된 자기 기록 저장 시스템(shingled magnetic recording storage system)에서 구현되는, 방법.

청구항 9

시스템으로서,

컴퓨터 실행가능한 명령어들을 저장하도록 구성된 메모리; 및

상기 메모리에 액세스하고 상기 컴퓨터 실행가능한 명령어들을 실행하여, 적어도:

볼륨에 데이터를 기입하라는 요청을 수신하고;

상기 볼륨에 맵핑된 구역 세그먼트를 식별하고 - 상기 구역 세그먼트는 복수의 구역을 포함함 -;

상기 구역 세그먼트 내의 기입 위치를 나타내는 세그먼트 포인터를 식별하고;

상기 기입 위치에서 시작하여, 상기 구역 세그먼트의 복수의 구역 중 하나 이상의 구역에 상기 데이터를 기입하고 - 상기 하나 이상의 구역은 적어도 상기 구역 세그먼트 내의 구역 및 후속 구역을 포함함 -;

상기 구역 세그먼트 내의 상기 데이터의 데이터 종료점에 따라 상기 세그먼트 포인터를 업데이트하도록

구성된 하나 이상의 프로세서

를 포함하는, 시스템.

청구항 10

제9항에 있어서, 상기 하나 이상의 프로세서는 상기 컴퓨터 실행가능한 명령어들을 실행하여, 추가로 적어도:

상기 요청에서 식별된 데이터의 크기를 확인하고;

상기 구역의 용량을 확인하고 — 상기 용량은 상기 기입 위치와 상기 구역의 구역 종료점 사이의 기입 경로에 대응함 —;

상기 구역의 용량과 상기 데이터의 크기를 비교하고;

상기 구역의 용량을 초과하는 데이터의 크기에 따라, 상기 데이터를 제1 블록과 제2 블록으로 분할하고 — 상기 제1 블록의 크기는 상기 구역의 용량에 대응함 —;

상기 기입 위치에서 시작하여 상기 제1 블록을 기입하고;

상기 제2 블록을 상기 구역 세그먼트의 상기 후속 구역에 기입하도록

추가로 구성되는, 시스템.

청구항 11

제9항에 있어서, 상기 세그먼트 포인터를 업데이트하는 것은,

상기 구역 세그먼트의 용량과 일치하는 상기 구역 세그먼트 내의 상기 데이터의 상기 데이터 종료점에 따라, 상기 구역 세그먼트를 전체 구역 세그먼트로서 마킹하는 것;

상기 볼륨에 맵핑된 새로운 구역 세그먼트를 식별하는 것; 및

상기 세그먼트 포인터를 상기 새로운 구역 세그먼트의 시작점으로 업데이트하는 것을 포함하는, 시스템.

청구항 12

제9항에 있어서, 상기 세그먼트 포인터를 식별하는 것은 비어 있거나 개방된 구역들에 대해 상기 볼륨에 맵핑된 구역 세그먼트를 스캔하는 것을 포함하는, 시스템.

청구항 13

제12항에 있어서, 상기 세그먼트 포인터를 식별하는 것은,

비어 있거나 개방된 구역을 포함하지 않는 볼륨에 맵핑된 구역 세그먼트에 따라, 새로운 구역 세그먼트를 상기 볼륨에 할당하는 것을 더 포함하고, 상기 세그먼트 포인터는 상기 새로운 구역 세그먼트의 제1 구역의 시작 위치인, 시스템.

청구항 14

제9항에 있어서, 상기 하나 이상의 프로세서는 상기 컴퓨터 실행가능한 명령어들을 실행하여, 추가로 적어도:

상기 구역 세그먼트의 이용 비율을 확인하고;

상기 이용 비율을 이용 임계값과 비교하고;

상기 이용 비율이 상기 이용 임계값을 충족하지 않는 것에 따라:

상기 구역 세그먼트의 복수의 구역에서 저장된 데이터를 식별하고;

상기 하나 이상의 소스 구역으로부터의 상기 저장된 데이터를 상기 복수의 구역 중 하나 이상의 비어 있거나 개방된 구역에 맵핑하고;

상기 저장된 데이터를 상기 비어 있거나 개방된 구역에 기입하고;

상기 소스 구역들을 개방 상태로 리셋하도록

추가로 구성되는, 시스템.

청구항 15

제14항에 있어서, 상기 복수의 구역에서 저장된 데이터를 식별하는 것은,
상기 볼륨에 대한 상기 구역 세그먼트의 맵핑을 기술하는 맵핑 메타데이터를 수신하는 것; 및
상기 맵핑 메타데이터에 따라 상기 복수의 구역에서 상기 저장된 데이터를 위치 파악하는 것
을 포함하는, 시스템.

청구항 16

컴퓨터 시스템의 하나 이상의 프로세서에 의해 실행될 때, 상기 하나 이상의 프로세서로 하여금 동작들을 수행하게 하는 상기 컴퓨터 시스템에 의해 실행가능한 명령어들을 저장한 비밀시적 컴퓨터 판독가능한 매체로서, 상기 동작들은,

컴퓨터 시스템에 의해, 볼륨에 데이터를 기입하라는 요청을 수신하는 동작;

상기 컴퓨터 시스템에 의해, 상기 볼륨에 맵핑된 구역 세그먼트를 식별하는 동작 - 상기 구역 세그먼트는 복수의 구역을 포함함 -;

상기 컴퓨터 시스템에 의해, 상기 구역 세그먼트 내의 기입 위치를 나타내는 세그먼트 포인터를 식별하는 동작;

상기 컴퓨터 시스템에 의해, 상기 기입 위치에서 시작하여, 상기 구역 세그먼트의 복수의 구역 중 하나 이상의 구역에 상기 데이터를 기입하는 동작 - 상기 하나 이상의 구역은 적어도 상기 구역 세그먼트 내의 구역 및 후속 구역을 포함함 -; 및

상기 컴퓨터 시스템에 의해, 상기 구역 세그먼트 내의 상기 데이터의 데이터 종료점에 따라 상기 세그먼트 포인터를 업데이트하는 동작

을 포함하는, 비밀시적 컴퓨터 판독가능한 매체.

청구항 17

제16항에 있어서, 상기 동작들은:

상기 요청에서 식별된 데이터의 크기를 확인하는 동작;

상기 구역의 용량을 확인하는 동작 - 상기 용량은 상기 기입 위치와 상기 구역의 구역 종료점 사이의 기입 경로에 대응함 -;

상기 구역의 용량과 상기 데이터의 크기를 비교하는 동작;

상기 구역의 용량을 초과하는 데이터의 크기에 따라, 상기 데이터를 제1 블록과 제2 블록으로 분할하는 동작 - 상기 제1 블록의 크기는 상기 구역의 용량에 대응함 -;

상기 기입 위치에서 시작하여 상기 제1 블록을 기입하는 동작; 및

상기 제2 블록을 상기 구역 세그먼트의 상기 후속 구역에 기입하는 동작

을 더 포함하는, 비밀시적 컴퓨터 판독가능한 매체.

청구항 18

제16항에 있어서, 상기 세그먼트 포인터를 업데이트하는 동작은:

상기 구역 세그먼트의 용량과 일치하는 상기 구역 세그먼트 내의 상기 데이터의 상기 데이터 종료점에 따라, 상기 구역 세그먼트를 전체 구역 세그먼트로서 마킹하는 동작;

상기 볼륨에 맵핑된 새로운 구역 세그먼트를 식별하는 동작; 및

상기 세그먼트 포인터를 상기 새로운 구역 세그먼트의 시작점으로 업데이트하는 동작

을 포함하는, 비밀시적 컴퓨터 판독가능한 매체.

청구항 19

제16항에 있어서, 상기 세그먼트 포인터를 식별하는 동작은 비어 있거나 개방된 구역들에 대해 상기 볼륨에 맵

핑된 구역 세그먼트를 스캔하는 동작을 포함하는, 비밀시적 컴퓨터 판독가능한 매체.

청구항 20

제19항에 있어서, 상기 세그먼트 포인터를 식별하는 동작은,

비어 있거나 개방된 구역을 포함하지 않는 볼륨에 맵핑된 구역 세그먼트에 따라, 새로운 구역 세그먼트를 상기 볼륨에 할당하는 동작을 더 포함하고, 상기 세그먼트 포인터는 상기 새로운 구역 세그먼트의 제1 구역의 시작 위치인, 비밀시적 컴퓨터 판독가능한 매체.

발명의 설명

기술 분야

[0001] 관련 출원의 상호참조

[0002] 본 출원은 2021년 4월 13일 출원된 발명의 명칭이 "ZONE SEGMENT DRIVE MANAGEMENT"인 미국 비-가출원 제 17/229,709호에 대한 우선권 및 이득을 주장한다. 미국 비-가출원 제17/229,709호의 내용들은 모든 목적을 위해 그 전문이 참조를 위해 본 명세서에 포함된다.

배경 기술

[0003] 클라우드 기반의 플랫폼은 사용자 데이터에 대한 스케일링가능하고 유연한 컴퓨팅 자원을 제공한다. IaaS(infrastructure as a service)라고도 하는 이러한 클라우드 기반의 플랫폼들은, 고객의 데이터 주위의 클라우드 솔루션들, 예를 들어, 변환들을 저작하고, 데이터를 로딩하고, 데이터를 제시하기 위한 솔루션들의 전체 스위트들(suites)을 제공할 수 있다.

[0004] 싱글링된 자기 기록(Shingled Magnetic Recording)(SMR)은 기록 트랙들을 부분적으로 중첩시키거나 싱글링함으로써 하드 디스크 드라이브들의 저장 용량을 증가시킨다. 중첩된 트랙들은 구역(zone)들로 그룹화될 수 있다. 판독 경로가 보존되는 방식으로 트랙들을 싱글링함으로써, 트랙들은 부분적으로 오버라이트되었더라도 여전히 판독될 수 있다. 대조적으로, 싱글링의 결과로서, 판독 동작들은 구역들 내에서 여전히 무작위로 수행될 수 있지만, 기입 동작들은, 구역 내의 범위가 아니라, 전체 구역에 순차적으로 적용될 필요가 있을 것이다.

발명의 내용

[0005] 싱글링된 자기 기록 드라이브들 상의 클라우드 자원 데이터의 객체 스토리지를 관리하기 위한 기술들(예컨대, 방법, 시스템, 하나 이상의 프로세서에 의해 실행가능한 코드 또는 명령어들을 저장하는 비밀시적인 컴퓨터 판독가능한 매체)이 제공된다.

[0006] 소정의 실시예들에서, 방법은, 컴퓨터 시스템에 의해, 볼륨(volume)에 데이터를 기입하라는 요청을 수신하는 단계를 포함할 수 있다. 이 방법은, 컴퓨터 시스템에 의해, 볼륨에 맵핑된 구역 세그먼트를 식별하는 단계를 포함할 수 있다. 구역 세그먼트는 복수의 구역을 포함할 수 있다. 이 방법은, 컴퓨터 시스템에 의해, 구역 세그먼트의 구역 내의 기입 위치를 나타내는 세그먼트 포인터를 식별하는 단계를 포함할 수 있다. 이 방법은, 컴퓨터 시스템에 의해, 그 기입 위치에서 시작하여 구역 세그먼트의 복수의 구역 중 하나 이상의 구역에 데이터를 기입하는 단계를 포함할 수 있다. 이 방법은 또한, 컴퓨터 시스템에 의해, 구역 세그먼트 내의 데이터의 종료점에 따라 세그먼트 포인터를 업데이트하는 단계를 포함할 수 있다.

[0007] 일부 실시예에서, 이 방법은 요청에서 식별된 데이터의 크기를 확인하는 단계를 더 포함할 수 있다. 이 방법은 구역의 용량을 확인하는 단계를 포함할 수 있으며, 여기서, 용량은 기입 위치와 구역의 종료점 사이의 기입 경로에 대응한다. 이 방법은 구역의 용량을 데이터의 크기와 비교하는 단계를 포함할 수 있다. 이 방법은, 구역의 용량을 초과하는 데이터의 크기에 따라, 데이터를 제1 블록과 제2 블록으로 분할하는 단계를 포함할 수 있다. 제1 블록의 크기는 구역의 용량에 대응할 수 있다. 이 방법은 기입 위치에서 시작하여 제1 블록을 기입하는 단계를 포함할 수 있다. 이 방법은 또한, 구역 세그먼트의 후속 구역에 제2 블록을 기입하는 단계를 포함할 수 있다.

[0008] 일부 실시예에서, 세그먼트 포인터를 업데이트하는 단계는, 구역 세그먼트의 용량과 일치하는 구역 세그먼트 내 데이터의 종료점에 따라, 구역 세그먼트를 전체 구역 세그먼트로서 마킹하는 단계를 포함할 수 있다. 세그먼트 포인터를 업데이트하는 단계는 또한, 볼륨에 맵핑된 새로운 구역 세그먼트를 식별하는 단계, 및 세그먼트 포인

터를 새로운 구역 세그먼트의 시작점으로 업데이트하는 단계를 포함할 수 있다. 세그먼트 포인터를 식별하는 단계는, 비어 있거나 개방된 구역들에 대해 볼륨에 맵핑된 구역 세그먼트를 스캔하는 단계를 포함할 수 있다.

[0009] 일부 실시예에서, 컴퓨터 시스템은, 하나 이상의 프로세서, 및 하나 이상의 프로세서와 통신하는 메모리를 포함하고, 메모리는 컴퓨터 실행가능한 명령어들을 저장하도록 구성되며, 컴퓨터 실행가능한 명령어들을 실행하는 것은 하나 이상의 프로세서로 하여금 위에서 설명된 방법 또는 그 변형들의 단계들 중 하나 이상을 수행하게 한다. 세그먼트 포인터를 식별하는 단계는, 비어 있거나 개방된 구역을 포함하지 않는 볼륨에 맵핑된 구역 세그먼트에 따라, 새로운 구역 세그먼트를 그 볼륨에 할당하는 단계를 더 포함할 수 있다. 세그먼트 포인터는 새로운 구역 세그먼트의 제1 구역의 시작 위치일 수 있다.

[0010] 일부 실시예에서, 이 방법은 구역 세그먼트의 이용 비율(usage fraction)을 확인하는 단계를 포함할 수 있다. 이 방법은 이용 비율을 이용 임계값과 비교하는 단계를 포함할 수 있다. 이 방법은, 이용 비율이 이용 임계값을 충족하지 않는 것에 따라, 구역 세그먼트의 복수의 구역에 저장된 데이터를 식별하고, 하나 이상의 소스 구역으로부터의 저장된 데이터를 복수의 구역의 하나 이상의 비어 있거나 개방된 구역에 맵핑하는 단계, 저장된 데이터를 비어 있거나 개방된 구역에 기입하는 단계, 및 소스 구역들을 개방 상태로 리셋하는 단계를 포함할 수 있다. 복수의 구역에 저장된 데이터를 식별하는 단계는, 구역 세그먼트를 볼륨에 맵핑하는 것을 기술하는 맵핑 메타데이터를 수신하는 단계, 및 맵핑 메타데이터에 따라 복수의 구역에서 저장된 데이터를 위치파악하는 단계를 포함할 수 있다.

[0011] 일부 실시예에서, 복수의 구역은 싱글링된 자기 기록 저장 시스템에서 구현될 수 있다.

[0012] 소정의 실시예들에서, 컴퓨터 시스템은, 하나 이상의 프로세서, 및 하나 이상의 프로세서에 의해 실행될 때 하나 이상의 프로세서로 하여금 전송된 방법 또는 그 변형들의 하나 이상의 동작을 수행하게 하는 명령어들을 저장하는 비일시적인 컴퓨터 판독가능한 메모리를 포함한다.

[0013] 소정의 실시예들에서, 컴퓨터 판독가능한 저장 매체는, 실행될 때 컴퓨터 시스템의 하나 이상의 프로세서로 하여금 전송된 방법 또는 그 변형들의 하나 이상의 동작을 수행하게 하는 컴퓨터 실행가능한 명령어들을 저장한다.

도면의 간단한 설명

[0014] 도 1은, 하나 이상의 실시예에 따른 객체 저장 서버를 관리하기 위한 예시적인 시스템을 나타낸다.

도 2는, 하나 이상의 실시예에 따른 예시적인 싱글링된 자기 기록 드라이브를 나타낸다.

도 3은, 하나 이상의 실시예에 따른 구역 세그먼트에 객체 데이터를 기입하기 위한 예시적인 흐름을 나타낸다.

도 4는, 하나 이상의 실시예에 따른 구역 세그먼트로부터 데이터를 판독하기 위한 예시적인 흐름을 나타낸다.

도 5는 하나 이상의 실시예에 따른 구역 세그먼트로부터 데이터를 삭제하기 위한 예시적인 흐름을 나타낸다.

도 6은 하나 이상의 실시예에 따른 구역 세그먼트를 조각모음(defragment)하기 위한 예시적인 흐름을 나타낸다.

도 7은 적어도 하나의 실시예에 따른 서비스 시스템으로서의 클라우드 인프라(cloud infrastructure)를 구현하기 위한 하나의 패턴을 나타내는 블록도이다.

도 8은 적어도 하나의 실시예에 따른 서비스 시스템으로서의 클라우드 인프라를 구현하기 위한 또 다른 패턴을 나타내는 블록도이다.

도 9는 적어도 하나의 실시예에 따른 서비스 시스템으로서의 클라우드 인프라를 구현하기 위한 또 다른 패턴을 나타내는 블록도이다.

도 10은 적어도 하나의 실시예에 따른 서비스 시스템으로서의 클라우드 인프라를 구현하기 위한 또 다른 패턴을 나타내는 블록도이다.

도 11은 적어도 하나의 실시예에 따른 예시적인 컴퓨터 시스템을 나타내는 블록도이다.

발명을 실시하기 위한 구체적인 내용

[0015] 이하의 설명에서, 다양한 실시예가 설명될 것이다. 설명의 목적을 위해, 실시예의 철저한 이해를 제공하기 위하여 특정한 구성들 및 상세사항들이 개시된다. 그러나, 실시예들은 특정한 상세사항 없이도 실시될 수 있다는

것이 본 기술분야의 통상의 기술자에게 명백할 것이다. 또한, 설명되는 실시예를 모호하게 하지 않기 위해 잘 알려진 피쳐들은 생략되거나 간소화될 수 있다.

[0016] 클라우드 기반의 플랫폼은 사용자들에게 스케일링가능하고 유연한 컴퓨팅 자원들을 제공한다. IaaS(infrastructure as a service)라고도 하는 이러한 클라우드 기반의 플랫폼들은, 고객의 데이터 주위의 클라우드 솔루션들, 예를 들어, 변환들을 저작하고, 데이터를 로딩하고, 데이터를 제시하기 위한 솔루션들의 전체 스위트들을 제공할 수 있다. 백엔드 데이터 저장 시스템의 일부일 수 있는 데이터 복제 및 백업 시스템에서, 싱글링된 자기 기록(SMR) 드라이브는 데이터 저장 시스템의 저장 용량을 증가시킬 수 있다. SMR 드라이브는 기록 트랙을 부분적으로 중첩시키거나 싱글링함으로써, 주어진 크기의 하드 디스크에 더 많은 트랙이 기입되는 것을 허용할 수 있다. 중첩된 트랙들은 구역들로 그룹화될 수 있고, 구역들은 구역 세그먼트들로 조직화될 수 있다. 관독 경로가 보존되는 방식으로 트랙들을 싱글링함으로써, 트랙들은 부분적으로 오버라이트되었더라도 여전히 관독될 수 있다. 대조적으로, 싱글링의 결과로서, 부분적으로 오버라이트된 데이터는 관독될 수는 있지만 오버라이트될 수는 없는 부가-전용(append-only) 프로세스에서 기입 동작이 적용될 수 있다. 따라서, SMR 드라이브들에 대한 기입에 관여된 추가적인 동작들로 인해, 아래에서 설명되는 바와 같이, 파일 시스템이 아니라, 블록 인터페이스에 의해 SMR 드라이브들에 대한 액세스가 제공될 수 있다.

[0017] 분산형 저장 시스템들에서, 사용자 데이터는 볼륨 스토리지(예컨대, 블록 볼륨 스토리지) 및 객체 스토리지에 저장될 수 있다. 사용자들은 볼륨 저장 시스템들과 직접 상호작용할 수 있는 반면, 데이터베이스 시스템의 백엔드 저장 서브시스템은, 예를 들어, 데이터 복제, 백업 또는 (예컨대, 데이터베이스 시스템 용량을 보충하기 위한) 기타의 데이터 저장을 위해 객체 스토리지를 이용할 수 있다. 데이터 복제 또는 백업 시스템들에서 SMR 드라이브들의 구현은, SMR 드라이브들에 데이터를 기입하고 재기입하는 상대적인 복잡성에 의해 제한된다. 도 2를 참조하여 더 상세히 설명되는 바와 같이, 구역들에서 부분적으로 오버라이트된 트랙들을 조직화하는 것은, 시스템이 구역 내의 가용 용량을 초과하는 데이터를 구역에 기입하려고 시도할 때 비효율성을 야기할 수 있다. 예를 들어, 기존 SMR 드라이브들에서, 시스템은 구역의 용량을 초과하는 데이터를 구역에 기입하라는 요청을 거부할 수 있다. 이러한 제한을 해결하기 위해, 구역들은 구역 세그먼트들로 조직화될 수 있고, 데이터 저장을 관리하는 시스템은 기입 요청들을 개개의 구역이 아니라 구역 세그먼트에 어드레싱할 수 있다. 이러한 방식으로, 구역 세그먼트 내의 기입 위치가 식별될 수 있고, 기입 요청에 의해 참조되는 데이터는 구역 세그먼트 내의 여러 구역에 걸쳐 기입될 수 있다.

[0018] 일부 예에서, 빠르고 효율적이며 신뢰성 있는 데이터 전송이 필요한 경우 블록 스토리지가 종종 이용된다. 블록 스토리지는 데이터를 블록들로 나누고 이들 블록들을 별개의 조각들(pieces)로서 저장할 수 있다(예컨대, 각각의 블록은 그 자신의 식별자를 가짐). 그 다음, 블록들은 상이한 시스템들에 걸쳐 저장될 수 있으며 각각의 블록은 상이한 시스템들에서 작동하도록 구성(예컨대, 파티셔닝)될 수 있다. 대안으로서, 객체 스토리지는 데이터 파일을 여러 조각(예컨대, 객체들)으로 나눈 다음, 이들을 여러 네트워킹된 시스템들에 걸쳐 분산될 수 있는 단일 저장소에 저장한다. 블록 스토리지의 정황에서, 볼륨(예컨대, 블록 볼륨)은 개발자가 인스턴스의 저장 용량을 동적으로 확장시키는 것을 허용하는 분리가 가능한(예컨대, 물리적으로 또는 가상으로) 블록 저장 디바이스일 수 있다. 블록 볼륨 스토리지의 크기를 증가 또는 감소시키기 위해, 블록 볼륨들은 큰 노력 없이 각각 증가되거나 감소될 수 있다. 데이터베이스 시스템이 볼륨 스토리지와 객체 스토리지 양쪽 모두를 구현하는 경우, I/O 요청들은 볼륨 식별자(예컨대, 블록 볼륨 식별자)를 통해 데이터를 참조할 수 있다. 볼륨 식별자는, 데이터베이스 시스템의 일부로서 포함된 메타데이터 데이터베이스에 저장된 볼륨 메타데이터를 통해, 객체 저장 시스템들 내의 저장 위치들에 맵핑될 수 있다. 기입 요청의 경우, 데이터베이스 시스템은 기입 요청에서 참조되는 볼륨에 맵핑된 데이터를 저장하는 SMR 시스템에서 구역 세그먼트를 식별할 수 있다. 일단 식별되고 나면, 데이터베이스 시스템은 새로운 데이터가 기입될 수 있는 구역 세그먼트에 포함된 한 구역 내의 기입 위치를 기술하는 구역 세그먼트 포인터를 식별할 수 있다. 구역을 오버라이트하는 것이 아니라, SMR 시스템은 구역에 기입할 수 있고, 데이터를 여러 구역에 걸쳐 기입될 블록들로 나눌 수 있으며, 데이터를 저장할 용량을 갖는 구역이 이용가능하지 않다면 요청을 거부하거나, 오버런 데이터(overrun data)를 저장할 새로운 구역 세그먼트(들)를 할당할 수 있다. 이러한 방식으로, SMR 시스템은, SMR 스토리지를 이용하여 객체 저장 시스템의 유효 용량이 증가되는 동시에 구역 용량을 초과하는 데이터 크기에 의해 야기되는 구역 기입 실패들의 영향을 제한하는 방식으로 하드 디스크 드라이브들에 데이터를 기입할 수 있다. 유사한 방식으로, 구역 세그먼트들에 저장된 데이터에 맵핑된 볼륨 메타데이터를 참조하여, SMR 시스템을 이용해, 관독, 삭제 및 조각모음 기술들이 역시 구현될 수 있다.

[0019] 한 예시적인 예에서, 데이터베이스 시스템은 클라우드 자원 데이터를 분산형 저장 시스템의 일부로서 저장하도

록 구성될 수 있다. 데이터베이스 시스템은 SMR 시스템을 포함할 수 있다. SMR 시스템은, 구역 세그먼트들로 조직화된 구역들을 이용하여 SMR 저장을 위해 구성된 하드 디스크 드라이브들이거나 이들을 포함할 수 있는 하나 이상의 SMR 데이터베이스 시스템과 통신하는 SMR 저장 서버를 포함할 수 있다. 이 예에서, 시스템은, 기입 요청들("put" 요청들이라고도 함), 판독 요청들("get" 요청들이라고도 함) 등의, 입력/출력(I/O) 요청들뿐만 아니라, 삭제 요청들 또는 조각모음 동작들을 포함한 그러나 이것으로 제한되지 않는 다른 I/O 프로세스들을 수신할 수 있다. SMR 구성의 일부로서, I/O 요청들은, 데이터베이스 시스템으로 하여금 SMR 시스템에서 구역 세그먼트들에 저장된 데이터를 식별하고 관리하는 개선된 프로세스들을 구현하게 할 수 있다.

[0020] 백엔드 저장 서브시스템으로 I/O 프로세스를 구현하는 프로세스는, 적어도 SMR 구성된 저장 시스템으로부터의 효율적인 I/O를 허용하기 때문에, 데이터 복제, 저장 및 백업 시스템들의 전반적인 성능을 개선한다. 예를 들어, SMR 시스템들은 이전에 기입된 자기 트랙들을 적어도 부분적으로 싱글링함으로써, 시스템에 하드 디스크 드라이브들을 추가하지 않고도 최대 25% 이상의 증가된 저장 용량을 제공할 수 있다. SMR 구역들을 구역 세그먼트들로 조직화하는 것은, SMR 드라이브들에 관해 이루어진 I/O 요청들의 서비스하는 속도, 효율성 및 성공률의 개선을 허용함으로써 I/O 프로세스들을 개선할 수 있다. 또한, 뒤따르는 도면들을 참조하여 설명되는 바와 같이 구역 드라이브들을 관리함으로써, 데이터베이스 시스템은, 객체 데이터를 볼륨 데이터에 맵핑하기 위해 볼륨 메타데이터를 참조하는 통합된 블록 인터페이스에 의해 제어되는 호스트 관리 드라이브들을 데이터베이스 시스템이 이용하도록 허용함으로써 시스템 복잡성을 감소시킬 수 있다.

[0021] 도 1은, 하나 이상의 실시예에 따른 객체 저장 서버를 관리하기 위한 예시적인 시스템(100)을 나타낸다. 데이터베이스 시스템(100)은 백엔드 데이터 저장 서브시스템의 일부로서 다양한 자기 기록 기술(예컨대, 수직 자기 기록(PMR), SMR 등)을 구현할 수 있는 여러 저장 시스템에 걸쳐 사용자 데이터를 분산시키고 저장하도록 구성될 수 있다. 이러한 방식으로, 데이터베이스 시스템(100)은 데이터 복제의 일부로서 객체 스토리지에 저장된 데이터에 대한 I/O 요청들을 서비스하거나, 볼륨 데이터를 블록 볼륨 시스템들에 복원하거나, 분산형 데이터 저장 시스템에서의 개선된 I/O 동작들을 용이화하는 한 접근법으로서 구역들을 구역 세그먼트들로 조직화하는 SMR 시스템들의 조각모음을 수행할 수 있다.

[0022] 일부 실시예에서, 시스템(100)은, I/O 요청(120)을 수신하고 I/O 요청(120)을 웹 서버(130)에 전송하도록 구성될 수 있는 부하 밸런서(110)를 포함할 수 있다. 부하 밸런서(110)는, 여러 상이한 물리적 위치들에 위치할 수 있고 그리고/또는 단일의 물리적 위치에 여러 웹 서버(130)를 포함할 수 있는, 분산형 저장 시스템의 일부로서 여러 웹 서버(130)에 걸쳐 I/O 요청들을 분산할 수 있다. 마찬가지로, 웹 서버(130)는 하나 이상의 저장 시스템(140)과 통신할 수 있다. 저장 시스템(140)은 PMR 또는 SMR 구성들로 데이터를 저장하도록 구성될 수 있다. 예를 들어, 저장 시스템(140)은 하나 이상의 PMR 드라이브(144)와 통신하는 저장 서버(142)를 포함할 수 있다. 추가적으로, 웹 서버(130)는, 도 2를 참조하여 더 상세히 설명되는 바와 같이, 하나 이상의 SMR 드라이브(148)와 통신하는 SMR 저장 서버(146)를 포함하는 저장 시스템과 통신할 수 있다. 시스템(100)은, 예를 들어, SMR 저장 서버(146)에 의한 소프트웨어 실행을 통해, SMR 드라이브들(148)에 저장된 데이터에 관한 I/O 요청(120)을 실행하도록 구성될 수 있다.

[0023] 시스템(100)은 볼륨 서비스(150)를 참조하여 저장 시스템(140)에 저장된 데이터를 볼륨 데이터에 맵핑할 수 있다. 이러한 방식으로, I/O 요청(120)은, 볼륨 메타데이터 데이터베이스(160)에서, 저장 시스템(140)의 스토리지 내의 위치에 맵핑될 수 있는 볼륨 식별자에 대한 참조를 포함할 수 있다. 한 예에서, 판독 요청은, 볼륨 메타데이터 데이터베이스(160)의 한 엔트리에서, 구역 세그먼트에 맵핑될 수 있는 볼륨 식별자, 세그먼트 포인터, 및 판독 크기를 참조할 수 있다.

[0024] 이러한 방식으로, 시스템(100)은 분산형 데이터 저장 시스템의 일부로서 SMR 드라이브들(148)을 관리할 수 있다. 유리하게도, 시스템(100)은, 백 엔드 데이터 저장 서브시스템이 프론트 엔드 블록 볼륨 시스템과의 통합을 유지하면서 SMR 드라이브들(148)의 증가된 저장 용량으로부터 혜택을 받는 것을 허용할 수 있다. 도 3 내지 도 6을 참조하여 더 상세히 설명되는 바와 같이, 데이터베이스 시스템(100)은, 예를 들어 데이터 복제 또는 백업의 일부로서 I/O 요청(120)을 수신할 수 있고, PMR 드라이브들(144) 및/또는 SMR 드라이브들(148)을 판독, 기입, 또는 조각모음을 수행함으로써 I/O 요청들(120)을 구현할 수 있다.

[0025] 도 2는 하나 이상의 실시예에 따른 예시적인 싱글링된 자기 기록 SMR 드라이브(200)를 나타낸다. SMR 드라이브(200)의 물리적 프로세스들 및 데이터 구조들은, SMR 드라이브(200)의 디스크들 상의 자기 트랙들(210)을 오버라이트하는 것에 기초한다. PMR 드라이브에서, 자기 트랙들은 오버라이트되지 않는다. 따라서, PMR 드라이브에 저장된 데이터는 무작위로 액세스될 수 있다(예컨대, PMR 드라이브는 랜덤 액세스 하드 드라이브 스토리지로

서 이용될 수 있다). 대조적으로, 랜덤 액세스가 아니라 데이터를 부가(appending)함으로써 기입할 수 있는 SMR 드라이브들은, 전형적으로, 수정될 가능성이 거의 없는 데이터를 효율적으로 판독하도록 구성된다.

[0026] SMR 드라이브(200) 내의 데이터는, 부가-전용 I/O 시스템의 일부로서, 파일 시스템이 아니라 블록 시스템을 이용하여 저장될 수 있다. 따라서, 데이터 블록은, 구역 세그먼트들(230)로 조직화될 수 있는, 고정된 크기(예를 들어, 256MB)의 구역들(220)에 기입될 수 있다. 물리적으로, 각각의 구역은, 데이터 트랙(250)을 제공하기 위해 부분적으로 오버라이트되는 고정된 기입 폭(240)을 갖는 다수의 자기 트랙(210)에 의해 기술될 수 있다. 데이터 트랙은, 예를 들어, SMR 드라이브(200)의 판독/기입 헤드(270)의 명세들에 의존하는 SMR 드라이브(200)의 파라미터일 수 있는 데이터 트랙 피치(260)에 의해 기술될 수 있다. 더 좁은 데이터 트랙 피치(260)를 이용하면, SMR 드라이브(200)의 전체 용량이 증가하며, 데이터 트랙(250)을 정확하게 판독하는 판독/기입 헤드(270)의 능력을 포함한 요인들에 의해 제한된다.

[0027] 구역들(220)과 구역 세그먼트들(230)의 계층적 조직화를 예증하기 위해 고정된 폭을 갖는 자기 트랙들(210)이 예시되어 있다. PMR 드라이브에서 이루어지는 것과 같이, 개별 트랙 내의 단일 위치에 데이터를 인덱싱하는 것 대신에, SMR 드라이브(200)는, 구역 세그먼트(230)의 시작점, 구역(220)의 시작점, 및/또는 데이터가 발견될 구역(220) 내의 위치를 나타내는 포인터들에 의해 인덱싱된 여러 자기 트랙(210)에 걸쳐 연장되는 구역들(220)에서 데이터를 조직화할 수 있다. 예를 들어, 포인터는 구역(220)에서 마지막으로 기입된 위치를 기술하는 기입 포인터를 포함할 수 있고, 그 곳에서 후속 기입 동작들이 새로운 데이터를 부가할 것이다.

[0028] 일부 경우에, 구역(220)은 데이터가 구역(220)에 기입될 수 있는지를 통제할 수 있는 하나 이상의 상태에 의해 기술될 수 있다. 예를 들어, 구역(220)의 상태들은, "비어 있음", "가득참", "개방됨", "폐쇄됨" 또는 "완료됨"을 포함할 수 있지만 이것으로 제한되는 것은 아니다. 비어 있음 상태는, 구역(220)이 구역(220)의 자기 트랙들(210)의 시작부로부터 기입될 수 있다는 것을 나타낼 수 있다(예를 들어, 구역(220)에 대한 기입 포인터는 구역(220)의 시작부에 있음). 마찬가지로, 가득참 상태는 구역(220)이 추가적인 기입 동작들에 이용가능하지 않음을 나타낼 수 있다(예컨대, 구역(220)에 대한 기입 포인터는 구역(220)의 끝 또는 끝 근처에 있음). 개방됨 상태는 구역(220)이 기입 동작들에 이용가능하다는 것을 나타낼 수 있다(예를 들어, 구역(220)은 구역(220)의 자기 트랙들(210)에 데이터를 기입하는데 이용가능한 용량을 갖고 SMR 드라이브(220)는 기입 동작들을 구현하기 위해 할당된 자원들을 갖는다). 대조적으로, 폐쇄됨 상태는 구역(220)이 기입 동작들에 이용가능하지 않지만, 개방될 수 있다는 것을 나타낼 수 있다. 완료됨 상태는, 데이터가 구역(220)에 기입되지 않을 것임을 나타낼 수 있다(예컨대, 구역(220)에 대한 기입 포인터는 추가적인 기입 동작들을 방지하기 위해 구역의 끝으로 이동될 수 있다). 완료된 구역은, 구역(220)을 빈 상태로 설정함으로써(예컨대, 기입 포인터를 구역(220)의 시작부로 이동시킴으로써) 개방될 수 있다.

[0029] 구역들(220)을 구역 세그먼트들(230)로 조직화하는 것은, 각각의 구역에 대한 기입 위치가 아니라, 구역 세그먼트(230)에 대한 기입 위치를 기술하는 세그먼트 포인터가 정의될 수 있는 방식으로 구역들(220)이 함께 관리되는 것을 허용할 수 있다. 구역 세그먼트(230)의 구역들(220) 내에서 구역-특정 기입 위치들은 전술된 상태들을 따를 수 있는 반면, 세그먼트 포인터는 구역 세그먼트(230)의 제1 이용가능한 개방된 구역의 기입 위치를 기술할 수 있다. 예를 들어, 구역들(220)은 구역 세그먼트(230)에서 순차적으로 조직화될 수 있다. 이러한 방식으로, 데이터는 구역 세그먼트(230) 내의 여러 구역(220)에 걸쳐 블록들에 기입될 수 있다. 순차적인 방식으로 구역들을 조직화하는 것은, SMR 드라이브(200)가 개방된 구역의 용량을 초과하는 길이의 데이터를 참조하는 기입 요청들을 더 양호하게 수용하는 것을 허용할 수 있다. SMR 드라이브(200)는 구역 세그먼트에서 여러 구역을 개방할 수 있고, 단일 기입 요청에 기초하여 여러 기입 동작을 정의할 수 있다. 아래의 도 3을 참조하여 더 상세히 설명되는 바와 같이, 기입 동작들은, 구역 세그먼트(230)의 구역(220)에서 데이터 기입을 시작할 위치를 나타내는 세그먼트 포인터를 식별함으로써 구현될 수 있다.

[0030] 도 3은, 하나 이상의 실시예에 따른 구역 세그먼트에 객체 데이터를 기입하기 위한 예시적인 흐름(300)을 나타낸다. 흐름의 동작들은 하드웨어 회로로 구현되고 그리고/또는 도 1의 시스템(100) 등의 컴퓨터 시스템의 비일시적인 컴퓨터 판독가능한 매체에 컴퓨터 판독가능한 명령어들로서 저장될 수 있다. 구현될 때, 명령어들은 컴퓨터 시스템의 프로세서(들)에 의해 실행가능한 회로 또는 코드를 포함하는 모듈들을 나타낸다. 이러한 명령어들의 실행은, 여기서 설명된 특정한 동작들을 수행하도록 컴퓨터 시스템을 구성한다. 프로세서와 조합된 각각의 회로 또는 코드는, 그 각자의 동작(들)을 수행한다. 동작들은 특정한 순서로 예시되어 있지만, 특정한 순서는 필요하지 않으며 하나 이상의 동작이 생략되거나, 스킵되고, 그리고/또는 재정렬될 수 있다는 점을 이해해야 한다.

- [0031] 예에서, 흐름(300)은, 컴퓨터 시스템이 기입 요청을 수신하는 동작(310)을 포함한다. 기입 요청(예컨대, 도 1의 I/O 요청(120))은, 예를 들어, 블록 볼륨 시스템이 아니라 객체 스토리지에 데이터를 저장하라는 사용자 요청에 기초하여, 데이터 복제, 아카이브, 또는 기타의 저장을 관리하는 분산형 데이터 저장 시스템으로부터 수신될 수 있다. 기입 요청 수신에 일부로서, 컴퓨터 시스템은, 도 1을 참조하여 더 상세히 설명된 바와 같이, 요청을 (예컨대, IaaS 인프라 시스템의 일부로서의) 다른 네트워킹된 시스템들에 전달할 수 있다. 예를 들어, 기입 요청은, 기입 요청을 처리할 저장 서버(예를 들어, 도 1의 SMR 저장 서버(146))를 선택할 수 있는 웹 서버(예를 들어, 도 1의 웹 서버(130))에 전달될 수 있다.
- [0032] 예에서, 흐름(300)은, 선택사항으로서, 컴퓨터 시스템이 비어 있거나 개방된 구역들을 찾기 위해 구역 세그먼트를 스캔하는 동작(320)을 포함한다. 기입 요청은 정의된 길이의 데이터 청크를 포함할 수 있고, 여기서 청크는 구역 세그먼트(예컨대, 도 2의 구역 세그먼트(230))에 기입될 데이터 블록이다. 도 2를 참조하여 더 상세히 설명되는 바와 같이, 구역 세그먼트는 구역들의 용량의 정수배에 상당하는 저장 용량을 제공하는 여러 구역(예를 들어, 도 2의 구역(220))을 포함할 수 있다. 구역 세그먼트 내의 하나 이상의 구역은 비어 있거나 개방되어 있을 수 있으므로 기입 동작들에 이용가능할 수 있다. 구역들은, 개방된 구역 다음에 빈 구역이 올 수 있도록 구역 세그먼트에서 순서대로 조직화될 수 있다. 빈 구역은 반드시 자기 트랙들이 비어 있는 구역을 나타내는 것이 아니라, 구역에 대한 기입 위치가 구역의 시작부에 있는 구역을 나타낼 수 있다는 것을 이해해야 한다.
- [0033] 예에서, 흐름(300)은, 선택사항으로서, 컴퓨터 시스템이 볼륨에 새로운 구역 세그먼트를 할당하는 동작(330)을 포함한다. 일부 경우에, 구역 세그먼트의 스캔은, 구역 세그먼트의 중단 구역 앞에 비어 있거나 개방된 구역을 식별하지 못할 수 있다. 일부 실시예에서, 시스템은 기입 요청에서 참조된 볼륨에(예를 들어, 볼륨 식별자에 의해) 새로운 구역 세그먼트를 할당하여 참조된 볼륨에 대한 데이터를 기입하는데 이용가능한 용량을 증가시킬 수 있다. 새로운 구역 세그먼트는 용량 부족으로 인해 거부되는 기입 요청들의 비율을 감소시키기 위한 목적으로 유지되는 자유 풀(free pool)의 구역 세그먼트들로부터 할당될 수 있다. 일부 실시예에서, 새로운 구역 세그먼트는 구역 세그먼트를 저장하는 동일한 SMR 드라이브로부터 할당될 수 있거나, SMR 저장 서버(예컨대, 도 1의 SMR 저장 서버(146))를 통해 조율되는 상이한 SMR 드라이브로부터 할당될 수 있다.
- [0034] 예에서, 흐름(300)은, 컴퓨터 시스템이 세그먼트 포인터를 식별하는 동작(340)을 포함한다. 구역 세그먼트(또는 새로운 구역 세그먼트) 내에서, 시스템은 개방된 또는 비어 있음 상태로 기술되는 구역 세그먼트의 첫 번째 구역을 위치파악함으로써 세그먼트 포인터를 식별할 수 있다. 예를 들어, 구역들이 구역 세그먼트 내에서 순차적으로 조직화되는 경우, 세그먼트 포인터는 첫 번째 개방된 구역의 기입 위치에 대응할 수 있다. 또 다른 예에서, 세그먼트 포인터는 첫 번째 빈 구역의 시작 위치에 대응할 수 있고, 여기서 구역 세그먼트는 개방된 구역을 포함하지 않는다.
- [0035] 예에서, 흐름(300)은, 컴퓨터 시스템이 세그먼트 포인터에 따라 구역 세그먼트들의 한 구역에 데이터를 기입하는 동작(350)을 포함한다. 세그먼트 포인터부터 시작하여, 시스템은 기입 요청에 의해 참조되는 데이터를 세그먼트 포인터에 대응하는 구역에 기입할 수 있다. 기입 동작은, 세그먼트 포인터에 대응하는 구역이 기입 요청에 의해 참조되는 전체 청크를 위한 용량을 갖고 있는지를 시스템에 의해 초기 체크하는 것을 포함할 수 있다. 구역이 불충분한 용량을 갖는 경우들에서, 시스템은 청크가 구역 세그먼트의 여러 구역에 걸쳐 분할될 수 있도록 여러 블록을 정의할 수 있다. 예시적인 예에서, 구역 세그먼트의 구역들은 순차적으로 조직화될 수 있고, 청크는 제1 블록과 제2 블록으로 분할하되, 제1 블록의 크기가 세그먼트 포인터에 대응하는 구역의 기입 용량에 대응하는 방식으로 분할할 수 있다. 제1 블록은 세그먼트 포인터에서 시작하여 기입되고 제2 블록은 다음 구역에 기입될 수 있다. 이 예는 구역들을 순서대로 기술하지만, 방법(300)은 다른 접근법들로 구현될 수 있다.
- [0036] 예에서, 흐름(300)은 컴퓨터 시스템이 세그먼트 포인터를 업데이트하는 동작(360)을 포함한다. 데이터가 여러 구역에 기입되는 다중 기입 동작들을 포함할 수 있는 기입 동작이 완료된 후, 시스템은 데이터의 끝 위치에 대한 세그먼트 포인터를 정의할 수 있다. 이러한 방식으로, 업데이트된 세그먼트 포인터는, 기입 요청의 수신 이전의 세그먼트 포인터에 대응하는 구역에 관해, 구역 세그먼트의 상이한 구역에서의 기입 위치에 대응할 수 있다.
- [0037] 예에서, 흐름(300)은 컴퓨터 시스템이 볼륨 메타데이터를 업데이트하는 동작(370)을 포함한다. 도 1을 참조하여 더 상세히 설명된 바와 같이, 구역 세그먼트에 기입된 데이터는 볼륨 메타데이터를 통해 블록 볼륨에 맵핑될 수 있다. 볼륨 메타데이터는 세그먼트 포인터를 포함할 수 있고, 그에 따라, 볼륨 메타데이터를 업데이트하는 것은, 볼륨 메타데이터 데이터베이스(예컨대, 도 1의 볼륨 메타데이터 데이터베이스(160))에서 세그먼트 포인터를 업데이트하는 것뿐만 아니라, 동작들 320 및 330의 일부로서, 새로운 구역 세그먼트들이 할당된 경우 볼륨-

구역 세그먼트 맵핑을 업데이트하는 것을 포함할 수 있다. 일부 예에서, 볼륨 메타데이터는, 동작 360에서 업데이트되기 전의 세그먼트 포인터 값에 대응하는 데이터가 기입된 시작 위치 포인터를 저장할 수 있다.

[0038] 예에서, 흐름(300)은 컴퓨터 시스템이 기입 요청의 상태를 출력하는 동작(380)을 포함한다. 일부 실시예에서, 시스템은 다양한 상이한 유형들의 상태 보고서들을 출력하도록 구성될 수 있다. 예를 들어, 이용률 및 용량 데이터 등의 자원 메타데이터가 IaaS 시스템 동작들에 대한 개선의 일부로서 수집 및 관리될 수 있다. 또 다른 예로서, 기입 요청의 성공 또는 실패가 시스템에 반환될 수 있어서, 시스템은 방법(300)의 동작들 중 일부 또는 전부를 반복할 수 있다. 또 다른 예로서, 시스템은, 기입 요청이 인간 사용자로부터 시작된 경우, IaaS 콘솔 인터페이스의 사용자에게 상태를 출력할 수 있다.

[0039] 구역 세그먼트 조직화는 SMR 드라이브들을 통합하는 데이터베이스 시스템의 동작에 대해 상당한 기술적 개선을 제공할 수 있다. 여러 구역을 포함하는 구역 세그먼트에 기입하는 기술들을 구현함으로써, 여러 구역에 걸쳐 데이터가 기입될 수 있고, 여러 구역의 용량이 공유될 수 있다. 이러한 방식으로, 구역 세그먼트 포인터들에 맵핑된 볼륨-메타데이터에 의해 용이화되는 데이터 복제, 저장, 및 백업 동작들은, SMR 기입 및 재/기입 동작들의 효율성을 개선할 수 있다.

[0040] 도 4는, 하나 이상의 실시예에 따른 구역 세그먼트로부터 데이터를 판독하기 위한 예시적인 흐름(400)을 나타낸다. 흐름의 동작들은 하드웨어 회로로 구현되고 그리고/또는 도 1의 시스템(100) 등의 컴퓨터 시스템의 비일시적인 컴퓨터 판독가능한 매체에 컴퓨터 판독가능한 명령어들로서 저장될 수 있다. 구현될 때, 명령어들은 컴퓨터 시스템의 프로세서(들)에 의해 실행가능한 회로 또는 코드를 포함하는 모듈들을 나타낸다. 이러한 명령어들의 실행은, 여기서 설명된 특정한 동작들을 수행하도록 컴퓨터 시스템을 구성한다. 프로세서와 조합된 각각의 회로 또는 코드는, 그 각자의 동작(들)을 수행한다. 동작들은 특정한 순서로 예시되어 있지만, 특정한 순서는 필요하지 않으며 하나 이상의 동작이 생략되거나, 스킵되고, 그리고/또는 재정렬될 수 있다는 점을 이해해야 한다.

[0041] 한 예에서, 흐름(400)은, 컴퓨터 시스템이 판독 요청을 수신하는 동작(410)을 포함한다. 도 1을 참조하여 더 상세히 설명된 바와 같이, 판독 요청은 판독될 볼륨과 데이터 블록을 참조할 수 있다. 볼륨은, 볼륨 메타데이터에서 SMR 드라이브에 맵핑될 수 있는, 볼륨 식별자 및 블록 식별자에 의해 참조될 수 있다. SMR 드라이브들은 구역들을 포함하는 구역 세그먼트들을 정의할 수 있고, 구역들은 구역 세그먼트들에서 순차적으로 조직화될 수 있다. 이러한 방식으로, 시스템은 데이터 액세스 동작의 일부로서, 예를 들어 데이터 복원 프로세스의 일부로서, 판독 요청을 수신할 수 있고, 데이터를 판독할 구역 세그먼트의 여러 구역에 액세스하도록 구성될 수 있다.

[0042] 예에서, 흐름(400)은, 컴퓨터 시스템이 (예컨대, 볼륨 메타데이터가 도 3의 370에서 업데이트될 때) 볼륨 메타데이터에 저장되어 있는 저장된 포인터 및 판독 길이를 식별하는 동작(420)을 포함한다. 저장된 포인터와 판독 길이는 판독 요청에서 식별된 볼륨에 대응할 수 있다. 볼륨 식별자는, 저장된 포인터에 의해, 구역 세그먼트 및 구역 세그먼트의 구역에 맵핑될 수 있다. 도 3을 참조하여 설명된 방법과는 대조적으로, 흐름(400)의 저장된 포인터는 기입 위치가 아니라 판독 위치를 참조할 수 있다. 그럼에도 불구하고, 이것은 SMR 드라이브가 데이터를 판독할 구역 세그먼트 내의 시작 위치를 참조하는 것으로 유사하게 이해될 수 있다.

[0043] 판독 요청은 또한, SMR 드라이브 상의 판독 길이에 맵핑될 수 있는, 판독될 데이터에 대한 참조를 포함할 수 있다. 판독 길이는, 판독 길이가 구역 세그먼트 내의 여러 구역과 중첩하는지를 시스템이 결정할 수 있도록, 판독될 트랙(예컨대, 도 2의 데이터 트랙(250))의 크기 또는 길이에 대응할 수 있다. 예를 들어, 구역들이 구역 세그먼트에서 순차적으로 조직화된 경우, 시스템은 세그먼트 포인터에서 첫 번째 판독 위치를 표시하고 구역 세그먼트 내의 다음 구역의 시작 위치에서 두 번째 판독 위치를 표시할 수 있다.

[0044] 예에서, 흐름(400)은, 선택사항으로서, 컴퓨터 시스템이 구역 세그먼트 내의 각각의 구역에 대한 단일 판독 커맨드들을 결정하는 동작(430)을 포함한다. 판독 길이가, 동작 420에서 식별된 세그먼트 포인터로부터 시작하여, 여러 구역에 걸쳐 확장되는 경우, 시스템은 구역 세그먼트의 상이한 구역들에 대한 여러 판독 커맨드를 결정할 수 있다. 여러 단일 판독 커맨드들은, 판독 커맨드에 의해 반환된 데이터가 동작 420에서 결정된 판독 길이에 따라 재구성될 수 있도록 상호참조될 수 있다. 이러한 방식으로, 여러 구역에 걸쳐 기입된 데이터는, 개개의 구역들에 어드레싱된 여러 판독 요청이 아니라, 구역 세그먼트를 참조하는 단일 판독 요청에 의해 판독될 수 있다.

[0045] 예에서, 흐름(400)은, 컴퓨터 시스템이 SMR 드라이브에게 판독 커맨드를 내리는 동작(440)을 포함한다. 도 1을

참조하여 더 상세히 설명된 바와 같이, 시스템은 저장 서버들(예컨대, 도 1의 저장 서버(142) 또는 SMR 서버(146))를 통해 데이터 저장을 관리할 수 있다. 판독 커맨드는 동작 420에서 결정된 세그먼트 포인터 및 판독 길이를 포함할 수 있고, 시스템은, 도 1을 참조하여 더 상세히 설명된 볼륨 메타데이터 등의, 맵핑 메타데이터에 의해 참조되는 SMR 저장 서버에게 판독 커맨드를 내릴 수 있다.

[0046] 예에서, 흐름(400)은, 컴퓨터 시스템이 판독 데이터 및 판독 요청의 상태를 출력하는 동작(450)을 포함한다. 데이터의 출력은, SMR 드라이브로부터의 데이터를 판독 요청에서 참조된 목적지에 전달하는 것을 포함할 수 있다. 예를 들어, 시스템은 데이터를 볼륨 저장 시스템에 출력하거나, 하나의 객체 스토리지 드라이브로부터 또 다른 객체 스토리지 드라이브로의 데이터 복제 동작들의 일부로서 출력할 수 있다. 흐름(300)과 유사하게, 상태는 요청에 대한 성공/실패 정보뿐만 아니라, 판독 커맨드에서 참조되는 구역들 및 데이터를 기술하는 메타데이터를 포함한 그러나 이것으로 제한되지 않는 기타의 정보를 포함할 수 있다.

[0047] 흐름(400)의 동작들을 이용하여 판독 커맨드들을 용이화하는 것은, SMR 드라이브들을 통합하는 데이터베이스 시스템의 동작에 대한 상당한 기술적 개선을 제공할 수 있다. 구역 세그먼트들로 조직화된 여러 구역에 어드레싱된 판독 요청들을 서비스하는 기술들을 구현함으로써, 구역 세그먼트 내의 각각의 구역에 대한 여러 판독 요청을 생성하지 않고도 SMR 드라이브들로부터 데이터를 더욱 효율적으로 판독할 수 있다.

[0048] 도 5는, 하나 이상의 실시예에 따른 구역 세그먼트로부터 데이터를 삭제하기 위한 예시적인 흐름(500)을 나타낸다. 흐름의 동작들은 하드웨어 회로로 구현되고 그리고/또는 도 1의 시스템(100) 등의 컴퓨터 시스템의 비일시적인 컴퓨터 판독가능한 매체에 컴퓨터 판독가능한 명령어들로서 저장될 수 있다. 구현될 때, 명령어들은 컴퓨터 시스템의 프로세서(들)에 의해 실행가능한 회로 또는 코드를 포함하는 모듈들을 나타낸다. 이러한 명령어들의 실행은, 여기서 설명된 특정한 동작들을 수행하도록 컴퓨터 시스템을 구성한다. 프로세서와 조합된 각각의 회로 또는 코드는, 그 각자의 동작(들)을 수행한다. 동작들은 특정한 순서로 예시되어 있지만, 특정한 순서는 필요하지 않으며 하나 이상의 동작이 생략되거나, 스킵되거나, 및/또는 재정렬될 수 있다는 점을 이해해야 한다.

[0049] 예에서, 흐름(500)은, 컴퓨터 시스템이 삭제 요청을 수신하는 동작(510)을 포함한다. 판독 또는 기입 동작들과는 대조적으로, 삭제 동작들은 시작 위치로서 세그먼트 포인터를 참조하지 않고 진행될 수 있다. 대신에, 삭제 요청은, 도 2를 참조하여 더 상세히 설명된 SMR 드라이브들(예컨대, 도 2의 SMR 드라이브(200)) 등의, 하나 이상의 드라이브의 객체 스토리지에 맵핑될 수 있는 볼륨 식별자 정보를 포함할 수 있다.

[0050] 예에서, 흐름(500)은, 컴퓨터 시스템이 볼륨 객체에 대한 맵핑 메타데이터를 판독하는 동작(520)을 포함한다. 도 1을 참조하여 더 상세히 설명된 바와 같이, 맵핑 메타데이터는 삭제 요청 대상인 블록 볼륨에 대한 데이터를 저장하는 구역 세그먼트를 참조할 수 있다. 이러한 방식으로, 시스템은 맵핑 메타데이터를 참조함으로써 삭제 요청에서 식별된 볼륨에 할당된 구역 세그먼트 또는 구역 세그먼트들을 식별할 수 있다.

[0051] 예에서, 흐름(500)은 컴퓨터 시스템이 볼륨 객체의 상태를 업데이트하는 동작(530)을 포함한다. 볼륨 객체의 상태를 업데이트하는 것은, 후속 기입 요청들에 대해 세그먼트 포인터가 구역 세그먼트 내의 제1 구역의 시작 위치에 대응하도록, 할당된 구역 세그먼트에 포함된 구역들의 상태를 "가득참" 또는 "개방됨"으로부터 "비어 있음"으로 수정하라는 명령어를 저장 서버(예컨대, 도 1의 SMR 저장 서버(146))에 전송하는 것을 포함할 수 있지만 이것으로 제한되지는 않는다. 추가적으로 또는 대안으로서, 시스템은, 후속 I/O 요청들(예컨대, 도 1의 I/O 요청(120))에 포함된 볼륨 식별자가 SMR 드라이브 내의 구역 세그먼트들과 연관되지 않도록, 맵핑 정보를 제거하기 위해 볼륨 메타데이터 데이터베이스(예컨대, 도 1의 볼륨 메타데이터 데이터베이스(160)) 내의 볼륨 메타데이터를 수정할 수 있다. 예를 들어, 도 3을 참조하여 설명된 흐름(300)의 동작들에서, 세그먼트 포인터를 식별하는 것은 참조된 볼륨 식별자에 어떠한 구역 세그먼트도 할당되지 않는다는 것을 반환할 수 있다. 이러한 방식으로, 삭제 요청들은 구역 세그먼트의 구역의 일부에 어드레싱될 수 있다. 전체 구역을 소거하는 것이 아니라, 기입 요청들이 부분적으로 오버라이트된 트랙들에 기입할 수 없다는 점을 제외하고, SMR 드라이브가 부가-전용 드라이브가 아니라 랜덤 액세스 드라이브와 더 유사한 방식으로 동작하도록 허용할 수 있다.

[0052] 예에서, 흐름(500)은 컴퓨터 시스템이 삭제 요청의 상태를 출력하는 동작(540)을 포함한다. 도 3 및 도 4의 이전 흐름들에서와 같이, 시스템은 삭제 요청에 의해 어드레싱된 자원들이 예를 들어 중앙 메타데이터 관리 시스템에 해제되었음을 나타내는 삭제 요청 상태를 반환할 수 있다. 추가적으로 또는 대안으로서, 시스템은 삭제 요청의 발신자에게 상태 성공/실패 메시지를 반환하여 요청이 재전송되어야 하는지 또는 재시도될 것인지를 나타낼 수 있다.

- [0053] 도 6은 하나 이상의 실시예에 따른 구역 세그먼트를 조각모음하기 위한 예시적인 흐름(600)을 나타낸다. 흐름의 동작들은 하드웨어 회로로 구현되고 그리고/또는 도 1의 시스템(100) 등의 컴퓨터 시스템의 비일시적인 컴퓨터 판독가능한 매체에 컴퓨터 판독가능한 명령어들로서 저장될 수 있다. 구현될 때, 명령어들은 컴퓨터 시스템의 프로세서(들)에 의해 실행가능한 회로 또는 코드를 포함하는 모듈들을 나타낸다. 이러한 명령어들의 실행은, 여기서 설명된 특정한 동작들을 수행하도록 컴퓨터 시스템을 구성한다. 프로세서와 조합된 각각의 회로 또는 코드는, 그 각자의 동작(들)을 수행한다. 동작들은 특정한 순서로 예시되어 있지만, 특정한 순서는 필요하지 않으며 하나 이상의 동작이 생략되거나, 스킵되고, 그리고/또는 재정렬될 수 있다는 점을 이해해야 한다.
- [0054] SMR 드라이브에 관한 판독/기입 동작들을 개선하는 것에 추가하여, 일부 실시예에서, SMR 드라이브의 구역들을 구역 세그먼트들로 조직화하고 구역들에 저장된 데이터를 리맵핑하여 스토리지의 비효율적인 이용을 감소시킴으로써 조각모음 동작들도 역시 개선될 수 있다. 유익하게도, 구역 세그먼트들을 조각모음하는 것은, SMR 드라이브들이 엄격하게 부가-전용 드라이브들이 아니라 랜덤 액세스 드라이브들처럼 기능하도록 허용할 수 있지만, 여전히 부가-전용 기입 동작들로 제한될 것이다. 예시적인 예에서, 도 5를 참조하여 더 상세히 설명되는 바와 같이, 삭제 요청들은, 드라이브들에 직접 기입된 데이터가 아니라 맵핑 메타데이터를 수정할 수 있다. 동작 과정에 걸쳐, SMR 드라이브(예컨대, 도 2의 SMR 드라이브(200))는, 구역 세그먼트의 하나의 구역에서 시작하여 후속 구역에서 끝나는 데이터가 삭제된 경우와 같이, 구역 세그먼트 용량의 일부를 이용할 수 있다. 부분적으로 중첩된 트랙들 내의 데이터는 SMR 드라이브들에 재기입될 수 없으므로, 이용되지 않고 남은 용량은 기입 동작들에 이용가능하지 않을 수 있어서, SMR 드라이브의 비효율성과 저장 용량의 저활용성을 도입할 수 있다. SMR 드라이브를 조각모음하는 것은, 데이터를 새로운 구역에 재기입하고 볼륨 메타데이터를 리맵핑함으로써 시스템이 이용되지 않은 저장 용량을 해제하는 것을 허용할 수 있다.
- [0055] 예에서, 흐름(600)은 컴퓨터 시스템이 구역 세그먼트의 이용 비율을 확인하는 동작(610)을 포함한다. 조각모음의 일부로서, 시스템은 임계값을 초과하는 이용 비율에 기초하여 자동(예컨대, 사람의 개입 없이) 프로세스를 구현하여 조각모음 동작들 트리거할 수 있다(예컨대, 이용 비율의 주기적 체크, 삭제 요청에 후속하는 이용 비율의 리프레시 등). 이 목적을 위해, 시스템은 볼륨 메타데이터에 액세스하여 이용 비율을 확인할 수 있다. 이용 비율은 볼륨 메타데이터에서 맵핑된 구역 세그먼트의 구역들 내의 저장 용량의 일부를 기술할 수 있다. 예를 들어, 볼륨 메타데이터는, 예를 들어 고정 용량을 갖는 구역 세그먼트에 기초하여, 구역 세그먼트의 총 용량의 일부에 해당하는, 구역 세그먼트에 저장된 데이터에 대한 시작점들, 종료점들, 체크 길이들 등을 참조할 수 있다.
- [0056] 예에서, 흐름(600)은 컴퓨터 시스템이 이용 비율을 이용 임계값과 비교하는 동작(620)을 포함한다. 이용 임계값은 이용 비율의 측면에서 표현될 수 있다. 예를 들어, 이용 비율이 50% 미만으로 떨어질 때 조각모음을 구현하도록 시스템이 구성된 경우, 이용 임계값은 0.5로서 표현될 수 있다.
- [0057] 예에서, 흐름(600)은 컴퓨터 시스템이 구역 세그먼트의 구역들에 저장된 데이터를 식별하는 동작(630)을 포함한다. 동작 620의 비교의 결과로서 조각모음 프로세스가 트리거되면, 시스템은, 볼륨 메타데이터를 이용하여 및/또는 구역 세그먼트를 스캔함으로써, 구역 세그먼트의 조각모음 동안에 이동될 데이터를 식별할 수 있다. 랜덤 액세스 드라이브의 조각모음이 일부 데이터를 이동시키면서 다른 데이터를 드라이브 상에 그대로 두는 것을 허용하지만, SMR 드라이브들의 부가-전용 기입은 데이터를 식별하여 새로운 구역에 리맵핑하고 데이터 복사가 이루어진 구역들을 소거하는 것을 포함할 수 있다.
- [0058] 예에서, 흐름(600)은 컴퓨터 시스템이 저장된 데이터를 비어 있거나 개방된 구역들에 맵핑하는 동작(640)을 포함한다. 일부 예에서, 이것은 2개-부분 동작이며, 여기서 제1 부분은 동작 640에서 발생하고(예컨대, 이것은, 도 3의 동작 340에서 기입 목적지 위치가 결정되는 방식과 유사함), 제2 부분은 (후술되는 바와 같이) 동작 650 후에 발생한다. 동작 630에서 식별된 맵핑 데이터는, 볼륨과 연관된 데이터가, 구역 세그먼트의 새로운 구역들, 새로 할당된 구역들, 또는 SMR 드라이브에 관한 조각모음에 이용하기 위해 자유롭게 유지되는 구역 세그먼트에 있을 수 있는 버퍼 스토리지에 맵핑되도록, 볼륨 메타데이터를 수정하는 것을 포함할 수 있다. 마찬가지로, 랜덤 액세스 메모리가 버퍼 메모리에 이용될 수 있다. 리맵핑은 저장된 데이터를 구역 세그먼트에 후속적으로 기입하기 위한 저장된 데이터의 구성을 결정하는 것을 포함할 수 있다.
- [0059] 예에서, 흐름(600)은 컴퓨터 시스템이 저장된 데이터를 비어 있거나 개방된 구역들에 기입하는 동작(650)을 포함한다. 저장된 데이터를 기입하는 것은, 도 3을 참조하여 더 상세히 설명되는 바와 같이, 흐름(300)의 동작들을 포함할 수 있다. 예를 들어, 시스템은 SMR 저장 서버(예를 들어, 도 1의 SMR 저장 서버(146))에 기입 커맨

드를 전송하여 저장된 데이터가 하나 이상의 비어 있는 개방 구역에 기입되게 할 수 있다. 조각모음을 위해 예약되어 있지 않은 구역들에서 기입 동작이 수행되는 경우, 데이터는 기입된 위치에 남아 있을 수 있으며, 맵핑은 동작 640에 의해 제공되는 바와 같이 유지될 수 있다.

[0060] 예에서, 동작 640에서 위에서 언급된 2개-부분 동작 중 제2 부분은, 동작 650에서 데이터가 복사된 후, 흐름(600)의 동작 655에서 발생한다. 예를 들어, 일단 데이터가 새로운 위치에서 안전하면, 블록 메타데이터가 업데이트된다(예컨대, 도 3의 동작 370과 유사).

[0061] 예에서, 흐름(600)은 컴퓨터 시스템이 저장된 데이터의 소스 구역을 리셋하는 동작(660)을 포함한다. 또한, 시스템은 후속 I/O 동작들(예를 들어, 도 1의 I/O 동작들(120))을 위해 구역 세그먼트의 소스 구역들을 리셋할 수 있다. 소스 구역들을 리셋하는 것은, 구역의 상태를 "가득참"으로부터 "비어 있음"으로 업데이트하거나 구역 세그먼트 내의 데이터의 새로운 구성을 반영하는 기타의 수정들을 포함할 수 있다.

[0062] 예에서, 흐름(600)은, 선택사항으로서, 컴퓨터 시스템이 과잉 구역들을 해제하는 동작(670)을 포함한다. 도 3을 참조하여 흐름(300)의 동작 330에 대해 설명된 바와 같이, 기입될 데이터가 참조된 구역 세그먼트의 용량을 초과하는 기입 동작들은, 추가적인 저장 용량을 제공하기 위해 새로운 구역 세그먼트를 할당하는 것을 포함할 수 있다. 이러한 방식으로, 조각모음은, 구역 세그먼트에서 데이터 용량의 미사용 부분을 감소시킴으로써, 데이터가 단일 구역 세그먼트에 저장될 수 있도록 만들 수 있다. 따라서, 이전에 구역 세그먼트에 대한 데이터를 저장하는데 이용되었던 구역 세그먼트들이 해제될 수 있다. 일부 실시예에서, 과잉 구역들은 640의 맵핑 동작들의 일부로서 식별될 수 있고, 동작 650을 참조하여 설명된 바와 같이 새로운 맵핑에 따라 저장된 데이터를 기입한 후에 해제될 수 있다.

[0063] 예에서, 흐름(600)은, 선택사항으로서, 컴퓨터 시스템이 조각모음 프로세스의 상태를 출력하는 동작(680)을 포함한다. 선행하는 도 3 내지 도 5의 출력 동작들에서와 같이, 동작 680은 자원 이용 상태를 자원 메타데이터의 중앙 데이터베이스에 보고하는 것, 및/또는 조각모음 프로세스의 성공/실패를 나타내는 것을 포함할 수 있다. 일부 실시예에서, 동작 680은 또한, 구역의 새로운 이용 비율을 확인하는 것 또는 타이머 또는 다른 트리거 조건을 리셋하는 것을 포함할 수 있다.

[0064] 구역-세그먼트 기반의 조각모음은 SMR 드라이브들을 통합하는 데이터베이스 시스템의 동작에 상당한 기술적 개선을 제공할 수 있다. 구역-세그먼트 조각모음의 기술들을 구현함으로써, SMR 드라이브 용량의 효율적인 이용이, 적어도 구역 세그먼트의 구역들에 걸쳐 데이터가 기입될 수 있는 구역 세그먼트의 이용 비율을 개선함으로써 개선될 수 있다. 이러한 방식으로, SMR 드라이브들의 특징인 부가-전용 기입 동작들에 의해 도입되는 제한들이 감소될 수 있다.

[0065] 위에서 언급된 바와 같이, IaaS(infrastructure as a service)는 한 특정한 유형의 클라우드 컴퓨팅이다. IaaS는, 공용 네트워크(예컨대, 인터넷)를 통해 가상화된 컴퓨팅 자원들을 제공하도록 구성될 수 있다. IaaS 모델에서, 클라우드 컴퓨팅 공급자는 인프라 컴포넌트들(예컨대, 서버들, 스토리지 디바이스들, 네트워크 노드들(예컨대, 하드웨어), 배치 소프트웨어, 플랫폼 가상화(예컨대, 하이퍼바이저 계층) 등)을 호스팅할 수 있다. 일부 경우에, IaaS 공급자는 또한, 이들 인프라 컴포넌트들과 함께 다양한 서비스(예컨대, 요금청구, 모니터링, 로깅, 보안, 부하 밸런싱 및 클러스터링 등)를 제공할 수 있다. 따라서, 이들 서비스들은 정책-구동형 수 있으므로, IaaS 사용자들은 애플리케이션 가용성 및 성능을 유지하기 위해 부하 밸런싱을 구동하는 정책들을 구현할 수 있을 것이다.

[0066] 일부 경우에, IaaS 고객들은 인터넷 등의 WAN(wide area network)을 통해 자원들 및 서비스들에 액세스할 수 있고, 클라우드 공급자의 서비스들을 이용하여 애플리케이션 스택의 나머지 요소들을 설치할 수 있다. 예를 들어, 사용자는 IaaS 플랫폼에 로그인하여 가상 머신(VM)들을 생성하고, 각각의 VM 상에 운영 체제(OS)를 설치하고, 데이터베이스들 등의 미들웨어를 배치하고, 워크로드들 및 백업들을 위한 스토리지 버킷들을 생성하고, 심지어 그 VM 내에 엔터프라이즈 소프트웨어를 설치할 수 있다. 그 다음, 고객은 공급자의 서비스들을 이용하여, 네트워크 트래픽 밸런싱, 애플리케이션 문제 해결, 성능 모니터링, 재해 복구 관리 등을 포함한 다양한 기능을 수행할 수 있다.

[0067] 대부분의 경우, 클라우드 컴퓨팅 모델은 클라우드 공급자의 참여를 요구할 것이다. 클라우드 공급자는, IaaS 제공(예컨대, 제공, 임대, 판매)을 전문으로 하는 제3자 서비스일 수 있지만, 반드시 그럴 필요는 없다. 엔터티는 또한, 사설 클라우드를 배치하여 인프라 서비스들의 자체 공급자가 될 수도 있다.

[0068] 일부 예에서, IaaS 배치는, 새로운 애플리케이션, 또는 애플리케이션의 새로운 버전을, 준비된 애플리케이션 서

버 등에 배치하는 프로세스이다. 이것은 서버를 준비하는 프로세스(예컨대, 라이브러리들, 데몬들의 설치 등)도 포함할 수 있다. 이것은, 종종 하이퍼바이저 계층(예컨대, 서버들, 스토리지, 네트워크 하드웨어, 및 가상화) 아래에서, 클라우드 공급자에 의해 관리된다. 따라서, 고객은, (예컨대, 요청시 가동될 수 있는) (예컨대, 셀프-서비스 가상 머신 등에서의) OS, 미들웨어 및/또는 애플리케이션 배치를 담당할 수 있다.

[0069] 일부 예에서, IaaS 프로비저닝(provisioning)이란, 이용할 컴퓨터들 또는 가상 호스트들을 취득하고, 나아가 이들 상에 필요한 라이브러리들 또는 서비스들을 설치하는 것을 지칭할 수 있다. 대부분의 경우, 배치는 프로비저닝을 포함하지 않으며, 프로비저닝은 먼저 수행될 필요가 있을 수 있다.

[0070] 일부 경우에, IaaS 프로비저닝에 대해 2개의 상이한 문제가 있다. 첫째, 어떤 것이든 실행되기 전에 초기 인프라 세트를 프로비저닝해야 하는 초기 문제가 있다. 둘째, 일단 모든 것이 프로비저닝되고 나면 기존 인프라를 진화(예컨대, 새로운 서비스들의 추가, 서비스들의 변경, 서비스들의 제거 등)시키는 문제가 있다. 일부 경우에, 인프라의 구성을 선언적으로 정의될 수 있게 함으로써 이들 2개의 문제가 해결될 수 있다. 즉, 인프라(예컨대, 필요한 컴포넌트들 및 이들이 상호작용 방법)는 하나 이상의 구성 파일에 의해 정의될 수 있다. 따라서, 인프라의 전체 토폴로지(예컨대, 어떤 자원들이 어떤 것에 의존하고, 이들 각각이 어떻게 함께 작동하는지)가 선언적으로 기술될 수 있다. 일부 경우에, 일단 토폴로지가 정의되고 나면, 구성 파일들에서 설명된 상이한 컴포넌트들을 생성 및/또는 관리하는 워크플로가 생성될 수 있다.

[0071] 일부 예에서, 인프라는 많은 상호접속된 요소를 가질 수 있다. 예를 들어, 코어 네트워크라고도 알려진, 하나 이상의 가상 사설 클라우드(VPC)(예컨대, 구성가능하거나 및/또는 공유된 컴퓨팅 자원들의 잠재적 온디맨드 풀(on-demand pool))가 있을 수 있다. 일부 예에서, 네트워크의 보안 셋업 방법과 하나 이상의 가상 머신(VM)을 정의하기 위해 프로비저닝되는 하나 이상의 보안 그룹 규칙이 있을 수도 있다. 부하 밸런서, 데이터베이스 등의 다른 인프라 요소들도 프로비저닝될 수 있다. 점점 더 많은 인프라 요소가 요구되고 그리고/또는 추가됨에 따라, 인프라가 점증적으로 발전할 수 있다.

[0072] 일부 경우에, 다양한 가상 컴퓨팅 환경에 걸친 인프라 코드의 배치를 가능케하기 위해 연속 배치 기술들이 채용될 수 있다. 추가로, 설명된 기술들은 이들 환경 내에서 인프라 관리를 가능케할 수 있다. 일부 예에서, 서비스 팀들은, 하나 이상의, 그러나 종종 많은, 상이한 프로덕션 환경(예컨대, 다양한 상이한 지리적 위치들에 걸쳐, 때로는 전 세계에 걸쳐)에 배치하려는 코드를 작성할 수 있다. 그러나, 일부 예에서, 코드가 배치될 인프라가 먼저 셋업되어야 한다. 일부 경우에, 프로비저닝이 수동으로 이루어질 수 있고, 프로비저닝 도구를 이용하여 자원들을 프로비저닝할 수 있고, 그리고/또는 일단 인프라가 프로비저닝되고 나면 배치 도구들을 이용하여 코드를 배치할 수 있다.

[0073] 도 7은 적어도 하나의 실시예에 따른 IaaS 아키텍처의 예시적인 패턴을 나타내는 블록도(700)이다. 서비스 운영자(702)는, 가상 클라우드 네트워크(VCN)(706) 및 보안 호스트 서브넷(708)을 포함할 수 있는 보안 호스트 테넌시(secure host tenancy)(704)에 통신가능하게 결합될 수 있다. 일부 예에서, 서비스 운영자(702)는, Microsoft Windows Mobile® 및/또는 iOS, Windows Phone, Android, BlackBerry 7, Palm OS 등의 다양한 모바일 운영 체제 등의 소프트웨어를 실행하고, 인터넷, 이메일, SMS(short message service), Blackberry® 또는 기타의 통신 프로토콜이 인에이블되어 있는, 휴대형 핸드헬드 디바이스들(예컨대, iPhone®, 셀룰러 전화, iPad®, 컴퓨팅 태블릿, PDA(personal digital assistant)) 또는 착용형 디바이스들(예컨대, Google Glass® 헤드 마운트형 디스플레이)일 수 있는 하나 이상의 클라이언트 컴퓨팅 디바이스들을 이용할 수 있다. 대안으로서, 클라이언트 컴퓨팅 디바이스들은, 예를 들어, 다양한 버전의 Microsoft Windows®, Apple Macintosh® 및/또는 Linux 운영 체제를 실행하는 개인용 컴퓨터들 및/또는 랩탑 컴퓨터들을 포함하는 범용 개인용 컴퓨터들일 수 있다. 클라이언트 컴퓨팅 디바이스들은, 예를 들어 Google Chrome OS 등의 다양한 GNU/Linux 운영 체제를 포함한 그러나 이것으로 제한되지 않는 다양한 상업적으로 이용가능한 UNIX® 또는 UNIX류 운영 체제 중 임의의 것을 실행하는 워크스테이션 컴퓨터들일 수 있다. 대안으로서, 또는 추가로, 클라이언트 컴퓨팅 디바이스들은, 쉘-클라이언트 컴퓨터(thin-client computer), 인터넷-가능형 게임 시스템(예컨대, Kinect® 제스처 입력 디바이스가 있거나 없는 Microsoft Xbox 게임 콘솔), 및/또는 VCN(706) 및/또는 인터넷에 액세스할 수 있는 네트워크를 통해 통신할 수 있는 개인 메시징 디바이스 등의, 임의의 다른 전자 디바이스일 수 있다.

[0074] VCN(706)은, SSH VCN(712)에 포함된 LPG(710)를 통해 SSH(secure shell) VCN(712)에 통신가능하게 결합될 수 있는 로컬 피어링 게이트웨이(LPG)(710)를 포함할 수 있다. SSH VCN(712)은 SSH 서브넷(714)을 포함할 수 있고, SSH VCN(712)은 제어 평면 VCN(716)에 포함된 LPG(710)를 통해 제어 평면 VCN(716)에 통신가능하게 결합될 수 있다. 또한, SSH VCN(712)은 LPG(710)를 통해 데이터 평면 VCN(718)에 통신가능하게 결합될 수 있다.

제어 평면 VCN(716) 및 데이터 평면 VCN(718)은 IaaS 공급자가 소유 및/또는 운영할 수 있는 서비스 테넌시(719)에 포함될 수 있다.

[0075] 제어 평면 VCN(716)은 경계 네트워크(예컨대, 기업 인트라넷과 외부 네트워크들 사이의 기업 네트워크의 일부분들)로서 작용하는 제어 평면 비무장 구역(DMZ) 티어(tier)(720)를 포함할 수 있다. DMZ-기반의 서버들은 제한된 책임들을 가질 수 있으며 보안 침해를 억제하는데 도움이 될 수 있다. 추가적으로, DMZ 티어(720)는, 하나 이상의 부하 밸런서(LB) 서브넷(들)(722), 앱 서브넷(들)(726)을 포함할 수 있는 제어 평면 앱 티어(724), 데이터베이스(DB) 서브넷(들)(730)(예컨대, 프론트엔드 DB 서브넷(들) 및/또는 백엔드 DB 서브넷(들))을 포함할 수 있는 제어 평면 데이터 티어(728)를 포함할 수 있다. 제어 평면 DMZ 티어(720)에 포함된 LB 서브넷(들)(722)은, 제어 평면 앱 티어(724)에 포함된 앱 서브넷(들)(726) 및 제어 평면 VCN(716)에 포함될 수 있는 인터넷 게이트웨이(734)에 통신가능하게 결합될 수 있고, 앱 서브넷(들)(726)은 제어 평면 데이터 티어(728)에 포함된 DB 서브넷(들)(730) 및 서비스 게이트웨이(736) 및 네트워크 주소 변환(NAT) 게이트웨이(738)에 통신가능하게 결합될 수 있다. 제어 평면 VCN(716)은 서비스 게이트웨이(736) 및 NAT 게이트웨이(738)를 포함할 수 있다.

[0076] 제어 평면 VCN(716)은, 앱 서브넷(들)(726)을 포함할 수 있는 데이터 평면 미러 앱 티어(740)를 포함할 수 있다. 데이터 평면 미러 앱 티어(740)에 포함된 앱 서브넷(들)(726)은, 컴퓨팅 인스턴스(744)를 실행할 수 있는 가상 네트워크 인터페이스 제어기(VNIC)(742)를 포함할 수 있다. 컴퓨팅 인스턴스(744)는, 데이터 평면 미러 앱 티어(740)의 앱 서브넷(들)(726)을 데이터 평면 앱 티어(746)에 포함될 수 있는 앱 서브넷(들)(726)에 통신가능하게 결합할 수 있다.

[0077] 데이터 평면 VCN(718)은, 데이터 평면 앱 티어(746), 데이터 평면 DMZ 티어(748), 및 데이터 평면 데이터 티어(750)를 포함할 수 있다. 데이터 평면 DMZ 티어(748)는, 데이터 평면 앱 티어(746)의 앱 서브넷(들)(726)과 데이터 평면 VCN(718)의 인터넷 게이트웨이(734)에 통신가능하게 결합될 수 있는 LB 서브넷(들)(722)을 포함할 수 있다. 앱 서브넷(들)(726)은, 데이터 평면 VCN(718)의 서비스 게이트웨이(736) 및 데이터 평면 VCN(718)의 NAT 게이트웨이(738)에 통신가능하게 결합될 수 있다. 데이터 평면 데이터 티어(750)는 또한, 데이터 평면 앱 티어(746)의 앱 서브넷(들)(726)에 통신가능하게 결합될 수 있는 DB 서브넷(들)(730)을 포함할 수 있다.

[0078] 제어 평면 VCN(716) 및 데이터 평면 VCN(718)의 인터넷 게이트웨이(734)는, 공용 인터넷(754)에 통신가능하게 결합될 수 있는 메타데이터 관리 서비스(752)에 통신가능하게 결합될 수 있다. 공용 인터넷(754)은, 제어 평면 VCN(716) 및 데이터 평면 VCN(718)의 NAT 게이트웨이(738)에 통신가능하게 결합될 수 있다. 제어 평면 VCN(716) 및 데이터 평면 VCN(718)의 서비스 게이트웨이(736)는 클라우드 서비스들(756)에 통신가능하게 결합될 수 있다.

[0079] 일부 예에서, 제어 평면 VCN(716) 또는 데이터 평면 VCN(718)의 서비스 게이트웨이(736)는, 공용 인터넷(754)을 거치지 않고 클라우드 서비스들(756)에 대한 애플리케이션 프로그래밍 인터페이스(API) 호출들을 할 수 있다. 서비스 게이트웨이(736)로부터의 클라우드 서비스들(756)에 대한 API 호출들은 일방향일 수 있다: 서비스 게이트웨이(736)는 클라우드 서비스들(756)에 대한 API 호출들을 할 수 있고, 클라우드 서비스들(756)은 요청된 데이터를 서비스 게이트웨이(736)에 보낼 수 있다. 그러나, 클라우드 서비스들(756)은 서비스 게이트웨이(736)에 대한 API 호출들을 개시하지 않을 수 있다.

[0080] 일부 예에서, 보안 호스트 테넌시(704)는 서비스 테넌시(719)에 직접 접속될 수 있으며, 이는 다른 방식으로 격리될 수 있다. 보안 호스트 서브넷(708)은, 다른 방식으로 격리된 시스템을 통한 양방향 통신을 가능하게 할 수 있는 LPG(710)를 통해 SSH 서브넷(714)과 통신할 수 있다. 보안 호스트 서브넷(708)을 SSH 서브넷(714)에 접속하는 것은, 서비스 테넌시(719) 내의 다른 엔티티들에 대한 보안 호스트 서브넷(708) 액세스 권한을 줄 수 있다.

[0081] 제어 평면 VCN(716)은 서비스 테넌시(719)의 사용자들이 원하는 자원들을 셋업하거나 다른 방식으로 프로비저닝하는 것을 허용할 수 있다. 제어 평면 VCN(716)에서 프로비저닝된 원하는 자원들은 데이터 평면 VCN(718)에서 배치되거나 다른 방식으로 이용될 수 있다. 일부 예에서, 제어 평면 VCN(716)은 데이터 평면 VCN(718)으로부터 격리될 수 있고, 제어 평면 VCN(716)의 데이터 평면 미러 앱 티어(740)는, 데이터 평면 미러 앱 티어(740) 및 데이터 평면 앱 티어(746)에 포함될 수 있는 VNIC들(742)을 통해 데이터 평면 VCN(718)의 데이터 평면 앱 티어(746)와 통신할 수 있다.

[0082] 일부 예에서, 시스템의 사용자들, 또는 고객들은, 요청들을 메타데이터 관리 서비스(752)에 통신할 수 있는 공

용 인터넷(754)을 통해, 예를 들어 생성, 관독, 업데이트 또는 삭제(CRUD) 동작들을 요청할 수 있다. 메타데이터 관리 서비스(752)는 인터넷 게이트웨이(734)를 통해 제어 평면 VCN(716)에 요청을 통신할 수 있다. 요청은 제어 평면 DMZ 티어(720)에 포함된 LB 서브넷(들)(722)에 의해 수신될 수 있다. LB 서브넷(들)(722)은 요청이 유효하다고 결정할 수 있고, 이 결정에 응답하여, LB 서브넷(들)(722)은 제어 평면 앱 티어(724)에 포함된 앱 서브넷(들)(726)에 요청을 전송할 수 있다. 요청이 검증되고 공용 인터넷(754)에 대한 호출을 요구한다면, 공용 인터넷(754)에 대한 호출은 공용 인터넷(754)을 호출할 수 있는 NAT 게이트웨이(738)에 전송될 수 있다. 요청에 의해 저장에 이루어지기를 원할 수 있는 메모리는 DB 서브넷(들)(730)에 저장될 수 있다.

[0083] 일부 예에서, 데이터 평면 미러 앱 티어(740)는 제어 평면 VCN(716)과 데이터 평면 VCN(718) 사이의 직접 통신을 용이화할 수 있다. 예를 들어, 데이터 평면 VCN(718)에 포함된 자원들에, 변경들, 업데이트들, 또는 구성에 대한 다른 적절한 수정들이 적용되기를 원할 수 있다. VNIC(742)를 통해, 제어 평면 VCN(716)은 데이터 평면 VCN(718)에 포함된 자원들과 직접 통신할 수 있고, 이로써 구성에 대한 변경들, 업데이트들, 또는 다른 적절한 수정들을 실행할 수 있다.

[0084] 일부 실시예에서, 제어 평면 VCN(716) 및 데이터 평면 VCN(718)은 서비스 테넌시(719)에 포함될 수 있다. 이 경우, 시스템의 사용자 또는 고객은, 제어 평면 VCN(716) 또는 데이터 평면 VCN(718)을 소유하거나 운영할 수 없다. 대신에, IaaS 공급자는 제어 평면 VCN(716) 및 데이터 평면 VCN(718)을 소유하거나 운영할 수 있으며, 이들 양쪽 모두는 서비스 테넌시(719)에 포함될 수 있다. 이 실시예는, 사용자들 또는 고객들이 다른 사용자들 또는 다른 고객들의 자원들과 상호작용하는 것을 방지할 수 있는 네트워크들의 격리를 가능하게 할 수 있다. 또한, 이 실시예는 시스템의 사용자들 또는 고객들이 저장을 위한 원하는 보안 레벨을 갖지 않을 수 있는 공용 인터넷(754)에 의존할 필요 없이 개인적으로 데이터베이스를 저장하는 것을 허용할 수 있다.

[0085] 다른 실시예들에서, 제어 평면 VCN(716)에 포함된 LB 서브넷(들)(722)은 서비스 게이트웨이(736)로부터 신호를 수신하도록 구성될 수 있다. 이 실시예에서, 제어 평면 VCN(716) 및 데이터 평면 VCN(718)은, 공용 인터넷(754)을 호출하지 않고 IaaS 공급자의 고객에 의해 호출되도록 구성될 수 있다. IaaS 공급자의 고객들은, 고객들이 이용하는 데이터베이스(들)가 IaaS 공급자에 의해 제어될 수 있고 공용 인터넷(754)으로부터 격리될 수 있는 서비스 테넌시(719)에 저장될 수 있기 때문에 이 실시예를 원할 수 있다.

[0086] 도 8은 적어도 하나의 실시예에 따른 IaaS 아키텍처의 또 다른 예시적인 패턴을 나타내는 블록도(800)이다. 서비스 운영자들(802)(예컨대, 도 7의 서비스 운영자들(702))은, 가상 클라우드 네트워크(VCN)(806)(예컨대, 도 7의 VCN(706)) 및 보안 호스트 서브넷(808)(예컨대, 도 7의 보안 호스트 서브넷(708))을 포함할 수 있는 보안 호스트 테넌시(804)(예컨대, 도 7의 보안 호스트 테넌시(704))에 통신가능하게 결합될 수 있다. VCN(806)은, SSH VCN(812)에 포함된 LPG(710)를 통해 보안 셸(SSH) VCN(812)(예컨대, 도 7의 SSH VCN(712))에 통신가능하게 결합될 수 있는 로컬 피어링 게이트웨이(LPG)(810)(예컨대, 도 7의 LPG(710))를 포함할 수 있다. SSH VCN(812)은 SSH 서브넷(814)(예컨대, 도 7의 SSH 서브넷(714))을 포함할 수 있고, SSH VCN(812)은 제어 평면 VCN(816)에 포함된 LPG(810)를 통해 제어 평면 VCN(816)(예컨대, 도 7의 제어 평면 VCN(716))에 통신가능하게 결합될 수 있다. 제어 평면 VCN(816)은 서비스 테넌시(819)(예컨대, 도 7의 서비스 테넌시(719))에 포함될 수 있고, 데이터 평면 VCN(818)(예컨대, 도 7의 데이터 평면 VCN(718))은 시스템의 사용자들 또는 고객들이 소유하거나 운영할 수 있는 고객 테넌시(821)에 포함될 수 있다.

[0087] 제어 평면 VCN(816)은, LB 서브넷(들)(822)(예컨대, 도 7의 LB 서브넷(들)(722))을 포함할 수 있는 제어 평면 DMZ 티어(820)(예컨대, 도 7의 제어 평면 DMZ 티어(720)), 앱 서브넷(들)(826)(예컨대, 도 7의 앱 서브넷(들)(726))을 포함할 수 있는 제어 평면 앱 티어(824)(예컨대, 도 7의 제어 평면 앱 티어(724)), 데이터베이스(DB) 서브넷(들)(830)(예컨대, 도 7의 DB 서브넷(들)(730))과 유사함을 포함할 수 있는 제어 평면 데이터 티어(828)(예컨대, 도 7의 제어 평면 데이터 티어(728))를 포함할 수 있다. 제어 평면 DMZ 티어(820)에 포함된 LB 서브넷(들)(822)은, 제어 평면 앱 티어(824)에 포함된 앱 서브넷(들)(826) 및 제어 평면 VCN(816)에 포함될 수 있는 인터넷 게이트웨이(834)(예컨대, 도 7의 인터넷 게이트웨이(734))에 통신가능하게 결합될 수 있고, 앱 서브넷(들)(826)은 제어 평면 데이터 티어(828)에 포함된 DB 서브넷(들)(830) 및 서비스 게이트웨이(836)(예컨대, 도 7의 서비스 게이트웨이) 및 네트워크 주소 변환(NAT) 게이트웨이(838)(예컨대, 도 7의 NAT 게이트웨이(738))에 통신가능하게 결합될 수 있다. 제어 평면 VCN(816)은 서비스 게이트웨이(836) 및 NAT 게이트웨이(838)를 포함할 수 있다.

[0088] 제어 평면 VCN(816)은, 앱 서브넷(들)(826)을 포함할 수 있는 데이터 평면 미러 앱 티어(840)(예컨대, 도 7의 데이터 평면 미러 앱 티어(740))를 포함할 수 있다. 데이터 평면 미러 앱 티어(840)에 포함된 앱 서브넷

(들)(826)은, 컴퓨팅 인스턴스(844)(예컨대, 도 7의 컴퓨팅 인스턴스(744)와 유사)를 실행할 수 있는 가상 네트워크 인터페이스 제어기(VNIC)(842)(예컨대, 742의 VNIC)를 포함할 수 있다. 컴퓨팅 인스턴스(844)는, 데이터 평면 미리 앱 티어(840)에 포함된 VNIC(842) 및 데이터 평면 앱 티어(846)에 포함된 VNIC(842)를 통해, 데이터 평면 미리 앱 티어(840)의 앱 서버(들)(826)과 데이터 평면 앱 티어(846)(예컨대, 도 7의 데이터 평면 앱 티어(746))에 포함될 수 있는 앱 서버(들)(826) 사이의 통신을 용이화할 수 있다.

[0089] 제어 평면 VCN(816)에 포함된 인터넷 게이트웨이(834)는, 공용 인터넷(854)(예컨대, 도 7의 공용 인터넷(754))에 통신가능하게 결합될 수 있는 메타데이터 관리 서비스(852)(예컨대, 도 7의 메타데이터 관리 서비스(752))에 통신가능하게 결합될 수 있다. 공용 인터넷(854)은 제어 평면 VCN(816)에 포함된 NAT 게이트웨이(838)에 통신가능하게 결합될 수 있다. 제어 평면 VCN(816)에 포함된 서비스 게이트웨이(836)는 클라우드 서비스들(856)(예컨대, 도 7의 클라우드 서비스들(756))에 통신가능하게 결합될 수 있다.

[0090] 일부 예에서, 데이터 평면 VCN(818)은 고객 테넌시(821)에 포함될 수 있다. 이 경우에, IaaS 공급자는 각각의 고객에게 제어 평면 VCN(816)을 제공할 수 있고, IaaS 공급자는, 각각의 고객에 대해, 서비스 테넌시(819)에 포함된 고유 컴퓨팅 인스턴스(844)를 셋업할 수 있다. 각각의 컴퓨팅 인스턴스(844)는, 서비스 테넌시(819)에 포함된 제어 평면 VCN(816)과 고객 테넌시(821)에 포함된 데이터 평면 VCN(818) 사이의 통신을 허용할 수 있다. 컴퓨팅 인스턴스(844)는, 서비스 테넌시(819)에 포함된 제어 평면 VCN(816)에서 프로비저닝되는 자원들이 고객 테넌시(821)에 포함된 데이터 평면 VCN(818)에 배치되거나 다른 방식으로 이용되는 것을 허용할 수 있다.

[0091] 다른 예들에서, IaaS 공급자의 고객은 고객 테넌시(821)에 상주하는 데이터베이스들을 가질 수 있다. 이 예에서, 제어 평면 VCN(816)은 앱 서버(들)(826)을 포함할 수 있는 데이터 평면 미리 앱 티어(840)를 포함할 수 있다. 데이터 평면 미리 앱 티어(840)는 데이터 평면 VCN(818)에 상주할 수 있지만, 데이터 평면 미리 앱 티어(840)는 데이터 평면 VCN(818)에 상주하지 않을 수 있다. 즉, 데이터 평면 미리 앱 티어(840)는 고객 테넌시(821)에 액세스할 수 있지만, 데이터 평면 미리 앱 티어(840)는 데이터 평면 VCN(818)에 존재하지 않거나 IaaS 공급자의 고객이 소유 또는 운영할 수 있다. 데이터 평면 미리 앱 티어(840)는 데이터 평면 VCN(818)을 호출하도록 구성될 수 있지만, 제어 평면 VCN(816)에 포함된 임의의 엔티티를 호출하도록 구성되지 않을 수 있다. 고객은 제어 평면 VCN(816)에서 프로비저닝되는 자원들을 데이터 평면 VCN(818)에서 배치하거나 다른 방식으로 이용하기를 원할 수 있으며, 데이터 평면 미리 앱 티어(840)는 고객의 원하는 자원 배치 또는 기타의 이용을 용이화할 수 있다.

[0092] 일부 실시예에서, IaaS 공급자의 고객은 데이터 평면 VCN(818)에 필터들을 적용할 수 있다. 이 실시예에서, 고객은 데이터 평면 VCN(818)이 무엇에 액세스할 수 있는지를 결정할 수 있고, 고객은 데이터 평면 VCN(818)으로부터 공용 인터넷(854)으로의 액세스를 제한할 수 있다. IaaS 공급자는, 필터들을 적용하거나 임의의 외부 네트워크들 또는 데이터베이스들에 대한 데이터 평면 VCN(818)의 액세스를 제어하지 못할 수도 있다. 고객에 의해 고객 테넌시(821)에 포함된 데이터 평면 VCN(818)에 필터들 및 제어들을 적용하는 것은, 데이터 평면 VCN(818)을 다른 고객들 및 공용 인터넷(854)으로부터 격리하는데 도움이 될 수 있다.

[0093] 일부 실시예에서, 클라우드 서비스들(856)은, 공용 인터넷(854), 제어 평면 VCN(816), 또는 데이터 평면 VCN(818) 상에 존재하지 않을 수도 있는 서비스들에 액세스하기 위해 서비스 게이트웨이(836)에 의해 호출될 수 있다. 클라우드 서비스들(856)과 제어 평면 VCN(816) 또는 데이터 평면 VCN(818) 사이의 접속은 살아 있거나 연속적이지 않을 수 있다. 클라우드 서비스들(856)은 IaaS 공급자가 소유하거나 운영하는 상이한 네트워크에 존재할 수 있다. 클라우드 서비스들(856)은 서비스 게이트웨이(836)로부터 호출들을 수신하도록 구성될 수 있고, 공용 인터넷(854)으로부터 호출들을 수신하지 않도록 구성될 수 있다. 일부 클라우드 서비스들(856)은 다른 클라우드 서비스들(856)로부터 격리될 수 있고, 제어 평면 VCN(816)은 제어 평면 VCN(816)과 동일한 지역에 있지 않을 수 있는 클라우드 서비스들(856)로부터 격리될 수 있다. 예를 들어, 제어 평면 VCN(816)은 "지역 1"에 위치할 수 있고, 클라우드 서비스 "배치 6"은 지역 1 및 "지역 2"에 위치할 수 있다. 배치 6에 대한 호출이 지역 1에 위치한 제어 평면 VCN(816)에 포함된 서비스 게이트웨이(836)에 의해 이루어진 경우, 그 호출은 지역 1의 배치 6에 전송될 수 있다. 이 예에서, 제어 평면 VCN(816) 또는 영역 1의 배치 6은, 영역 2의 배치 6에 통신가능하게 결합되지 않거나 다른 방식으로 통신하지 않을 수 있다.

[0094] 도 9은 적어도 하나의 실시예에 따른 IaaS 아키텍처의 또 다른 예시적인 패턴을 나타내는 블록도(900)이다. 서비스 운영자들(902)(예컨대, 도 7의 서비스 운영자들(702))은, 가상 클라우드 네트워크(VCN)(906)(예컨대, 도 7의 VCN(706)) 및 보안 호스트 서버(들)(908)(예컨대, 도 7의 보안 호스트 서버(들)(708))을 포함할 수 있는 보안 호스트 테넌시(904)(예컨대, 도 7의 보안 호스트 테넌시(704))에 통신가능하게 결합될 수 있다. VCN(906)은, SSH

VCN(912)에 포함된 LPG(910)를 통해 SSH VCN(912)(예컨대, 도 7의 SSH VCN(712))에 통신가능하게 결합될 수 있는 LPG(910)(예컨대, 도 7의 LPG(710))를 포함할 수 있다. SSH VCN(912)은 SSH 서버넷(914)(예컨대, 도 7의 SSH 서버넷(714))을 포함할 수 있고, SSH VCN(912)은 제어 평면 VCN(916)에 포함된 LPG(910)를 통해 제어 평면 VCN(916)(예컨대, 도 7의 제어 평면 VCN(716))에, 및 데이터 평면 VCN(918)에 포함된 LPG(910)를 통해 데이터 평면 VCN(918)(예컨대, 도 7의 데이터 평면(718))에 통신가능하게 결합될 수 있다. 제어 평면 VCN(916) 및 데이터 평면 VCN(918)은 서비스 테넌시(919)(예컨대, 도 7의 서비스 테넌시(719))에 포함될 수 있다.

[0095] 제어 평면 VCN(916)은, 부하 밸런서(LB) 서버넷(들)(922)(예컨대, 도 7의 LB 서버넷(들)(722))을 포함할 수 있는 제어 평면 DMZ 티어(920)(예컨대, 도 7의 제어 평면 DMZ 티어(720)), 앱 서버넷(들)(926)(예컨대, 도 7의 앱 서버넷(들)(726)과 유사)을 포함할 수 있는 제어 평면 앱 티어(924)(예컨대, 도 7의 제어 평면 앱 티어(724)), DB 서버넷(들)(930)을 포함할 수 있는 제어 평면 데이터 티어(928)(예컨대, 도 7의 제어 평면 데이터 티어(728))를 포함할 수 있다. 제어 평면 DMZ 티어(920)에 포함된 LB 서버넷(들)(922)은, 제어 평면 앱 티어(924)에 포함된 앱 서버넷(들)(926) 및 제어 평면 VCN(916)에 포함될 수 있는 인터넷 게이트웨이(934)(예컨대, 도 7의 인터넷 게이트웨이(734))에 통신가능하게 결합될 수 있고, 앱 서버넷(들)(926)은 제어 평면 데이터 티어(928)에 포함된 DB 서버넷(들)(930) 및 서비스 게이트웨이(936)(예컨대, 도 7의 서비스 게이트웨이) 및 네트워크 주소 변환(NAT) 게이트웨이(938)(예컨대, 도 7의 NAT 게이트웨이(738))에 통신가능하게 결합될 수 있다. 제어 평면 VCN(916)은 서비스 게이트웨이(936) 및 NAT 게이트웨이(938)를 포함할 수 있다.

[0096] 데이터 평면 VCN(918)은, 데이터 평면 앱 티어(946)(예컨대, 도 7의 데이터 평면 앱 티어(746)), 데이터 평면 DMZ 티어(948)(예컨대, 도 7의 데이터 평면 DMZ 티어(748)), 및 데이터 평면 데이터 티어(950)(예컨대, 도 7의 데이터 평면 데이터 티어(750))를 포함할 수 있다. 데이터 평면 DMZ 티어(948)는, 데이터 평면 앱 티어(946)의 신뢰받는 앱 서버넷(들)(960) 및 비-신뢰받는 앱 서버넷(들)(962)에 통신가능하게 결합될 수 있는 LB 서버넷(들)(922) 및 데이터 평면 VCN(918)에 포함된 인터넷 게이트웨이(934)를 포함할 수 있다. 신뢰받는 앱 서버넷(들)(960)은, 데이터 평면 VCN(918)에 포함된 서비스 게이트웨이(936), 데이터 평면 VCN(918)에 포함된 NAT 게이트웨이(938), 및 데이터 평면 데이터 티어(950)에 포함된 DB 서버넷(들)(930)에 통신가능하게 결합될 수 있다. 비-신뢰받는 앱 서버넷(들)(962)은 데이터 평면 VCN(918)에 포함된 서비스 게이트웨이(936) 및 데이터 평면 데이터 티어(950)에 포함된 DB 서버넷(들)(930)에 통신가능하게 결합될 수 있다. 데이터 평면 데이터 티어(950)는, 데이터 평면 VCN(918)에 포함된 서비스 게이트웨이(936)에 통신가능하게 결합될 수 있는 DB 서버넷(들)(930)을 포함할 수 있다.

[0097] 비-신뢰받는 앱 서버넷(들)(962)은 테넌트 가상 머신들(VMs)(966(1)-(N))에 통신가능하게 결합될 수 있는 하나 이상의 1차 VNIC(964(1)-(N))를 포함할 수 있다. 각각의 테넌트 VM(966(1)-(N))은 각자의 고객 테넌시(970(1)-(N))에 포함될 수 있는 각자의 컨테이너 출구 VCN들(968(1)-(N))에 포함될 수 있는 각자의 앱 서버넷(967(1)-(N))에 통신가능하게 결합될 수 있다. 각자의 2차 VNIC들(972(1)-(N))은, 데이터 평면 VCN(918)에 포함된 비-신뢰받는 앱 서버넷(들)(962)과 컨테이너 출구 VCN들(968(1)-(N))에 포함된 앱 서버넷 사이의 통신을 용이화할 수 있다. 각각의 컨테이너 출구 VCN들(968(1)-(N))은 공용 인터넷(954)(예컨대, 도 7의 공용 인터넷(754))에 통신가능하게 결합될 수 있는 NAT 게이트웨이(938)를 포함할 수 있다.

[0098] 제어 평면 VCN(916)에 포함되고 데이터 평면 VCN(918)에 포함된 인터넷 게이트웨이(934)는, 공용 인터넷(954)에 통신가능하게 결합될 수 있는 메타데이터 관리 서비스(952)(예컨대, 도 7의 메타데이터 관리 시스템(752))에 통신가능하게 결합될 수 있다. 공용 인터넷(954)은, 제어 평면 VCN(916)에 포함되고 데이터 평면 VCN(918)에 포함된 NAT 게이트웨이(938)에 통신가능하게 결합될 수 있다. 제어 평면 VCN(916)에 포함되고 데이터 평면 VCN(918)에 포함된 서비스 게이트웨이(936)는 클라우드 서비스들(956)에 통신가능하게 결합될 수 있다.

[0099] 일부 실시예에서, 데이터 평면 VCN(918)은 고객 테넌시들(970)과 통합될 수 있다. 이 통합은, 코드를 실행할 때 지원이 필요한 경우 등의 일부 경우에 IaaS 공급자의 고객들에게 유용하거나 바람직할 수 있다. 고객은, 파괴적이거나 다른 고객 자원들과 통신하거나 바람직하지 않은 효과들을 초래할 수 있는 실행 코드를 제공할 수 있다. 이에 응답하여, IaaS 공급자는, 고객이 IaaS 공급자에게 제공한 코드를 실행할지를 결정할 수 있다.

[0100] 일부 예에서, IaaS 공급자의 고객은 IaaS 공급자에게 임시 네트워크 액세스 권한을 부여하고 데이터 평면 티어 앱(946)에 첨부될 기능을 요청할 수 있다. 기능을 실행하기 위한 코드는 VM들(966(1)-(N))에서 실행될 수 있고, 코드는 데이터 평면 VCN(918)의 다른 곳에서 실행되도록 구성되지 않을 수 있다. 각각의 VM(966(1)-(N))은 하나의 고객 테넌시(970)에 접속될 수 있다. VM들(966(1)-(N))에 포함된 각자의 컨테이너들(971(1)-(N))은 코드를 실행하도록 구성될 수 있다. 이 경우, 이중 격리가 있을 수 있고(예컨대, 코드를 실행하는 컨테

이너들(971(1)-(N)), 여기서 컨테이너들(971(1)-(N))이 적어도 비-신뢰받는 앱 서브넷(들)(962)에 포함된 VM(966(1)-(N))에 포함될 수 있음), 이것은 부정확하거나 달리 바람직하지 않은 코드가 IaaS 공급자의 네트워크를 손상시키거나 상이한 고객의 네트워크를 손상시키는 것을 방지하는데 도움이 될 수 있다. 컨테이너들(971(1)-(N))은 고객 테넌시(970)에 통신가능하게 결합될 수 있고 고객 테넌시(970)로부터 데이터를 전송하거나 수신하도록 구성될 수 있다. 컨테이너들(971(1)-(N))은 데이터 평면 VCN(918)의 임의의 다른 엔티티로부터 데이터를 전송하거나 수신하도록 구성되지 않을 수 있다. 코드 실행이 완료되면, IaaS 공급자는 컨테이너들 971(1)-(N)을 죽이거나 다른 방식으로 폐기할 수 있다.

[0101] 일부 실시예에서, 신뢰받는 앱 서브넷(들)(960)은 IaaS 공급자가 소유하거나 운영할 수 있는 코드를 실행할 수 있다. 이 실시예에서, 신뢰받는 앱 서브넷(들)(960)은 DB 서브넷(들)(930)에 통신가능하게 결합될 수 있고 DB 서브넷(들)(930)에서 CRUD 동작들을 실행하도록 구성될 수 있다. 비-신뢰받는 앱 서브넷(들)(962)은 DB 서브넷(들)(930)에 통신가능하게 결합될 수 있지만, 이 실시예에서, 비-신뢰받는 앱 서브넷(들)은 DB 서브넷(들)(930)에서 판독 동작들을 실행하도록 구성될 수 있다. 각각의 고객의 VM들 966(1)-(N)에 포함될 수 있고 고객으로부터의 코드를 실행할 수 있는 컨테이너들 971(1)-(N)은 DB 서브넷(들)(930)과 통신가능하게 결합되지 않을 수 있다.

[0102] 다른 실시예들에서, 제어 평면 VCN(916) 및 데이터 평면 VCN(918)은 통신가능하게 직접 결합되지 않을 수 있다. 이 실시예에서, 제어 평면 VCN(916)과 데이터 평면 VCN(918) 사이에 어떠한 직접적인 통신도 없을 수 있다. 그러나, 통신은 적어도 하나 이상의 방법을 통해 간접적으로 발생할 수 있다. LPG(910)는 제어 평면 VCN(916)과 데이터 평면 VCN(918) 사이의 통신을 용이화할 수 있는 IaaS 공급자에 의해 확립될 수 있다. 또 다른 예에서, 제어 평면 VCN(916) 또는 데이터 평면 VCN(918)은 서비스 게이트웨이(936)를 통해 클라우드 서비스들(956)을 호출할 수 있다. 예를 들어, 제어 평면 VCN(916)으로부터의 클라우드 서비스들(956)에 대한 호출은 데이터 평면 VCN(918)과 통신할 수 있는 서비스에 대한 요청을 포함할 수 있다.

[0103] 도 10은 적어도 하나의 실시예에 따른 IaaS 아키텍처의 또 다른 예시적인 패턴을 나타내는 블록도(1000)이다. 서비스 운영자들(1002)(예컨대, 도 7의 서비스 운영자들(702))은, 가상 클라우드 네트워크(VCN)(1006)(예컨대, 도 7의 VCN(706)) 및 보안 호스트 서브넷(1008)(예컨대, 도 7의 보안 호스트 서브넷(708))을 포함할 수 있는 보안 호스트 테넌시(1004)(예컨대, 도 7의 보안 호스트 테넌시(704))에 통신가능하게 결합될 수 있다. VCN(1006)은, SSH VCN(1012)에 포함된 LPG(1010)를 통해 SSH VCN(1012)(예컨대, 도 7의 SSH VCN(712))에 통신가능하게 결합될 수 있는 LPG(1010)(예컨대, 도 7의 LPG(710))를 포함할 수 있다. SSH VCN(1012)은 SSH 서브넷(1014)(예컨대, 도 7의 SSH 서브넷(714))을 포함할 수 있고, SSH VCN(1012)은 제어 평면 VCN(1016)에 포함된 LPG(1010)를 통해 제어 평면 VCN(1016)(예컨대, 도 7의 제어 평면 VCN(716))에, 및 데이터 평면 VCN(1018)에 포함된 LPG(1010)를 통해 데이터 평면 VCN(1018)(예컨대, 도 7의 데이터 평면(718))에 통신가능하게 결합될 수 있다. 제어 평면 VCN(1016) 및 데이터 평면 VCN(1018)은 서비스 테넌시(1019)(예컨대, 도 7의 서비스 테넌시(719))에 포함될 수 있다.

[0104] 제어 평면 VCN(1016)은, LB 서브넷(들)(1022)(예컨대, 도 7의 LB 서브넷(들)(722))을 포함할 수 있는 제어 평면 DMZ 티어(1020)(예컨대, 도 7의 제어 평면 DMZ 티어(720)), 앱 서브넷(들)(1026)(예컨대, 도 7의 앱 서브넷(들)(726))을 포함할 수 있는 제어 평면 앱 티어(1024)(예컨대, 도 7의 제어 평면 앱 티어(724)), DB 서브넷(들)(1030)(예컨대, 도 9의 DB 서브넷(들)(930))을 포함할 수 있는 제어 평면 데이터 티어(1028)(예컨대, 도 7의 제어 평면 데이터 티어(728))를 포함할 수 있다. 제어 평면 DMZ 티어(1020)에 포함된 LB 서브넷(들)(1022)은, 제어 평면 앱 티어(1024)에 포함된 앱 서브넷(들)(1026) 및 제어 평면 VCN(1016)에 포함될 수 있는 인터넷 게이트웨이(1034)(예컨대, 도 7의 인터넷 게이트웨이(734))에 통신가능하게 결합될 수 있고, 앱 서브넷(들)(1026)은 제어 평면 데이터 티어(1028)에 포함된 DB 서브넷(들)(1030) 및 서비스 게이트웨이(1036)(예컨대, 도 7의 서비스 게이트웨이) 및 네트워크 주소 변환(NAT) 게이트웨이(1038)(예컨대, 도 7의 NAT 게이트웨이(738))에 통신가능하게 결합될 수 있다. 제어 평면 VCN(1016)은 서비스 게이트웨이(1036) 및 NAT 게이트웨이(1038)를 포함할 수 있다.

[0105] 데이터 평면 VCN(1018)은, 데이터 평면 앱 티어(1046)(예컨대, 도 7의 데이터 평면 앱 티어(746)), 데이터 평면 DMZ 티어(1048)(예컨대, 도 7의 데이터 평면 DMZ 티어(748)), 및 데이터 평면 데이터 티어(1050)(예컨대, 도 7의 데이터 평면 데이터 티어(750))를 포함할 수 있다. 데이터 평면 DMZ 티어(1048)은, 데이터 평면 앱 티어(1046)의 신뢰받는 앱 서브넷(들)(1060)(예컨대, 도 9의 신뢰받는 앱 서브넷(들)(960)) 및 비-신뢰받는 앱 서브넷(들)(1062)(예컨대, 도 9의 비-신뢰받는 앱 서브넷(들)(962)) 및 데이터 평면 VCN(1018)에 포함된 인터넷 게이트웨이(1034)에 통신가능하게 결합될 수 있는 LB 서브넷(들)(1022)을 포함할 수 있다. 신뢰받는 앱 서브넷

(들)(1060)은, 데이터 평면 VCN(1018)에 포함된 서비스 게이트웨이(1036), 데이터 평면 VCN(1018)에 포함된 NAT 게이트웨이(1038), 및 데이터 평면 데이터 티어(1050)에 포함된 DB 서버넷(들)(1030)에 통신가능하게 결합될 수 있다. 비-신뢰받는 앱 서버넷(들)(1062)은 데이터 평면 VCN(1018)에 포함된 서비스 게이트웨이(1036) 및 데이터 평면 데이터 티어(1050)에 포함된 DB 서버넷(들)(1030)에 통신가능하게 결합될 수 있다. 데이터 평면 데이터 티어(1050)는, 데이터 평면 VCN(1018)에 포함된 서비스 게이트웨이(1036)에 통신가능하게 결합될 수 있는 DB 서버넷(들)(1030)을 포함할 수 있다.

[0106] 비-신뢰받는 앱 서버넷(들)(1062)은, 비-신뢰받는 앱 서버넷(들)(1062) 내에 상주하는 테넌트 가상 머신(VM)들(1066(1)-(N))에 통신가능하게 결합될 수 있는 1차 VNIC(1064(1)-(N))를 포함할 수 있다. 각각의 테넌트 VM(1066(1)-(N))은 각자의 컨테이너(1067(1)-(N))에서 코드를 실행할 수 있고, 컨테이너 출구 VCN(1068)에 포함될 수 있는 데이터 평면 앱 티어(1046)에 포함될 수 있는 앱 서버넷(1026)에 통신가능하게 결합될 수 있다. 각각의 2차 VNIC들(1072(1)-(N))은, 데이터 평면 VCN(1018)에 포함된 비-신뢰받는 앱 서버넷(들)(1062)과 컨테이너 출구 VCN(1068)에 포함된 앱 서버넷 사이의 통신을 용이화할 수 있다. 컨테이너 출구 VCN은 공용 인터넷(1054)(예컨대, 도 7의 공용 인터넷(754))에 통신가능하게 결합될 수 있는 NAT 게이트웨이(1038)를 포함할 수 있다.

[0107] 제어 평면 VCN(1016)에 포함되고 데이터 평면 VCN(1018)에 포함된 인터넷 게이트웨이(1034)는, 공용 인터넷(1054)에 통신가능하게 결합될 수 있는 메타데이터 관리 서비스(1052)(예컨대, 도 7의 메타데이터 관리 시스템(752))에 통신가능하게 결합될 수 있다. 공용 인터넷(1054)은, 제어 평면 VCN(1016)에 포함되고 데이터 평면 VCN(1018)에 포함된 NAT 게이트웨이(1038)에 통신가능하게 결합될 수 있다. 제어 평면 VCN(1016)에 포함되고 데이터 평면 VCN(1018)에 포함된 서비스 게이트웨이(1036)는 클라우드 서비스들(1056)에 통신가능하게 결합될 수 있다.

[0108] 일부 예에서, 도 10의 블록도(1000)의 아키텍처에 의해 나타낸 패턴은 도 9의 블록도(900)의 아키텍처에 의해 나타낸 패턴에 대한 예외로서 간주될 수 있고, IaaS 공급자가 고객과 직접 통신할 수 없다면(예컨대, 접속이 끊긴 지역) IaaS 공급자의 고객에게 바람직할 수 있다. 각각의 고객에 대한 VM들(1066(1)-(N))에 포함된 각자의 컨테이너들(1067(1)-(N))은 고객에 의해 실시간으로 액세스될 수 있다. 컨테이너들(1067(1)-(N))은, 컨테이너 출구 VCN(1068)에 포함될 수 있는 데이터 평면 앱 티어(1046)의 앱 서버넷(들)(1026)에 포함된 각자의 2차 VNIC들(1072(1)-(N))을 호출하도록 구성될 수 있다. 2차 VNIC들(1072(1)-(N))은 호출들을 공용 인터넷(1054)에 전송할 수 있는 NAT 게이트웨이(1038)에 호출들을 전송할 수 있다. 이 예에서, 고객에 의해 실시간으로 액세스될 수 있는 컨테이너들(1067(1)-(N))은 제어 평면 VCN(1016)으로부터 격리될 수 있고 데이터 평면 VCN(1018)에 포함된 다른 엔티티들로부터 격리될 수 있다. 컨테이너들(1067(1)-(N))은 또한, 다른 고객들로부터의 자원들로부터 격리될 수 있다.

[0109] 다른 예들에서, 고객은 클라우드 서비스들(1056)을 호출하기 위해 컨테이너들(1067(1)-(N))을 이용할 수 있다. 이 예에서, 고객은 클라우드 서비스들(1056)로부터 서비스를 요청하는 컨테이너들(1067(1)-(N))에서 코드를 실행할 수 있다. 컨테이너들(1067(1)-(N))은, 공용 인터넷(1054)에 요청을 전송할 수 있는 NAT 게이트웨이에 요청을 전송할 수 있는 2차 VNIC들(1072(1)-(N))에 이 요청을 전송할 수 있다. 공용 인터넷(1054)은 인터넷 게이트웨이(1034)를 통해 제어 평면 VCN(1016)에 포함된 LB 서버넷(들)(1022)에 요청을 전송할 수 있다. 요청이 유효하다고 결정하는 것에 응답하여, LB 서버넷(들)은 서비스 게이트웨이(1036)를 통해 클라우드 서비스들(1056)에 요청을 전송할 수 있는 앱 서버넷(들)(1026)에 요청을 전송할 수 있다.

[0110] 도면들에 도시된 IaaS 아키텍처들(700, 800, 900, 1000)은 도시된 것들 이외의 컴포넌트들을 가질 수 있다는 것을 이해해야 한다. 또한, 도면들에 도시된 실시예들은 본 개시내용의 한 실시예를 통합할 수 있는 클라우드 인프라 시스템의 일부 예일 뿐이다. 일부 다른 실시예에서, IaaS 시스템들은 도면들에 도시된 것보다 더 많거나 더 적은 수의 컴포넌트를 가질 수 있거나, 2개 이상의 컴포넌트를 결합할 수 있거나, 컴포넌트들의 상이한 구성 또는 배열을 가질 수 있다.

[0111] 소정의 실시예들에서, 여기서 설명된 IaaS 시스템들은, 셀프 서비스, 가입-기반의, 탄력적으로 스케일링가능하고, 신뢰할 수 있고, 가용성이 높으며, 안전한 방식으로, 고객에게 제공되는 일련의 애플리케이션, 미들웨어 및 데이터베이스 서비스 제품들을 포함할 수 있다. 이러한 IaaS 시스템의 한 예는 본 출원의 양수인에 의해 제공되는 Oracle Cloud Infrastructure(OCI)이다.

[0112] 도 11은 다양한 실시예가 구현될 수 있는 한 예시적인 컴퓨터 시스템(1100)을 나타낸다. 시스템(1100)은 전술된 임의의 컴퓨터 시스템을 구현하는데 이용될 수 있다. 도면에 도시된 바와 같이, 컴퓨터 시스템(1100)은 버

스 서브시스템(1102)을 통해 다수의 주변 서브시스템과 통신하는 처리 유닛(1104)을 포함한다. 이들 주변 서브시스템들은, 처리 가속 유닛(1106), I/O 서브시스템(1108), 저장 서브시스템(1118) 및 통신 서브시스템(1124)을 포함할 수 있다. 저장 서브시스템(1118)은 유형의(tangible) 컴퓨터 판독가능한 저장 매체(1122) 및 시스템 메모리(1110)를 포함한다.

[0113] 버스 서브시스템(1102)은 컴퓨터 시스템(1100)의 다양한 컴포넌트 및 서브시스템이 의도한 대로 서로 통신하게 하는 메커니즘을 제공한다. 버스 서브시스템(1102)이 개략적으로 단일 버스로서 도시되어 있지만, 버스 서브시스템의 대안적인 실시예들은 여러 버스를 이용할 수 있다. 버스 서브시스템(1102)은, 메모리 버스 또는 메모리 제어기, 주변기기 버스, 및 임의의 다양한 버스 아키텍처를 이용하는 로컬 버스를 포함한 여러 유형들의 버스 구조들 중 임의의 것일 수 있다. 예를 들어, 이러한 아키텍처들은, ISA(Industry Standard Architecture) 버스, MCA(Micro Channel Architecture) 버스, EISA(Enhanced ISA) 버스, VESA(Video Electronics Standards Association) 로컬 버스, 및 IEEE P1386.1 표준에 따라 제조된 Mezzanine 버스로서 구현될 수 있는 PCI(Peripheral Component Interconnect) 버스를 포함할 수 있다.

[0114] 하나 이상의 집적 회로(예컨대, 종래의 마이크로프로세서 또는 마이크로제어기)로 구현될 수 있는 처리 유닛(1104)은 컴퓨터 시스템(1100)의 동작을 제어한다. 하나 이상의 프로세서가 처리 유닛(1104)에 포함될 수 있다. 이들 프로세서들은 단일 코어 또는 다중코어 프로세서들을 포함할 수 있다. 소정의 실시예들에서, 처리 유닛(1104)은, 각각의 처리 유닛에 포함된 단일 또는 다중코어 프로세서들을 갖는 하나 이상의 독립적인 처리 유닛(1132 및/또는 1134)으로서 구현될 수 있다. 다른 실시예들에서, 처리 유닛(1104)은 또한, 2개의 듀얼 코어 프로세서를 단일 칩으로 통합함으로써 형성된 쿼드 코어 처리 유닛으로서 구현될 수 있다.

[0115] 다양한 실시예에서, 처리 유닛(1104)은 프로그램 코드에 응답하여 다양한 프로그램을 실행할 수 있고 여러 동시 실행되는 프로그램들 또는 프로세스들을 유지할 수 있다. 임의의 주어진 시간에, 실행될 프로그램 코드의 일부 또는 전부는 프로세서(들)(1104) 및/또는 스토리지 서브시스템(1118)에 상주할 수 있다. 적절한 프로그래밍을 통해, 프로세서(들)(1104)는 위에서 설명된 다양한 기능을 제공할 수 있다. 컴퓨터 시스템(1100)은, 디지털 신호 프로세서(DSP), 특별 목적 프로세서, 및/또는 기타 등등을 포함할 수 있는 처리 가속 유닛(1106)을 추가로 포함할 수 있다.

[0116] I/O 서브시스템(1108)은 사용자 인터페이스 입력 디바이스들 및 사용자 인터페이스 출력 디바이스들을 포함할 수 있다. 사용자 인터페이스 입력 디바이스들은, 키보드, 마우스 또는 트랙볼 등의 포인팅 디바이스들, 디스플레이에 통합된 터치패드 또는 터치 스크린, 스크롤 휠, 클릭 휠, 다이얼, 버튼, 스위치, 키패드, 음성 명령어 인식 시스템들을 갖춘 오디오 입력 디바이스들, 마이크로폰들, 및 기타 유형들의 입력 디바이스들을 포함할 수 있다. 사용자 인터페이스 입력 디바이스들은, 예를 들어, 제스처들과 음성 명령어들을 이용하는 자연스러운 사용자 인터페이스를 통해, 사용자들이 Microsoft Xbox® 360 게임 제어기 등의 입력 디바이스를 제어하고 상호작용할 수 있게 해주는 Microsoft Kinect® 모션 센서 등의 모션 감지 및/또는 제스처 인식 디바이스들을 포함할 수 있다. 사용자 인터페이스 입력 디바이스들은 또한, 사용자의 눈 활동(예컨대, 사진 촬영 및/또는 메뉴 선택 중 '깜박임')을 검출하고 눈 제스처들을 입력 디바이스(예컨대, Google Glass®)의 입력으로 변환하는 Google Glass® 깜박임 검출기 등의 눈 제스처 인식 디바이스들을 포함할 수 있다. 추가로, 사용자 인터페이스 입력 디바이스들은, 사용자들이 음성 명령어들을 통해 음성 인식 시스템들(예컨대, Siri® 네비게이터)과 상호작용할 수 있게 하는 음성 인식 감지 디바이스들을 포함할 수 있다.

[0117] 사용자 인터페이스 입력 디바이스들은 또한, 제한없이, 3차원(3D) 마우스, 조이스틱들 또는 포인팅 스틱들, 게임 패드들 및 그래픽 태블릿들, 스피커들, 디지털 카메라들, 디지털 캠코더들, 휴대형 미디어 재생기들, 웹캠들, 이미지 스캐너들, 지문 스캐너들, 바코드 판독기 3D 스캐너들, 3D 프린터들, 레이저 거리 측정기들, 및 시선 추적 디바이스들 등의 오디오/비주얼 디바이스들을 포함할 수 있다. 또한, 사용자 인터페이스 입력 디바이스들은, 예를 들어, 컴퓨터 단층 촬영, 자기 공명 촬영, 위치 방출 단층 촬영, 의료용 초음파 디바이스들 등의 의료 촬영 입력 디바이스들을 포함할 수 있다. 사용자 인터페이스 입력 디바이스들은 또한, 예를 들어, MIDI 키보드들, 디지털 악기들 등의 오디오 입력 디바이스들을 포함할 수 있다.

[0118] 사용자 인터페이스 출력 디바이스들은, 디스플레이 서브시스템, 표시등, 또는 오디오 출력 디바이스들 등의 비-시각적 디스플레이들을 포함할 수 있다. 디스플레이 서브시스템은, 음극선관(CRT), 액정 디스플레이(LCD) 또는 플라즈마 디스플레이 등을 이용하는 것들 등의 평면-패널 디바이스, 프로젝션 디바이스, 터치 스크린 등 일 수 있다. 일반적으로, "출력 디바이스"라는 용어의 사용은, 컴퓨터 시스템(1100)으로부터 사용자 또는 다른 컴퓨터로 정보를 출력하기 위한 모든 가능한 유형의 디바이스들 및 메커니즘들을 포함하기 위한 의도이다. 예를 들

어, 사용자 인터페이스 출력 디바이스들은, 제한없이, 모니터들, 프린터들, 스피커들, 헤드폰들, 자동차 내비게이션 시스템들, 플로터들, 음성 출력 디바이스들 및 모뎀들 등의, 텍스트, 그래픽 및 오디오/비디오 정보를 시각적으로 전달하는 다양한 디스플레이 디바이스를 포함할 수 있다.

[0119] 컴퓨터 시스템(1100)은, 현재는 시스템 메모리(1110) 내에 위치하는 것으로 도시된, 소프트웨어 요소들을 포함하는 스토리지 서브시스템(1118)을 포함할 수 있다. 시스템 메모리(1110)는, 로딩가능하고 처리 유닛(1104)에서 실행가능한 프로그램 명령어들뿐만 아니라, 이들 프로그램들의 실행 동안 생성된 데이터를 저장할 수 있다.

[0120] 컴퓨터 시스템(1100)의 구성 및 유형에 따라, 시스템 메모리(1110)는, (RAM(random access memory) 등의) 휘발성 및/또는 (ROM(read-only memory), 플래시 메모리 등의) 비휘발성일 수 있다. RAM은 전형적으로 처리 유닛(1104)에 의해 즉시 액세스가능하거나 및/또는 현재 작동되고 실행되는 데이터 및/또는 프로그램 모듈들을 포함한다. 일부 구현에서, 시스템 메모리(1110)는, SRAM(static random access memory) 또는 DRAM(dynamic random access memory) 등의 여러 상이한 유형들의 메모리를 포함할 수 있다. 일부 구현에서, 시동 등의 동안 컴퓨터 시스템(1100) 내의 요소들 사이에서 정보를 전송하는 것을 돕는 기본 루틴들을 포함하는 기본 입력/출력 시스템(BIOS)은 전형적으로 ROM에 저장될 수 있다. 제한이 아닌 예로서, 시스템 메모리(1110)는 또한, 클라이언트 애플리케이션들, 웹 브라우저들, 중간-티어 애플리케이션들, 관계형 데이터베이스 관리 시스템(RDBMS) 등을 포함할 수 있는 애플리케이션 프로그램(1112), 프로그램 데이터(1114), 및 운영 체제(1116)를 나타낸다. 예로서, 운영 체제(1116)는, 다양한 버전의 Microsoft Windows®, Apple Macintosh®, 및/또는 Linux 운영 체제, 다양한 상업적으로 이용가능한 UNIX® 또는 UNIX류 운영 체제(다양한 GNU/Linux 운영 체제, Google Chrome® OS 등을 포함하지만 이것으로 제한되지 않음) 및/또는 iOS, Windows® Phone, Android® OS, BlackBerry® 11 OS, 및 Palm® OS 운영 체제들 등의 모바일 운영 체제들을 포함할 수 있다.

[0121] 스토리지 서브시스템(1118)은 또한, 일부 실시예의 기능을 제공하는 기본 프로그래밍 및 데이터 구성들을 저장하기 위한 유형의 컴퓨터 판독가능한 저장 매체를 제공할 수 있다. 프로세서에 의해 실행될 때 위에서 설명된 기능을 제공하는 소프트웨어(프로그램, 코드 모듈들, 명령어들)는 스토리지 서브시스템(1118)에 저장될 수 있다. 이들 소프트웨어 모듈들 또는 명령어들은 처리 유닛(1104)에 의해 실행될 수 있다. 스토리지 서브시스템(1118)은 또한, 본 개시내용에 따라 이용되는 데이터를 저장하기 위한 저장소를 제공할 수 있다.

[0122] 스토리지 서브시스템(1100)은 또한, 컴퓨터 판독가능한 저장 매체(1122)에 더 접속될 수 있는 컴퓨터 판독가능한 저장 매체 판독기(1120)를 포함할 수 있다. 함께, 및 선택사항으로서는 시스템 메모리(1110)와 조합하여, 컴퓨터 판독가능한 저장 매체(1122)는, 원격, 로컬, 고정 및/또는 착탈식 스토리지 디바이스들과 컴퓨터 판독가능한 정보를 일시적으로 및/또는 더 영구적으로 보관, 저장, 전송 및 회수하기 위한 저장 매체를 포괄적으로 나타낼 수 있다.

[0123] 코드 또는 코드의 일부분들을 포함하는 컴퓨터 판독가능한 저장 매체(1122)는 또한, 정보의 저장 및/또는 전송을 위한 임의의 방법 또는 기술로 구현된 휘발성 및 비휘발성, 착탈식 및 비착탈식 매체 등의 그러나 이것으로 제한되지 않는 저장 매체 및 통신 매체를 포함한, 본 기술분야에서 알려지거나 이용되는 임의의 적절한 매체를 포함할 수 있다. 이것은, RAM, ROM, EEPROM(electronically erasable programmable ROM), 플래시 메모리 또는 기타의 메모리 기술, CD-ROM, 디지털 다목적 디스크(DVD), 또는 다른 광학 스토리지, 자기 카세트들, 자기 테이프, 자기 디스크 스토리지 또는 기타의 자기 스토리지 디바이스들, 또는 기타의 유형의 컴퓨터 판독가능한 매체들의 유형의 컴퓨터 판독가능한 저장 매체를 포함할 수 있다.

[0124] 예로서, 컴퓨터 판독가능한 저장 매체(1122)는, 비착탈식 비휘발성 자기 매체로부터 판독하거나 여기에 기입하는 하드 디스크 드라이브, 착탈식 비휘발성 자기 디스크로부터 판독하거나 여기에 기입하는 자기 디스크 드라이브, CD ROM, DVD, Blu-Ray® 디스크 또는 기타의 광학 매체 등의 착탈식 비휘발성 광학 디스크로부터 판독하거나 여기에 기입하는 광학 디스크 드라이브를 포함할 수 있다. 컴퓨터 판독가능한 저장 매체(1122)는, Zip® 드라이브들, 플래시 메모리 카드들, 범용 직렬 버스(USB) 플래시 드라이브들, 보안 디지털(SD) 카드들, DVD 디스크들, 디지털 비디오 테이프 등을 포함할 수 있지만 이것으로 제한되는 것은 아니다. 컴퓨터 판독가능한 저장 매체(1122)는 또한, 플래시 메모리 기반의 SSD들, 엔터프라이즈 플래시 드라이브들, 솔리드 스테이트 ROM 등의 비휘발성 메모리 기반의 솔리드 스테이트 드라이브(SSD), 솔리드 스테이트 RAM, 동적 RAM, 정적 RAM, DRAM 기반의 SSD들, 자기 저항 RAM(MRAM) SSD들 등의 휘발성 메모리 기반의 SSD, 및 DRAM과 플래시 메모리 기반의 SSD들의 조합을 이용하는 하이브리드 SSD를 포함할 수 있다. 디스크 드라이브들 및 그들의 연관된 컴퓨터 판독가능한 매체는, 컴퓨터 판독가능한 명령어들, 데이터 구조들, 프로그램 모듈들, 및 컴퓨터 시스템(1100)을 위한 기타의 데이터의 비휘발성 스토리지를 제공할 수 있다.

- [0125] 통신 서브시스템(1124)은 다른 컴퓨터 시스템들 및 네트워크들에 대한 인터페이스를 제공한다. 통신 서브시스템(1124)은 컴퓨터 시스템(1100)으로부터 데이터를 수신하고 다른 시스템들에 데이터를 전송하기 위한 인터페이스로서 역할한다. 예를 들어, 통신 서브시스템(1124)은 컴퓨터 시스템(1100)이 인터넷을 통해 하나 이상의 디바이스에 접속할 수 있게 할 수 있다. 일부 실시예에서 통신 서브시스템(1124)은, (예컨대, 셀룰러 전화 기술, 3G, 4G 또는 EDGE(enhanced data rates for global evolution), WiFi(IEEE 802.11 패밀리 표준들) 또는 기타의 모바일 통신 기술, 또는 이들의 임의의 조합 등의, 진보된 데이터 네트워크 기술을 이용하여) 무선 음성 및/또는 데이터 네트워크에 액세스하기 위한 무선 주파수(RF) 트랜시버 컴포넌트들, GPS(global positioning system) 수신기 컴포넌트들, 및/또는 기타의 컴포넌트들을 포함할 수 있다. 일부 실시예에서 통신 서브시스템(1124)은 무선 인터페이스에 추가하여 또는 그 대신에 유선 네트워크 접속(예컨대, Ethernet)을 제공할 수 있다.
- [0126] 일부 실시예에서, 통신 서브시스템(1124)은 또한, 컴퓨터 시스템(1100)을 이용할 수 있는 한 명 이상의 사용자를 대신하여 구조화된 및/또는 비구조화된 데이터 피드들(1126), 이벤트 스트림들(1128), 이벤트 업데이트들(1130) 등의 형태로 입력 통신을 수신할 수 있다.
- [0127] 예로서, 통신 서브시스템(1124)은, Twitter® 피드들, Facebook® 업데이트들, RSS(Rich Site Summary) 피드들 등의 웹 피드들, 및/또는 하나 이상의 제3자 정보 소스로부터의 실시간 업데이트들 등의 소셜 네트워크 및/또는 다른 통신 서비스들의 사용자들로부터 데이터 피드들(1126)을 실시간으로 수신하도록 구성될 수 있다.
- [0128] 추가로, 통신 서브시스템(1124)은 또한, 명시적 종료 없이 본질적으로 연속적이거나 무한할 수 있는 실시간 이벤트들 및/또는 이벤트 업데이트들(1130)의 이벤트 스트림들(1128)을 포함할 수 있는 연속 데이터 스트림들의 형태로 데이터를 수신하도록 구성될 수 있다. 연속 데이터를 생성하는 애플리케이션들의 예들로는, 예를 들어 센서 데이터 애플리케이션들, 금융 시세 표시기들, 네트워크 성능 측정 도구들(예컨대, 네트워크 모니터링 및 트래픽 관리 애플리케이션들), 클릭스트림 분석 도구들, 자동차 트래픽 모니터링 등이 포함될 수 있다.
- [0129] 통신 서브시스템(1124)은 또한, 구조화된 및/또는 비구조화된 데이터 피드들(1126), 이벤트 스트림들(1128), 이벤트 업데이트들(1130) 등을 컴퓨터 시스템(1100)에 결합된 하나 이상의 스트리밍 데이터 소스 컴퓨터와 통신할 수 있는 하나 이상의 데이터베이스로 출력하도록 구성될 수 있다.
- [0130] 컴퓨터 시스템(1100)은, 핸드헬드 휴대형 디바이스(예컨대, iPhone® 셀룰러 전화, iPad® 컴퓨팅 태블릿, PDA), 착용형 디바이스(예컨대, Google Glass® 헤드 장착형 디스플레이), PC, 워크스테이션, 메인프레임, 키오스크, 서버 랙, 또는 기타 임의의 데이터 처리 시스템을 포함함 다양한 유형들 중 하나일 수 있다.
- [0131] 끊임없이 변화하는 컴퓨터들 및 네트워크들의 속성으로 인해, 도면에 묘사된 컴퓨터 시스템(1100)의 설명은 특정한 예로서만 의도된 것이다. 도면에 묘사된 시스템보다 더 많거나 더 적은 수의 컴포넌트를 포함하는 많은 다른 구성이 가능하다. 예를 들어, 맞춤형 하드웨어가 역시 이용될 수도 있고 그리고/또는 특정한 요소들이 하드웨어, 펌웨어, 소프트웨어(애플릿 포함), 또는 조합으로 구현될 수도 있다. 또한, 네트워크 입력/출력 디바이스들 등의 다른 컴퓨팅 디바이스들에 대한 접속이 채용될 수 있다. 본 명세서에 제공된 개시내용 및 교시들에 기초하여, 본 기술분야의 통상의 기술자라면 다양한 실시예를 구현하기 위한 다른 방식들 및/또는 방법들을 이해할 것이다.
- [0132] 특정한 실시예들이 설명되었지만, 다양한 수정, 변경, 대안적 구성, 및 균등물도 역시 본 개시내용의 범위 내에 포함된다. 실시예들은 소정의 특정한 데이터 처리 환경들 내에서의 동작으로 제한되지 않고, 복수의 데이터 처리 환경 내에서 자유롭게 동작한다. 추가적으로, 실시예들이 특정한 일련의 트랜잭션 및 단계들을 이용하여 설명되었지만, 본 개시내용의 범위가 설명된 일련의 트랜잭션 및 단계들로 제한되지 않는다는 것이 본 기술분야의 통상의 기술자에게 명백해야 한다. 전술된 실시예들의 다양한 피쳐 및 양태들은 개별적으로 또는 공동으로 이용될 수 있다.
- [0133] 또한, 실시예들이 하드웨어와 소프트웨어의 특정한 조합을 이용하여 설명되었지만, 하드웨어와 소프트웨어의 다른 조합들도 본 개시내용의 범위 내에 있다는 것을 인식해야 한다. 실시예들은, 하드웨어만으로, 또는 소프트웨어만으로, 또는 이들의 조합을 이용하여 구현될 수 있다. 여기서 설명된 다양한 프로세스는 동일한 프로세서 또는 상이한 프로세서들에서 임의의 조합으로 구현될 수 있다. 따라서, 컴포넌트들 또는 모듈들이 소정의 동작들을 수행하도록 구성된 것으로 설명되는 경우, 이러한 구성은, 예를 들어 그 동작을 수행하는 전자 회로들을 설계하거나, 그 동작을 수행하는 프로그래밍가능한 전자 회로들(마이크로프로세서들 등)을 프로그래밍하거나, 또는 이들의 임의의 조합에 의해 달성될 수 있다. 프로세스들은, 프로세스간 통신을 위한 종래 기술을 포함한다.

그러나 이것으로 제한되지 않는 다양한 기술을 이용하여 통신할 수 있고, 상이한 프로세스 쌍들은 상이한 기술들을 이용할 수 있거나, 동일한 프로세스 쌍이 상이한 시간들에서 상이한 기술들을 이용할 수도 있다.

[0134] 따라서, 본 명세서 및 도면은 제한적 의미라기보다는 예시적인 의미로 간주되어야 한다. 그러나, 청구항들에 기재된 더 넓은 사상 및 범위를 벗어나지 않고, 추가들, 차감들, 삭제들, 및 기타의 수정들 및 변경들이 이루어질 수 있음이 명백할 것이다. 따라서, 특정한 개시 실시예들이 설명되었지만, 이들은 제한하려는 의도가 아니다. 다양한 수정 및 균등물이 이하의 청구항들의 범위 내에 있다.

[0135] 본 개시된 실시예들을 설명하는 정황에서(특히, 이하의 청구항들의 정황에서) 사용되는 용어 "한(an)" 및 "하나(an)" 및 "그 하나(the)" 및 유사한 지시대상물은, 본 명세서에서, 달리 표시하거나 문맥상 명확하게 상충되지 않는 한, 단수 및 복수 양쪽 모두를 포괄하는 것으로 해석되어야 한다. 용어 "포함하는", "갖는", "내포하는", 및 "담고 있는"은, 달리 언급되지 않는 한 제약을 두지 않는 용어(즉, "포함한 그러나 이것으로 제한되지 않는"을 의미함)로서 해석되어야 한다. 용어 "접속된"은, 중간에 무언가가 있더라도 부분적으로 또는 전체적으로 ~ 내에 포함되거나, ~에 부착되거나, ~함께 결합된 것으로서 해석되어야 한다. 본 명세서에서 값들의 범위들을 열거한 것은, 본 명세서에서 달리 표시되지 않는 한, 그 범위 내에 속하는 각각의 별개의 값을 개별적으로 언급하는 약식 방법으로서 역할하며, 각각의 별개의 값은 본 명세서에서 개별적으로 나열된 것처럼 본 명세서에 통합된다. 본 명세서에서 설명된 모든 방법은 본 명세서에서 달리 표시되지 않는 한 또는 문맥상 명확하게 상충되지 않는 한 임의의 적절한 순서로 수행될 수 있다. 본 개시내용에서 제공된 임의의 예 및 모든 예들, 또는 예시적인 용어(예컨대, "~ 등의")의 사용은, 단지 실시예들을 더 명료하게 하기 위한 것일 뿐이며, 달리 청구되지 않는 한 본 개시내용의 범위에 어떠한 제한을 두는 것은 아니다. 본 명세서의 어떠한 용어도 본 개시내용의 실시예에 필수적인 임의의 청구되지 않은 요소를 가리키는 것으로 해석되어서는 안 된다.

[0136] 문구 "X, Y, 또는 Z" 중 적어도 하나" 등의 논리합 용어는, 구체적으로 달리 언급되지 않는 한, 또는 항목, 항 등을 제시하기 위해 일반적으로 사용될 때 문맥 내에서 달리 이해되지 않는 한, X, Y, 또는 Z이거나, 이들의 임의의 조합(예컨대, X, Y, 및/또는 Z)일 수 있다. 따라서, 이러한 논리합 언어는, 일반적으로, 소정 실시예가 X 중 적어도 하나, Y 중 적어도 하나, 또는 Z 중 적어도 하나가 각각 존재할 것을 요구한다는 것을 암시하고자 함이 아니며 암시해서도 안 된다.

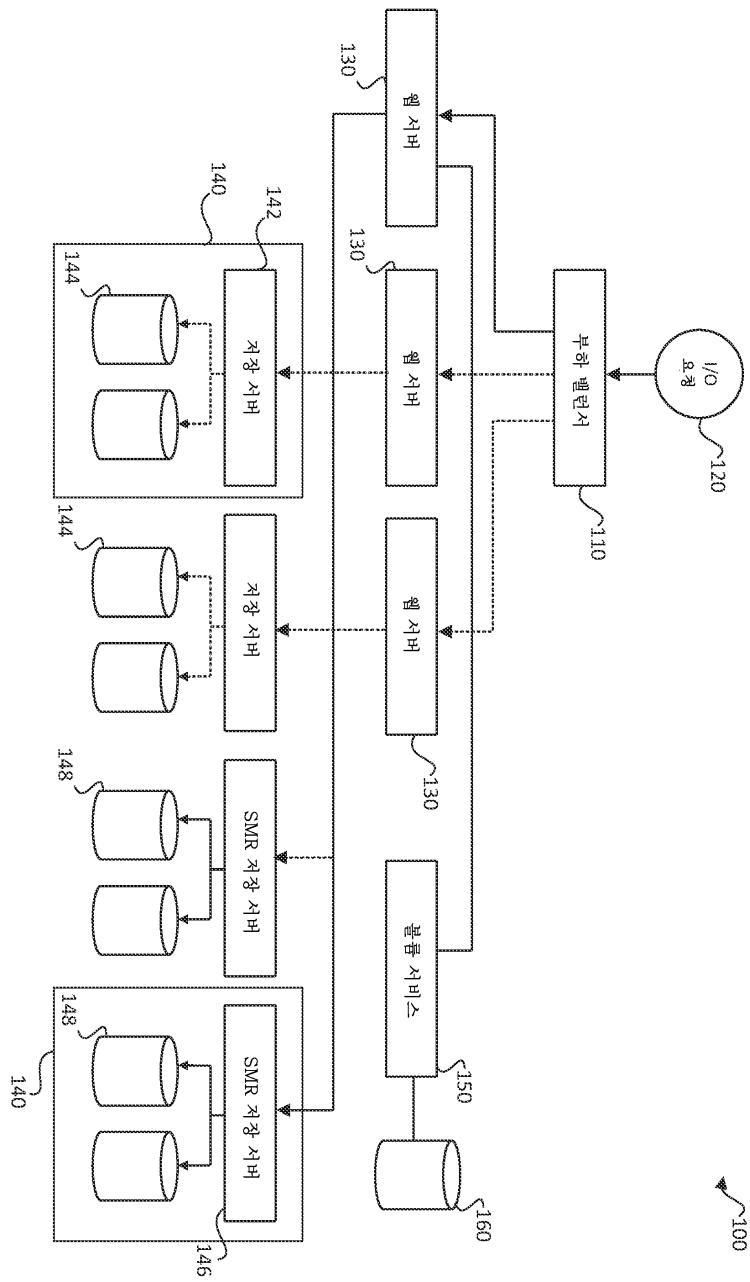
[0137] 본 개시내용을 실행하기 위한 알려진 최상의 모드를 포함한, 본 개시내용의 바람직한 실시예들이 여기서 설명되었다. 이들 바람직한 실시예들의 변형들은 전술된 설명을 읽을 때 본 기술분야의 통상의 기술자에게 명백해질 수 있다. 본 기술분야의 통상의 기술자라면, 적절하다면 이러한 변형들을 채택할 수 있고 본 개시내용은 여기서 구체적으로 설명된 것과는 다른 방식으로 실시될 수 있다는 것을 이해할 수 있을 것이다. 따라서, 본 개시내용은 적용가능한 법률이 허용하는 한 본 명세서에 첨부된 청구항들에 기재된 본 주제의 모든 수정 및 균등물을 포함한다. 더욱이, 본 명세서에서 달리 표시되지 않는 한, 모든 가능한 변형들에서의 전술된 요소들의 임의의 조합은 본 개시내용에 의해 포괄된다.

[0138] 본 명세서에서 인용된 간행물들, 특허 출원들, 및 특허들을 포함한 모든 참고 문헌은, 마치 각각의 참고 문헌이 개별적으로 그리고 구체적으로 참조에 의해 포함되고 그 전체내용이 여기에 기재된 것과 동일한 정도로 참조에 의해 본 명세서에 포함된다.

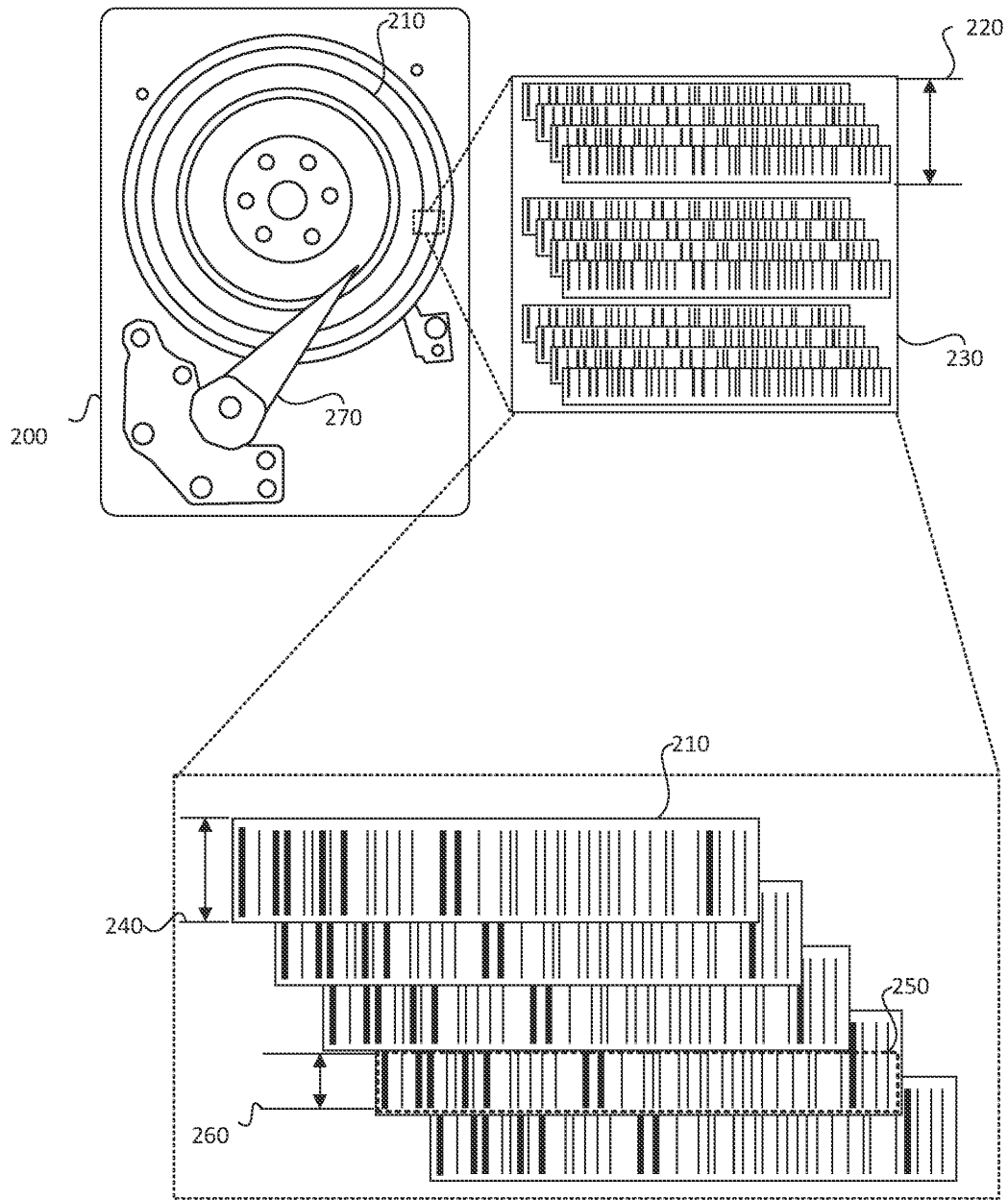
[0139] 전술된 명세서에서, 본 개시내용의 양태들이 그 특정한 실시예들을 참조하여 설명되었지만, 본 기술분야의 통상의 기술자라면 본 개시내용이 이것으로 제한되는 것은 아님을 인식할 것이다. 전술된 개시내용의 다양한 피처 및 양태들은 개별적으로 또는 공동으로 이용될 수 있다. 또한, 실시예들은 본 명세서의 더 넓은 사상 및 범위를 벗어나지 않고 본 명세서에 설명된 것들 이외의 임의의 수의 환경 및 응용에서 이용될 수 있다. 따라서, 명세서 및 도면들은 제한적인 것이 아니라 예시적인 것으로 간주되어야 한다.

도면

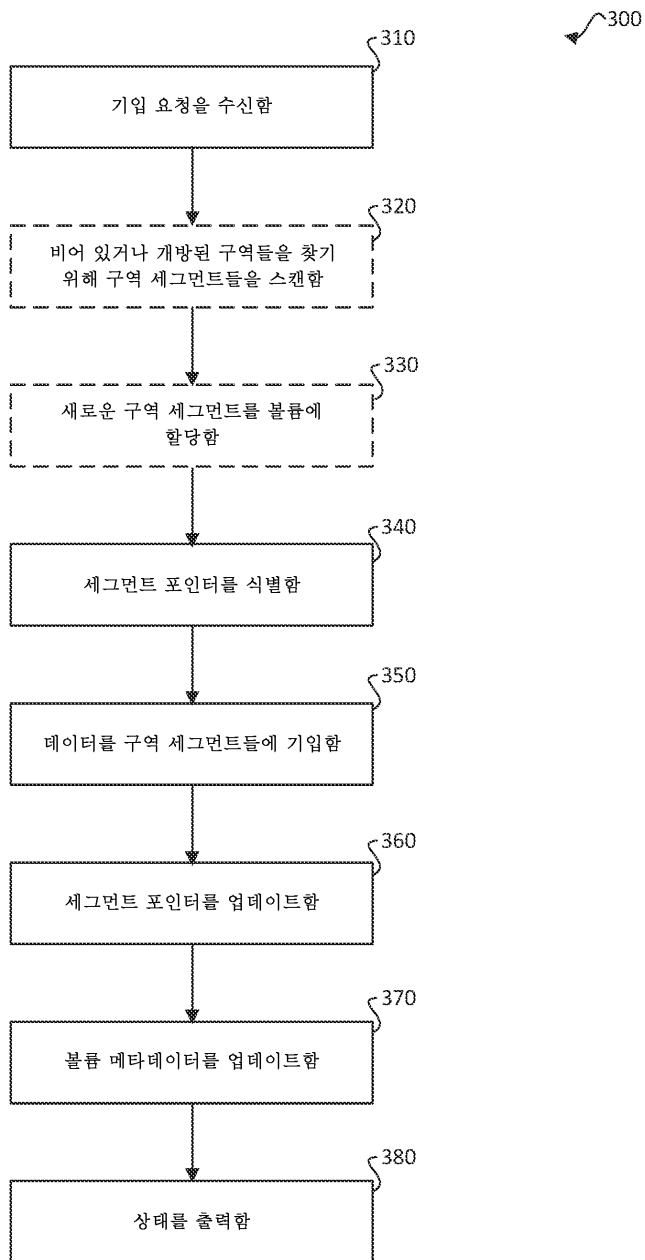
도면1



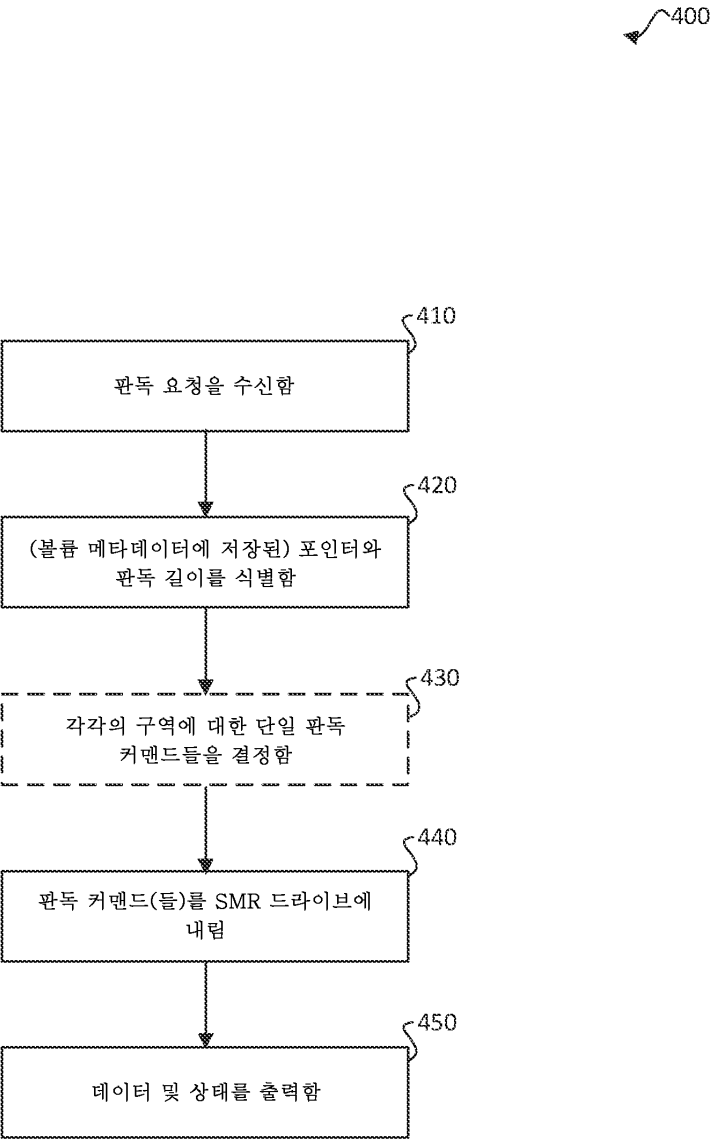
도면2



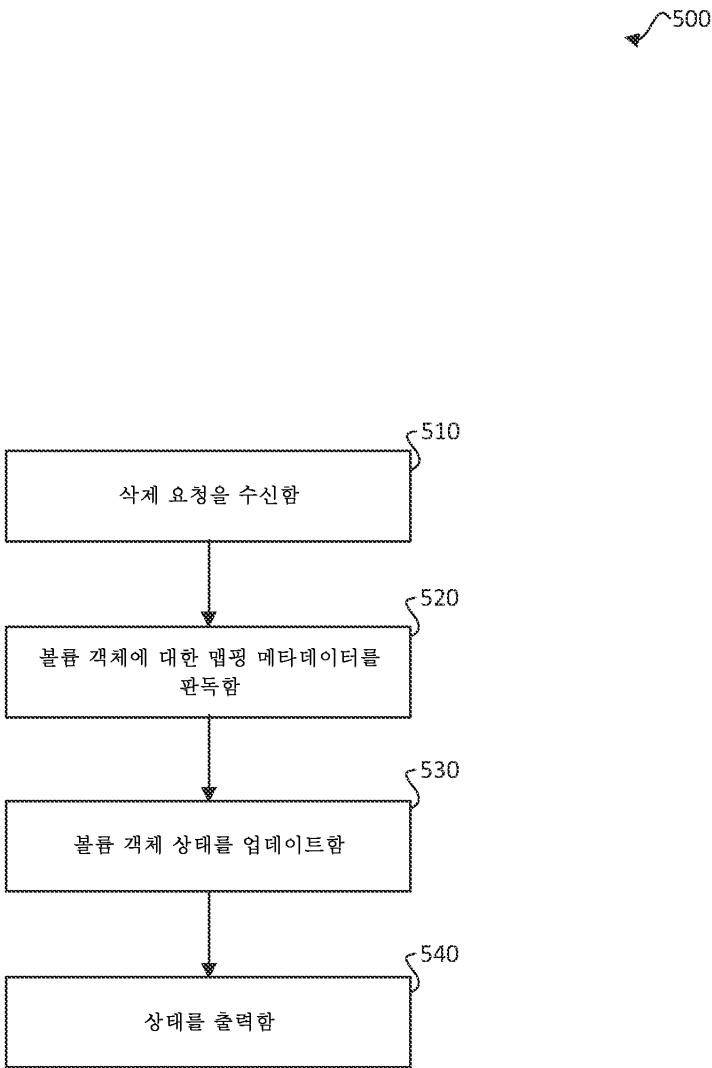
도면3



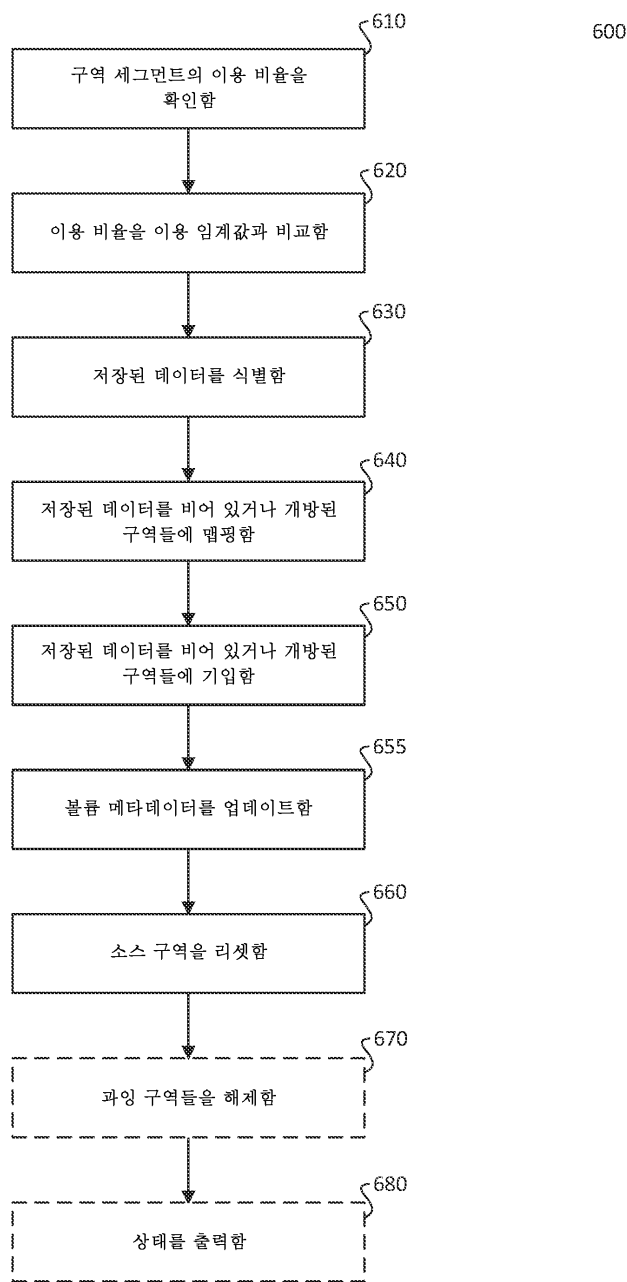
도면4



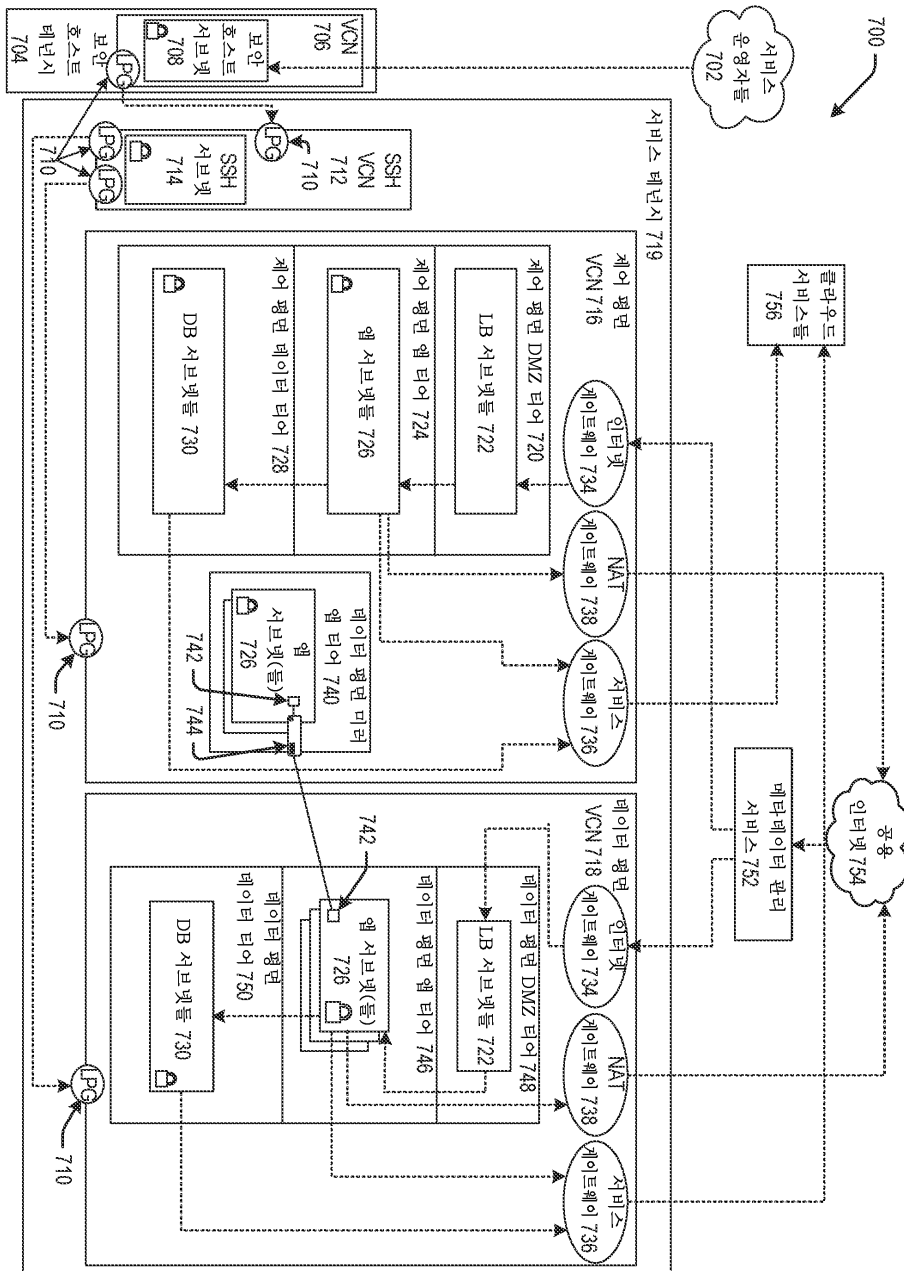
도면5



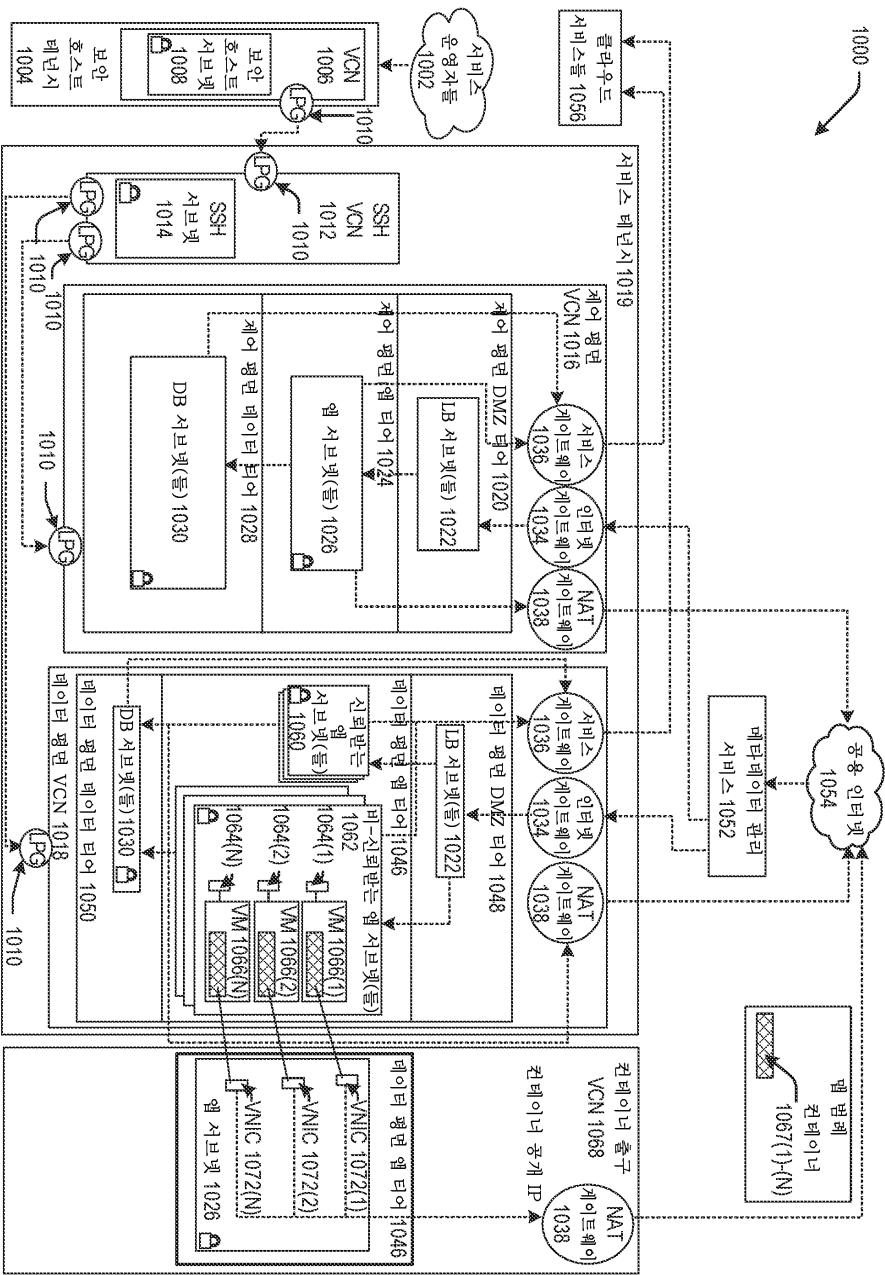
도면6



도면7



도면10



도면11

