(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2024년01월24일
(11) 등록번호 10-2628362
(24) 등록일자 2024년01월18일

(51) 국제특허분류(Int. Cl.)
G06F 9/48 (2018.01) *G06F 11/14* (2006.01)
G06F 16/28 (2019.01) *G06F 9/455* (2018.01)
G06F 9/50 (2018.01) *H04L 41/0897* (2022.01)
(52) CPC특허분류
G06F 9/4856 (2013.01)
G06F 11/1479 (2013.01)
(21) 출원번호 10-2022-0097293(분할)
(22) 출원일자 2022년08월04일
심사청구일자 2022년08월04일
(65) 공개번호 10-2022-0113663
(43) 공개일자 2022년08월16일
(62) 원출원 특허 10-2020-0113106
원출원일자 2020년09월04일
심사청구일자 2020년09월04일
(30) 우선권주장
62/899,794 2019년09월13일 미국(US)
16/579,945 2019년09월24일 미국(US)
(56) 선행기술조사문헌
George John ET AL: "Blue/Green Deployments on AWS", August 2016
US20060053216 A1

(73) 특허권자
구글 엘엘씨
미국 캘리포니아 마운틴 뷰 엠피시어터 파크웨이 1600 (우:94043)
(72) 발명자
스미스 다니엘 베리타스
미국 캘리포니아 마운틴 뷰 엠피시어터 파크웨이 1600 (우:94043)
(74) 대리인
박장원

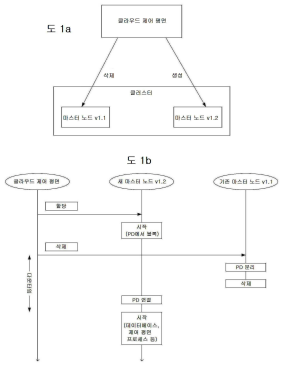
전체 청구항 수 : 총 20 항 심사관 : 김종기

(54) 발명의 명칭 **컨테이너화된 환경에서 클러스터의 라이브 마이그레이션**

(57) 요 약

이 기술은 제1 클러스터로부터 제2 클러스터로의 라이브 마이그레이션을 제공한다. 예를 들어, 하나 이상의 클러스터 제어 평면에 대한 요청이 수신되면, 수신된 요청의 미리 결정된 부분이 제2 클러스터의 제어 평면에 할당될 수 있으며, 수신된 요청의 나머지 부분은 제1 클러스터의 제어 평면에 할당될 수 있다. 요청의 미리 결정된 부분은 제2 클러스터의 제어 평면을 사용하여 처리된다. 요청의 미리 결정된 부분을 처리하는 동안, 제2 클러스터에 실패가 있는지 여부를 검출한다. 상기 제2 클러스터에서 실패를 검출하지 못한 것에 기초하여, 모든 요청이 제2 클러스터의 제어 평면에 할당될 때까지 미리 결정된 단계에서 상기 제2 클러스터의 제어 평면에 할당된 상기 요청의 미리 결정된 부분을 증가시킬 수 있다.

대 표 도



(52) CPC특허분류
G06F 16/285 (2019.01)
G06F 9/45533 (2013.01)
G06F 9/5077 (2013.01)
G06F 9/5088 (2013.01)
H04L 41/0897 (2022.05)
G06F 2009/4557 (2019.08)

명 세 서

청구범위

청구항 1

제1 클러스터에서 제2 클러스터로 마이그레이션하는 방법으로서,

하나 이상의 프로세서에 의해, 하나 이상의 클러스터 제어 평면에 대한 요청을 수신하는 단계, 상기 하나 이상의 클러스터 제어 평면은 상기 제1 클러스터의 제어 평면 및 상기 제2 클러스터의 제어 평면을 포함하며;

상기 하나 이상의 프로세서에 의해, 상기 수신된 요청의 미리 결정된 부분을 상기 제2 클러스터의 제어 평면에 할당하고, 상기 수신된 요청의 나머지 부분을 상기 제1 클러스터의 제어 평면에 할당하는 단계;

상기 하나 이상의 프로세서에 의해, 상기 제2 클러스터의 제어 평면을 사용하여 상기 요청의 미리 결정된 부분을 처리하는 단계, 상기 요청의 미리 결정된 부분을 처리하는 단계는:

상기 하나 이상의 프로세서에 의해, 상기 제2 클러스터에서 실행되는 포드가 상기 제1 클러스터의 스토리지를 참조한다고 결정하는 단계;

상기 하나 이상의 프로세서에 의해, 상기 제2 클러스터에 스토리지를 생성하는 단계, 상기 제1 클러스터의 스토리지와 상기 제2 클러스터의 스토리지는 상이한 위치에 위치되며;

상기 하나 이상의 프로세서에 의해, 스토리지 드라이버를 사용하여, 상기 제2 클러스터의 포드와 관련된 데이터에 상기 제1 클러스터의 스토리지를 판독하는 단계; 및

상기 하나 이상의 프로세서에 의해, 상기 제2 클러스터의 포드에 의해 변경되지 않은 데이터를 상기 제1 클러스터의 스토리지로부터 상기 제2 클러스터의 스토리지로 복사하는 단계를 포함하며,

상기 하나 이상의 프로세서에 의해, 상기 요청의 미리 결정된 부분을 처리하는 동안 상기 제2 클러스터에 실패가 있는지를 검출하는 단계; 및

상기 하나 이상의 프로세서에 의해, 상기 제2 클러스터에서 실패를 검출하지 못한 것에 기초하여, 수신된 모든 요청이 제2 클러스터의 제어 평면에 할당될 때까지 미리 결정된 단계에서 상기 제2 클러스터의 제어 평면에 할당된 상기 요청의 미리 결정된 부분을 증가시키는 단계를 포함하는, 방법.

청구항 2

청구항 1에 있어서, 상기 수신된 요청은 상기 제1 클러스터의 클러스터 브릿징 애그리게이터 및 상기 제2 클러스터의 클러스터 브릿징 애그리게이터에 의해 할당되고, 상기 제1 클러스터 및 상기 제2 클러스터는 동일한 클라우드에서 동작되는, 방법.

청구항 3

청구항 1에 있어서, 상기 수신된 요청은 상기 제1 클러스터에서 실행되는 워크로드로부터의 요청을 포함하고, 상기 워크로드로부터의 요청은 상기 제1 클러스터에 주입된 사이드카 컨테이너에 의해 인터셉트되고 상기 제2 클러스터의 클러스터 브릿징 애그리게이터로 라우팅되며, 상기 제1 클러스터와 상기 제2 클러스터는 서로 다른 클라우드에서 동작되는, 방법.

청구항 4

청구항 1에 있어서, 상기 수신된 요청의 할당은 복수의 미리 결정된 복수에서 수행되며, 상기 요청은 사용자-에이전트, 사용자 계정, 사용자 그룹, 오브젝트 유형, 리소스 유형, 상기 오브젝트의 위치 또는 상기 요청의 송신자의 위치 중 하나 이상에 기초하여 상기 제1 클러스터 또는 상기 제2 클러스터로 향하는, 방법.

청구항 5

청구항 1에 있어서,

상기 하나 이상의 프로세서에 의해, 상기 제2 클러스터의 제어 평면의 하나 이상의 데이터베이스를 상기 제1 클

러스터의 제어 평면의 하나 이상의 데이터베이스를 포함하는 쿼럼(quorum)에 조인(join)하는 단계를 더 포함하며, 상기 제1 클러스터 및 상기 제2 클러스터는 동일한 클라우드에서 실행되는, 방법.

청구항 6

청구항 1에 있어서,

상기 하나 이상의 프로세서에 의해, 상기 제2 클러스터의 제어 평면의 하나 이상의 데이터베이스를 상기 제1 클러스터의 제어 평면의 하나 이상의 데이터베이스와 동기화하는 단계를 더 포함하며, 상기 제1 클러스터 및 상기 제2 클러스터는 서로 다른 클라우드에서 동작되는, 방법.

청구항 7

청구항 1에 있어서,

상기 하나 이상의 프로세서에 의해, 오브젝트 잠금(object lock)의 미리 결정된 부분을 상기 제2 클러스터의 하나 이상의 제어기에 할당하고, 오브젝트 잠금의 나머지 부분을 상기 제1 클러스터의 하나 이상의 제어기에 할당하는 단계;

상기 하나 이상의 프로세서에 의해, 상기 제2 클러스터의 하나 이상의 제어기에 의해 잠긴 오브젝트를 작동시키는 단계;

상기 하나 이상의 프로세서에 의해, 상기 잠긴 오브젝트를 작동시키는 동안 상기 제2 클러스터에서 실패가 있었는지 여부를 검출하는 단계;

상기 하나 이상의 프로세서에 의해, 상기 제2 클러스터에서 실패를 검출하지 않는 것에 기초하여, 상기 제2 클러스터의 하나 이상의 제어기에 할당된 오브젝트 잠금의 미리 결정된 부분을 증가시키는 단계를 더 포함하는, 방법.

청구항 8

청구항 1에 있어서,

상기 하나 이상의 프로세서에 의해, 수신된 모든 요청이 상기 제2 클러스터의 제어 평면에 할당된다는 것을 결정하는 단계;

상기 하나 이상의 프로세서에 의해, 상기 결정에 기초하여, 상기 제1 클러스터의 제어 평면을 삭제하는 단계를 더 포함하며, 상기 제1 클러스터 및 상기 제2 클러스터는 동일한 클라우드에서 동작되는, 방법.

청구항 9

청구항 1에 있어서,

상기 하나 이상의 프로세서에 의해, 상기 제2 클러스터에서 하나 이상의 실패를 검출하는 것에 기초하여, 상기 수신된 요청을 상기 제2 클러스터의 제어 평면에 할당하는 것을 중지하는 단계를 더 포함하는, 방법.

청구항 10

청구항 1에 있어서,

상기 하나 이상의 프로세서에 의해, 상기 제2 클러스터에서 하나 이상의 실패를 검출하는 것에 기초하여, 상기 검출된 실패에 대한 정보를 포함하는 출력을 생성하는 단계를 더 포함하는, 방법.

청구항 11

청구항 1에 있어서,

상기 하나 이상의 프로세서에 의해, 상기 제2 클러스터에서 실패를 검출한 것에 기초하여, 수신된 모든 요청이 제1 클러스터의 제어 평면에 할당될 때까지 상기 제2 클러스터의 제어 평면에 할당된 상기 요청의 미리 결정된 부분을 감소시키는 단계를 더 포함하는, 방법.

청구항 12

청구항 1에 있어서,

상기 하나 이상의 프로세서에 의해, 수신된 모든 요청이 상기 제1 클러스터의 제어 평면에 할당된다는 것을 결정하는 단계;

상기 하나 이상의 프로세서에 의해, 상기 결정에 기초하여, 상기 제2 클러스터를 삭제하는 단계를 더 포함하는, 방법.

청구항 13

청구항 1에 있어서,

상기 하나 이상의 프로세서에 의해, 상기 제2 클러스터에서 추가 포드(pod)를 스케줄링하는 단계;

상기 하나 이상의 프로세서에 의해, 상기 제1 클러스터의 포드의 상태를 기록하는 단계;

상기 하나 이상의 프로세서에 의해, 상기 제1 클러스터의 상기 포드의 기록된 상태를 상기 제2 클러스터의 포드로 전송하는 단계를 더 포함하는, 방법.

청구항 14

청구항 13에 있어서,

상기 하나 이상의 프로세서에 의해, 상기 제1 클러스터의 상기 포드에 의한 워크로드의 실행을 일시 중지하는 단계;

상기 하나 이상의 프로세서에 의해, 상기 제1 클러스터의 상기 포드의 상태를 기록한 이후 상기 제1 클러스터의 상기 포드의 상태 변화를 복사하는 단계;

상기 하나 이상의 프로세서에 의해, 상기 복사된 상태 변화를 제2 클러스터의 추가 포드로 전송하는 단계;

상기 하나 이상의 프로세서에 의해, 상기 제2 클러스터의 상기 추가 포드에 의한 워크로드의 실행을 재개하는 단계;

상기 하나 이상의 프로세서에 의해, 상기 제1 클러스터의 포드로 향하는 트래픽을 상기 제2 클러스터의 추가 포드로 포워딩하는 단계;

상기 하나 이상의 프로세서에 의해, 상기 제1 클러스터의 포드를 삭제하는 단계를 더 포함하는, 방법.

청구항 15

청구항 1에 있어서,

상기 하나 이상의 프로세서에 의해, 상기 제1 클러스터의 제1 작업자 노드가 상기 제2 클러스터로 이동될 하나 이상의 포드를 갖는다고 결정하는 단계;

상기 하나 이상의 프로세서에 의해, 상기 제1 클러스터의 제1 작업자 노드가 새로운 포드를 추가하는 것을 방지하는 단계;

상기 하나 이상의 프로세서에 의해, 상기 제1 작업자 노드의 하나 이상의 포드 중 일부를 상기 제2 클러스터의 하나 이상의 기존 작업자 노드로 이동시키는 단계;

상기 하나 이상의 프로세서에 의해, 상기 제2 클러스터의 기존 작업자 노드에 더 이상 용량이 없다고 결정하는 단계;

상기 하나 이상의 프로세서에 의해, 상기 제2 클러스터에 하나 이상의 추가 작업자 노드를 생성하는 단계;

상기 하나 이상의 프로세서에 의해, 상기 제1 작업자 노드의 나머지 하나 이상의 포드를 상기 제2 클러스터의 추가 작업자 노드로 이동시키는 단계;

상기 하나 이상의 프로세서에 의해, 상기 제1 클러스터의 제1 작업자 노드가 상기 제2 클러스터로 이동될 포드를 더 이상 갖지 않는다고 결정하는 단계;

상기 하나 이상의 프로세서에 의해, 상기 제1 클러스터의 제1 작업자 노드를 삭제하는 단계를 더 포함하는, 방

법.

청구항 16

청구항 13에 있어서,

상기 하나 이상의 프로세서에 의해, 하나 이상의 워크로드에 대한 요청을 수신하는 단계, 상기 하나 이상의 워크로드는 상기 제1 클러스터에서 실행되는 워크로드 및 상기 제2 클러스터에서 실행되는 워크로드를 포함하며;

상기 하나 이상의 프로세서에 의해, 적어도 하나의 글로벌 로드 밸런서를 사용하여, 상기 수신된 요청을 상기 제1 클러스터에서 실행되는 워크로드와 상기 제2 클러스터에서 실행되는 워크로드 사이의 하나 이상의 워크로드에 할당하는 단계를 더 포함하는, 방법.

청구항 17

청구항 1에 있어서,

상기 하나 이상의 프로세서에 의해, 스토리지 드라이버를 사용하여, 상기 제2 클러스터 내의 포드와 관련된 데이터에 대한 상기 제2 클러스터의 스토리지를 판독하는 단계를 더 포함하는, 방법.

청구항 18

청구항 17에 있어서,

상기 하나 이상의 프로세서에 의해, 상기 제2 클러스터의 포드에 의해 이루어진 변경을 상기 제2 클러스터의 스토리지에 기록하는 단계를 더 포함하는, 방법.

청구항 19

제1 클러스터에서 제2 클러스터로 마이그레이션하는 시스템으로서,

하나 이상의 프로세서를 포함하며, 상기 하나 이상의 프로세서는:

하나 이상의 클러스터 제어 평면에 대한 요청을 수신하고, 상기 하나 이상의 클러스터 제어 평면은 상기 제1 클러스터의 제어 평면 및 상기 제2 클러스터의 제어 평면을 포함하며;

상기 수신된 요청의 미리 결정된 부분을 상기 제2 클러스터의 제어 평면에 할당하고, 상기 요청의 나머지 부분을 상기 제1 클러스터의 제어 평면에 할당하고;

상기 제2 클러스터의 제어 평면을 사용하여 상기 요청의 미리 결정된 부분을 처리하고, 상기 요청의 미리 결정된 부분을 처리하는 것은:

상기 제2 클러스터에서 실행되는 포드가 상기 제1 클러스터의 스토리지를 참조한다고 결정하는 것;

상기 제2 클러스터에 스토리지를 생성하는 것, 상기 제1 클러스터의 스토리지와 상기 제2 클러스터의 스토리지는 상이한 위치에 위치되며;

상기 제2 클러스터의 포드와 관련된 데이터에 상기 제1 클러스터의 스토리지를 판독하는 것; 및

상기 제2 클러스터의 포드에 의해 변경되지 않은 데이터를 상기 제1 클러스터의 스토리지로부터 상기 제2 클러스터의 스토리지로 복사하는 것을 포함하며,

상기 요청의 미리 결정된 부분을 처리하는 동안 상기 제2 클러스터에 실패가 있는지를 검출하고; 그리고

상기 제2 클러스터에서 실패를 검출하지 못한 것에 기초하여, 수신된 모든 요청이 제2 클러스터의 제어 평면에 할당될 때까지 미리 결정된 단계에서 상기 제2 클러스터의 제어 평면에 할당된 상기 요청의 미리 결정된 부분을 증가시키도록 구성되는, 시스템.

청구항 20

청구항 19에 있어서, 상기 제1 클러스터 및 상기 제2 클러스터는 상이한 소프트웨어 버전을 동작, 상이한 위치에서 동작, 상이한 클라우드 제공자에 의해 제공되는 상이한 클라우드에서 동작, 적어도 하나의 클라우드가 사용자의 온-프레미스 데이터센터이거나 상이한 네트워크에 연결된 상이한 클라우드에서 동작하는 것 중 적어도

하나인, 시스템.

발명의 설명

기 술 분 야

[0001] 관련 출원에 대한 상호 참조

[0002] 본 출원은 2019년 9월 13일 자로 출원된 미국 가출원 제62/899,794호에 대해 우선권을 주장하며, 그 전체 내용은 본원에 참조로서 통합된다.

배 경 기 술

[0003] 컨테이너화된 환경은 분산 또는 클라우드 컴퓨팅 시스템에서 애플리케이션을 효율적으로 실행하는데 사용될 수 있다. 예를 들어, 애플리케이션의 다양한 서비스가 컨테이너로 패키징될 수 있다. 컨테이너는 논리적으로 포드로 그룹화되어, 가상 머신("VM")인 노드의 클러스터와 같은 클라우드 컴퓨팅 시스템에 배포될 수 있다. 클러스터는 컨테이너를 실행하는 하나 이상의 작업자 노드와 다양한 클라우드 및 사용자 정의 구성 및 정책에 따라 작업자 노드의 워크로드 및 리소스를 관리하는 하나 이상의 마스터 노드를 포함할 수 있다. 클러스터 제어 평면은 클러스터의 마스터 노드에서 실행되는 논리 서비스로, 다수의 소프트웨어 프로세스와 클러스터의 현재 상태를 저장하는 데이터베이스를 포함할 수 있다. 가용성을 높이기 위해, 클러스터의 마스터 노드를 복제될 수 있고, 이 경우 마스터 노드 복제본의 쿼럼(quorum)은 클러스터의 임의의 상태를 수정하기 위해 클러스터에 대해 동의해야 한다. 클러스터는 클라우드 제공자에 의해 동작되거나 최종 사용자가 자체 관리할 수 있다. 예를 들어, 클라우드 제공자는 클라우드 상의 모든 클러스터에 대한 규칙 및 정책을 설정하거나 사용자가 클러스터에서 관리 작업을 쉽게 수행할 수 있는 방법을 제공하는 클라우드 제어 평면을 가질 수 있다.

[0004] 클라우드 제공자 또는 최종 사용자가 클러스터 환경을 변경하면, 변경 사항이 클러스터에 위험을 초래할 수 있다. 환경 변경의 예는 노드, 클러스터 제어 평면 또는 클라우드 제어 평면에 대한 업그레이드일 수 있는 소프트웨어 업그레이드를 포함할 수 있다. 또 다른 환경 변경의 예는 서로 다른 물리적 위치에 있는 데이터센터 간과 같은 위치들 간 또는 동일한 데이터센터 내의 지역 또는 영역과 같은 서로 다른 논리적 위치들 간의 클러스터의 리소스 이동을 포함할 수 있다. 또한 사용자는 사용자가 클라우드 제공자로 동작하는 자체 관리 클러스터로부터 클라우드 제공자가 관리하는 클러스터로 또는 일반적으로 상이한 클라우드 제공자에 의해 관리되는 두 클러스터들 사이에서 마이그레이션하기 원할 수 있다. 이러한 마이그레이션은 클러스터의 제어 평면을 새로운 클라우드 제공자의 제어로 전환하는 것을 포함하므로 위험이 따른다. 또 다른 예로서, 사용자는 클러스터를 중지하지 않고 클러스터에 대한 클라우드를 변경하기를 원할 수 있으며, 이는 현재 클러스터에서 실행 중인 프로세스에 위험할 수 있다.

[0005] 도 1a 및 1b는 클러스터의 환경, 특히 클러스터 제어 평면에 대한 소프트웨어 업그레이드를 변경하기 위한 현재 프로세스를 도시한다. 예를 들어, 클라우드 제어 평면은 클라우드 제공자가 호스팅하는 VM에 대한 새 버전의 구성 및 정책과 같은 소프트웨어 업그레이드를 도입할 수 있다. 도 1a에 도시된 바와 같이, 클러스터를 이전 버전 "v1.1"에서 새 버전 "v1.2"로 스위칭하기 위해, 클라우드 제어 평면은 클러스터에서 이전 마스터 노드를 삭제하고 그 자리에 새 마스터 노드를 만든다. 그림 1b에 도시된 바와 같이, 이 교체 프로세스 동안, 새 마스터 노드는 기존 마스터 노드가 PD(persistent disk)로부터 분리되고 이전 마스터 노드가 삭제될 때까지 영구 디스크("PD")에 연결되지 않도록 차단될 수 있다.

발명의 내용

[0006] 본 개시는 제1 클러스터에서 제2 클러스터로 마이그레이션하는 것을 제공하며, 하나 이상의 프로세서에 의해, 하나 이상의 클러스터 제어 평면에 대한 요청을 수신하는 단계, 상기 하나 이상의 클러스터 제어 평면은 상기 제1 클러스터의 제어 평면 및 상기 제2 클러스터의 제어 평면을 포함하며; 상기 하나 이상의 프로세서에 의해, 상기 수신된 요청의 미리 결정된 부분을 상기 제2 클러스터의 제어 평면에 할당하고, 상기 수신된 요청의 나머지 부분을 상기 제1 클러스터의 제어 평면에 할당하는 단계; 상기 하나 이상의 프로세서에 의해, 상기 제2 클러스터의 제어 평면을 사용하여 상기 요청의 미리 결정된 부분을 처리하는 단계; 상기 하나 이상의 프로세서에 의해, 상기 요청의 미리 결정된 부분을 처리하는 동안 상기 제2 클러스터에 실패가 있는지를 검출하는 단계; 및 상기 하나 이상의 프로세서에 의해, 상기 제2 클러스터에서 실패를 검출하지 못한 것에 기초하여, 수신된 모든 요청이 제2 클러스터의 제어 평면에 할당될 때까지 미리 결정된 단계에서 상기 제2 클러스터의 제어 평면에 할

당된 상기 요청의 미리 결정된 부분을 증가시키는 단계를 포함한다.

[0007] 상기 수신된 요청은 상기 제1 클러스터의 클러스터 브릿징 애그리게이터 및 상기 제2 클러스터의 클러스터 브릿징 애그리게이터에 의해 할당되고, 상기 제1 클러스터 및 상기 제2 클러스터는 동일한 클라우드에서 동작된다. 상기 수신된 요청은 상기 제1 클러스터에서 실행되는 워크로드로부터의 요청을 포함할 수 있고, 상기 워크로드로부터의 요청은 상기 제1 클러스터에 주입된 사이드카 컨테이너에 의해 인터셉트될 수 있고 상기 제2 클러스터의 클러스터 브릿징 애그리게이터로 라우팅되며, 상기 제1 클러스터와 상기 제2 클러스터는 서로 다른 클라우드에서 동작된다.

[0008] 상기 수신된 요청의 할당은 복수의 미리 결정된 복수에서 수행될 수 있고, 상기 요청은 사용자-에이전트, 사용자 계정, 사용자 그룹, 오브젝트 유형, 리소스 유형, 상기 오브젝트의 위치 또는 상기 요청의 송신자의 위치 중 하나 이상에 기초하여 상기 제1 클러스터 또는 상기 제2 클러스터로 향한다.

[0009] 방법은 상기 하나 이상의 프로세서에 의해, 상기 제2 클러스터의 제어 평면의 하나 이상의 데이터베이스를 상기 제1 클러스터의 제어 평면의 하나 이상의 데이터베이스를 포함하는 쿼럼(quorum)에 조인(join)하는 단계를 더 포함하며, 상기 제1 클러스터 및 상기 제2 클러스터는 동일한 클라우드에서 실행된다. 방법은 상기 하나 이상의 프로세서에 의해, 상기 제2 클러스터의 제어 평면의 하나 이상의 데이터베이스를 상기 제1 클러스터의 제어 평면의 하나 이상의 데이터베이스와 동기화하는 단계를 더 포함하며, 상기 제1 클러스터 및 상기 제2 클러스터는 서로 다른 클라우드에서 동작된다.

[0010] 상기 방법은 상기 하나 이상의 프로세서에 의해, 오브젝트 잠금(object lock)의 미리 결정된 부분을 상기 제2 클러스터의 하나 이상의 제어기에 할당하고, 오브젝트 잠금의 나머지 부분을 상기 제1 클러스터의 하나 이상의 제어기에 할당하는 단계; 상기 하나 이상의 프로세서에 의해, 상기 제2 클러스터의 하나 이상의 제어기에 의해 잠긴 오브젝트를 작동시키는 단계; 상기 하나 이상의 프로세서에 의해, 상기 잠긴 오브젝트를 작동시키는 동안 상기 제2 클러스터에서 실패가 있었는지 여부를 검출하는 단계; 상기 하나 이상의 프로세서에 의해, 상기 제2 클러스터에서 실패를 검출하지 않는 것에 기초하여, 상기 제2 클러스터의 하나 이상의 제어기에 할당된 오브젝트 잠금의 미리 결정된 부분을 증가시키는 단계를 더 포함할 수 있다.

[0011] 상기 방법은 상기 하나 이상의 프로세서에 의해, 수신된 모든 요청이 상기 제2 클러스터의 제어 평면에 할당된다는 것을 결정하는 단계; 상기 하나 이상의 프로세서에 의해, 상기 결정에 기초하여, 상기 제1 클러스터의 제어 평면을 삭제하는 단계를 더 포함하며, 상기 제1 클러스터 및 상기 제2 클러스터는 동일한 클라우드에서 동작된다. 상기 방법은 상기 하나 이상의 프로세서에 의해, 상기 제2 클러스터에서 하나 이상의 실패를 검출하는 것에 기초하여, 상기 수신된 요청을 상기 제2 클러스터의 제어 평면에 할당하는 것을 중지하는 단계를 더 포함할 수 있다. 상기 방법은 상기 하나 이상의 프로세서에 의해, 상기 제2 클러스터에서 하나 이상의 실패를 검출하는 것에 기초하여, 상기 검출된 실패에 대한 정보를 포함하는 출력을 생성하는 단계를 더 포함할 수 있다. 상기 방법은 상기 하나 이상의 프로세서에 의해, 상기 제2 클러스터에서 실패를 검출한 것에 기초하여, 수신된 모든 요청이 제1 클러스터의 제어 평면에 할당될 때까지 상기 제2 클러스터의 제어 평면에 할당된 상기 요청의 미리 결정된 부분을 감소시키는 단계를 더 포함할 수 있다. 상기 방법은 상기 하나 이상의 프로세서에 의해, 수신된 모든 요청이 상기 제1 클러스터의 제어 평면에 할당된다는 것을 결정하는 단계; 상기 하나 이상의 프로세서에 의해, 상기 결정에 기초하여, 상기 제2 클러스터를 삭제하는 단계를 더 포함할 수 있다.

[0012] 상기 방법은 상기 하나 이상의 프로세서에 의해, 상기 제2 클러스터의 포드(pod)를 스케줄링하는 단계; 상기 하나 이상의 프로세서에 의해, 상기 제1 클러스터의 포드의 상태를 기록하는 단계; 상기 하나 이상의 프로세서에 의해, 상기 제1 클러스터의 상기 포드의 기록된 상태를 상기 제2 클러스터의 포드로 전송하는 단계를 더 포함할 수 있다. 상기 방법은 상기 하나 이상의 프로세서에 의해, 상기 제1 클러스터의 상기 포드에 의한 워크로드의 실행을 일시 중지하는 단계; 상기 하나 이상의 프로세서에 의해, 상기 제1 클러스터의 상기 포드의 상태를 기록한 이후 상기 제1 클러스터의 상기 포드의 상태 변화를 복사하는 단계; 상기 하나 이상의 프로세서에 의해, 상기 복사된 상태 변화를 제2 클러스터의 포드로 전송하는 단계; 상기 하나 이상의 프로세서에 의해, 상기 제2 클러스터의 상기 포드에 의한 워크로드의 실행을 재개하는 단계; 상기 하나 이상의 프로세서에 의해, 상기 제1 클러스터의 포드로 향하는 트래픽을 상기 제2 클러스터의 포드로 포워딩하는 단계; 상기 하나 이상의 프로세서에 의해, 상기 제1 클러스터의 포드를 삭제하는 단계를 더 포함할 수 있다.

[0013] 상기 방법은 상기 하나 이상의 프로세서에 의해, 상기 제1 클러스터의 제1 작업자 노드가 상기 제2 클러스터로 이동될 하나 이상의 포드를 갖는다고 결정하는 단계; 상기 하나 이상의 프로세서에 의해, 상기 제2 클러스터에 제2 작업자 노드를 생성하는 단계; 상기 하나 이상의 프로세서에 의해, 상기 제1 클러스터의 제1 작업자 노드가

새로운 포드를 추가하는 것을 방지하는 단계; 상기 하나 이상의 프로세서에 의해, 상기 제1 작업자 노드의 하나 이상의 포드를 상기 제2 클러스터의 제2 작업자 노드로 이동시키는 단계; 상기 하나 이상의 프로세서에 의해, 상기 제1 클러스터의 제1 작업자 노드가 상기 제2 클러스터로 이동될 포드를 더 이상 갖지 않는다고 결정하는 단계; 상기 하나 이상의 프로세서에 의해, 상기 제1 클러스터의 제1 작업자 노드를 삭제하는 단계를 더 포함할 수 있다.

[0014] 상기 방법은 상기 하나 이상의 프로세서에 의해, 하나 이상의 워크로드에 대한 요청을 수신하는 단계, 상기 하나 이상의 워크로드는 상기 제1 클러스터에서 실행되는 워크로드 및 상기 제2 클러스터에서 실행되는 워크로드를 포함하며; 상기 하나 이상의 프로세서에 의해, 적어도 하나의 글로벌 로드 밸런서를 사용하여, 상기 수신된 요청을 상기 제1 클러스터에서 실행되는 워크로드와 상기 제2 클러스터에서 실행되는 워크로드 사이의 하나 이상의 워크로드에 할당하는 단계를 더 포함할 수 있다.

[0015] 상기 방법은 상기 하나 이상의 프로세서에 의해, 상기 제2 클러스터에서 실행되는 포드가 상기 제1 클러스터의 스토리지를 참조한다고 결정하는 단계; 상기 하나 이상의 프로세서에 의해, 상기 제2 클러스터에 스토리지를 생성하는 단계, 상기 제1 클러스터의 스토리지와 상기 제2 클러스터의 스토리지는 상이한 위치에 위치되며; 상기 하나 이상의 프로세서에 의해, 스토리지 드라이버를 사용하여, 상기 제2 클러스터 내의 포드와 관련된 데이터에 대한 상기 제2 클러스터의 스토리지를 판독하는 단계; 상기 하나 이상의 프로세서에 의해, 상기 스토리지 드라이버를 사용하여, 상기 제2 클러스터의 포드와 관련된 데이터에 상기 제1 클러스터의 스토리지를 판독하는 단계를 더 포함할 수 있다. 상기 방법은 상기 하나 이상의 프로세서에 의해, 상기 제2 클러스터의 포드에 의해 이루어진 변경을 상기 제2 클러스터의 스토리지에 기록하는 단계; 상기 하나 이상의 프로세서에 의해, 상기 포드에 의해 변경되지 않은 데이터를 상기 제1 클러스터의 스토리지로부터 상기 제2 클러스터의 스토리지로 복사하는 단계를 더 포함할 수 있다.

[0016] 본 개시는 제1 클러스터에서 제2 클러스터로 마이그레이션하기 위한 시스템을 제공하며, 상기 시스템은 하나 이상의 프로세서를 포함하며, 상기 하나 이상의 프로세서는: 하나 이상의 클러스터 제어 평면에 대한 요청을 수신하고, 상기 하나 이상의 클러스터 제어 평면은 상기 제1 클러스터의 제어 평면 및 상기 제2 클러스터의 제어 평면을 포함하며; 상기 수신된 요청의 미리 결정된 부분을 상기 제2 클러스터의 제어 평면에 할당하고, 상기 요청의 나머지 부분을 상기 제1 클러스터의 제어 평면에 할당하고; 상기 제2 클러스터의 제어 평면을 사용하여 상기 요청의 미리 결정된 부분을 처리하고; 상기 요청의 미리 결정된 부분을 처리하는 동안 상기 제2 클러스터에 실패가 있는지를 검출하고; 그리고 상기 제2 클러스터에서 실패를 검출하지 못한 것에 기초하여, 수신된 모든 요청이 제2 클러스터의 제어 평면에 할당될 때까지 미리 결정된 단계에서 상기 제2 클러스터의 제어 평면에 할당된 상기 요청의 미리 결정된 부분을 증가시키도록 구성된다.

[0017] 상기 제1 클러스터 및 상기 제2 클러스터는 상이한 소프트웨어 버전을 동작, 상이한 위치에서 동작, 상이한 클라우드 제공자에 의해 제공되는 상이한 클라우드에서 동작, 적어도 하나의 클라우드가 사용자의 온-프레미스 데이터센터이거나 상이한 네트워크에 연결된 상이한 클라우드에서 동작하는 것 중 적어도 하나일 수 있다.

도면의 간단한 설명

[0018] 도 1a 및 1b는 클러스터에 대한 환경 변경을 구현하기 위한 기존 프로세스를 도시한다.

도 2는 본 개시의 양태들에 따라 클러스터가 동작될 수 있는 예시적 분산 시스템을 도시한다.

도 3은 본 개시의 양태들에 따라 라이브 클러스터 마이그레이션이 발생할 수 있는 예시적 분산 시스템을 도시한다.

도 4는 본 개시의 양태들에 따른 예시적 클러스터를 도시한다.

도 5는 본 개시의 양태들에 따른 라이브 클러스터 마이그레이션에 관련된 예시적 컴포넌트를 도시한다.

도 6은 본 개시의 양태들에 따라 클라우드 내의 라이브 마이그레이션 동안 클러스터 제어 평면에 대한 라우팅 요청의 예시적 구성을 도시한다.

도 7은 본 개시의 양태들에 따라 상이한 클라우드들 간의 라이브 마이그레이션 동안 클러스터 제어 평면에 대한 라우팅 요청의 예시적 구성을 도시한다.

도 8은 본 개시의 양태들에 따라 상이한 위치들 또는 클라우드들 간의 라이브 마이그레이션 동안 클러스터 제어 평면에 대한 스토리지 동기화를 수행하는 예시적 구성을 도시한다.

도 9는 본 개시의 양태들에 따른 워크로드의 마이그레이션의 예시적 구성을 도시한다.

도 10은 본 개시의 양태들에 따라 상이한 위치들 또는 클라우드들 간의 워크로드에 대해 라이브 스토리지 마이그레이션을 수행하는 예시적 구성을 도시한다.

도 11a, 11b 및 11c는 본 개시의 양태들에 따라 클러스터 제어 평면에 대한 예시적 라이브 마이그레이션을 도시하는 타이밍 다이어그램이다.

도 12는 본 개시의 양태들에 따라 워크로드에 대한 예시적 라이브 마이그레이션을 도시하는 타이밍 다이어그램이다.

도 13은 본 개시의 양태들에 따라 마이그레이션 후 액션을 도시하는 타이밍 다이어그램이다.

도 14는 본 개시의 양태들에 따른 예시적 흐름도이다.

발명을 실시하기 위한 구체적인 내용

[0019] 개요

[0020] 이 기술은 일반적으로 분산 컴퓨팅 환경에서 노드의 클러스터의 환경을 수정하는 것과 관련된다. 소프트웨어 업그레이드 또는 위치, 네트워크 또는 클라우드 간 이동과 관련된 환경 변경에 대한 위험과 다운타임을 줄이기 위해, 시스템은 단계적 롤아웃에서 라이브 마이그레이션을 통해 클러스터 환경을 수정하도록 구성된다. 이와 관련하여, 제1, 소스 클러스터가 계속 실행되는 동안, 제2 목적지 클러스터가 생성될 수 있다.

[0021] 라이브 마이그레이션 중에 동작은 소스 클러스터와 목적지 클러스터 모두에서 처리된다. 이와 관련하여, 다양한 동작 및/또는 컴포넌트는 소스 클러스터에 의해 처리되는 것에서 목적지 클러스터에 의해 처리되는 것으로 점진적으로 시프트될 수 있다. 시프트는 단계적 롤아웃일 수 있으며, 각 단계에서, 서로 다른 동작 및/또는 컴포넌트의 세트가 소스 클러스터로부터 목적지 클러스터로 시프트될 수 있다. 또한, 장애 발생시 손상을 완화하기 위해, 각 단계 내에서, 동작 또는 컴포넌트를 소스 클러스터로부터 목적지 클러스터로 시프트하는 것은 점진적이거나 또는 "조용히 일부에 우선 적용될(canaried)" 수 있다. 라이브 마이그레이션은 클러스터의 제어 평면과 클러스터의 워크로드에 대해 수행될 수 있다.

[0022] 예를 들어, 클러스터 제어 평면의 라이브 마이그레이션 동안에, 소스 클러스터의 클러스터 제어 평면과 목적지 클러스터의 클러스터 제어 평면 간에 트래픽이 할당될 수 있다. 이와 관련하여, 소스 클러스터와 목적지 클러스터가 동일한 클라우드에서 동작되는 경우, 클러스터 브릿징 애그리케이터는 사용자 애플리케이션 및/또는 워크로드로부터 API 호출과 같은 수신 요청을 소스 클러스터 및 목적지 클러스터 둘 모두의 클러스터 제어 평면으로 라우팅하도록 구성될 수 있다. 소스 클러스터와 목적지 클러스터가 상이한 클라우드에서 동작되는 경우, 특히 클라우드 중 하나가 클러스터 마이그레이션을 지원하지 않을 수 있는 경우, 클러스터 브릿징 애그리게이터가 없는 클러스터에 하나 이상의 사이드카 컨테이너가 삽입될 수 있다. 이러한 사이드카 컨테이너는 추가 라우팅/재-라우팅을 위해 클러스터 브릿징 애그리게이터가 있는 클러스터로 API 호출을 인터셉트하고 라우팅할 수 있다.

[0023] 클러스터 제어 평면에 대한 요청 트래픽의 할당은 라이브 마이그레이션 동안에 조용히 적용될 수 있다. 예를 들어, 초기에 요청의 미리 결정된 부분이 목적지 클러스터의 클러스터 제어 평면에 할당될 수 있으며, 요청의 나머지 부분은 소스 클러스터의 클러스터 제어 평면에 할당될 수 있다. 목적지 클러스터는 클러스터 제어 평면이 요청의 미리 결정된 부분을 처리하는 동안 모니터링될 수 있다. 실패가 검출되지 않으면, 모든 요청이 결국 목적지 클러스터의 클러스터 제어 평면에 할당될 때까지 목적지 클러스터의 클러스터 제어 평면에 대한 요청의 할당을 점차 증가시킬 수 있다.

[0024] 소스 클러스터와 목적지 클러스터의 클러스터 제어 평면 사이의 요청의 할당은 미리 결정된 규칙에 기초할 수 있다. 예를 들어, 요청은 리소스 유형, 오브젝트 유형 또는 위치에 기초하여 할당될 수 있다. 또한, 요청은 미리 결정된 단계로 할당될 수 있다.

[0025] 예를 들어, 클러스터 제어 평면의 라이브 마이그레이션 동안에, 소스 클러스터의 클러스터 제어 평면과 목적지 클러스터의 클러스터 제어 평면 간에 오브젝트 작동(object actuation이 할당될 수 있다. 실패 발생시 손상을 더욱 완화하기 위해, 오브젝트 작동의 할당이 조용히 적용될 수 있다. 예를 들어, 처음에, 오브젝트 잠금의 미리 결정된 부분이 목적지 클러스터의 제어기에 할당될 수 있고, 오브젝트 잠금의 나머지 부분이 소스 클러스터의 제어기에 할당될 수 있다. 목적지 클러스터는 오브젝트 잠금의 미리 결정된 부분에 의해 잠긴 오브젝트를 작동시키는 동안 모니터링될 수 있다. 실패가 검출되지 않거나 적어도 마이그레이션 전에 소스 클러스터에서 아직

발생하지 않은 추가 실패가 없는 경우, 모든 오브젝트가 목적지 클러스터의 제어기에 의해 사실상 작동될 때까지 목적지 클러스터의 제어기에 대한 오브젝트 잠금의 할당이 증가될 수 있다.

[0026] 또한, 클러스터 제어 평면에 대한 일관된 데이터 스토리지는 라이브 마이그레이션 동안에 유지되어야 한다. 이와 관련하여, 소스 클러스터와 목적지 클러스터가 동일한 데이터센터에 있어 동일한 스토리지 백엔드를 공유하는 경우, 예를 들어 동일한 쿼럼을 조인함으로써 소스 클러스터와 목적지 클러스터의 데이터베이스를 브릿징할 수 있다. 한편, 소스 클러스터와 목적지 클러스터가 서로 다른 위치 또는 클라우드에서 동작되어 서로의 스토리지 백엔드에 액세스할 수 없는 경우, 소스 클러스터와 목적지 클러스터의 데이터베이스는 동기화될 수 있다.

[0027] 또한, 클러스터에서 실행되는 워크로드에 대해 마이그레이션을 수행할 수 있다. 이와 관련하여 워크로드 마이그레이션도 실시간으로 진행될 수 있다. 예를 들어, 목적지 클러스터에 새 노드가 생성되면 목적지 클러스터에 포드가 생성될 수 있다. 소스 클러스터에서 포드를 즉시 삭제하지 않고, 소스 클러스터에서 포드 실행을 일시 중지할 수 있다. 소스 클러스터의 포드 상태는 목적지 클러스터의 포드로 전송될 수 있으며, 실행은 목적지 클러스터의 포드에서 재개될 수 있다. 또한, 글로벌 로드 밸런서는 소스 클러스터와 목적지 클러스터 둘 모두에서 실행되는 워크로드로 요청을 라우팅하도록 구성할 수 있다. 워크로드 마이그레이션이 서로 다른 위치 또는 클라우드 사이에 있는 경우, 워크로드에 대한 라이브 스토리지 마이그레이션을 수행하여 워크로드에 대한 스토리지 위치를 변경할 수 있다.

[0028] 클러스터 제어 평면의 모든 컴포넌트 및/또는 워크로드의 모든 컴포넌트가 목적지 클러스터로 시프트되고 마이그레이션 전에 소스 클러스터에서 아직 발생하지 않은 추가 실패가 없는 경우, 소스 클러스터의 컴포넌트는 할당 해제되거나 삭제될 수 있다. 그러나 라이브 마이그레이션 중 또는 이후에 실패가 검출되면, 라이브 마이그레이션이 중지될 수 있다. 또한 목적지 클러스터에서 다시 소스 클러스터로 롤백이 개시되고, 목적지 클러스터의 컴포넌트가 할당 해제되고 삭제될 수 있다.

[0029] 이 기술은 클러스터 인프라를 수정하기 위해 점진적으로 모니터링되는 롤아웃 프로세스를 제공하기 때문에 이점이 있다. 단계별 및 조용히 일부에 우선 적용되는(canaried) 롤아웃 프로세스는 문제가 발생할 경우 업그레이드를 중지할 수 있는 더 많은 기회를 제공하여 대규모 손해를 방지한다. 클러스터 제어 평면에 대한 요청 및/또는 워크로드에 대한 요청과 같은 트래픽 할당은 동시에 실행되는 소스 및 목적지 클러스터 사이에서 업그레이드 중에 다운 타임을 줄이거나 제거할 수 있다. 또한 트래픽 할당으로 인해 클라이언트의 관점에서 라이브 마이그레이션 중에 하나의 클러스터만 존재하는 것처럼 보일 수 있다. 업그레이드가 실패한 경우, 성공적인 업그레이드가 완료되지 않는 한 소스 클러스터가 삭제되지 않으므로 시스템은 롤백 옵션을 제공한다. 이 기술은 또한 상이한 위치에 위치된 클러스터들 간 뿐만 아니라 클라우드 중 하나가 라이브 마이그레이션을 지원하지 않는 다른 클라우드에서 동작되는 클러스터들 간에 라이브 마이그레이션을 가능하게 하는 구성을 제공한다.

[0030] 예시적 시스템들

[0031] 도 2는 클러스터가 동작될 수 있는 예시적 분산 시스템(200)을 보여주는 기능도이다. 도시된 바와 같이, 시스템(200)은 네트워크(290)에 연결된 서버 컴퓨터(210, 220, 230, 240)와 같은 다수의 컴퓨팅 디바이스를 포함할 수 있다. 예를 들어, 서버 컴퓨터(210, 220, 230, 240)는 클라우드 제공자에 의해 동작되는 클라우드 컴퓨팅 시스템의 일부일 수 있다. 클라우드 제공자는 스토리지(280) 및 스토리지(282)와 같은 하나 이상의 스토리지를 추가로 유지할 수 있다. 또한 도시된 바와 같이, 시스템(200)은 네트워크(290)를 통해 서버 컴퓨터(210, 220, 230, 240)와 통신할 수 있는 클라이언트 컴퓨터(250)와 같은 하나 이상의 클라이언트 컴퓨팅 디바이스를 포함할 수 있다.

[0032] 서버 컴퓨터(210, 220, 230, 240) 및 스토리지(280, 282)는 하나 이상의 데이터센터에서 클라우드 제공자에 의해 유지될 수 있다. 예를 들어, 도시된 바와 같이, 서버 컴퓨터(210, 220) 및 스토리지(280)는 데이터센터(260)에 위치할 수 있고, 서버 컴퓨터(230, 240) 및 스토리지(282)는 다른 데이터센터(270)에 위치할 수 있다. 데이터센터(260, 270) 및/또는 서버 컴퓨터(210, 220, 230, 240)는 다른 도시, 주, 국가, 대륙 등과 같이 서로 상당한 거리에 위치할 수 있다. 또한, 데이터센터(260, 270) 내에서 하나 이상의 영역 또는 구역이 있을 수 있다. 예를 들어, 영역 또는 구역은 적절한 속성에 기초하여 논리적으로 분할될 수 있다.

[0033] 클러스터는 분산 시스템(200)에서 동작될 수 있다. 예를 들어, 클러스터는 서버 컴퓨터(210)의 프로세서(212) 또는 서버 컴퓨터(230 및 240)의 프로세서(232 및 242)와 같은 데이터센터의 하나 이상의 프로세서에 의해 구현될 수 있다. 또한, 영구 디스크("PD")와 같은 클러스터의 상태에 대한 지속적이고 일관된 기록을 유지하기 위한 스토리지 시스템은 스토리지(280, 282) 또는 서버 컴퓨터(210, 220, 230, 240)의 데이터(218, 228)와 같은 클

라우드 컴퓨팅 시스템에서 구현될 수 있다.

[0034] 서버 컴퓨터(210, 220, 230, 240)는 유사하게 구성될 수 있다. 예를 들어, 도시된 바와 같이, 서버 컴퓨터(210)는 하나 이상의 프로세서(212), 메모리(214) 및 일반적 범용 컴퓨터에 존재하는 다른 컴포넌트를 포함할 수 있다. 메모리(214)는 프로세서(212)에 의해 액세스가능한 정보를 저장할 수 있고, 프로세서(212)에 의해 실행될 수 있는 명령어(216)를 포함한다. 메모리는 또한 프로세서(212)에 의해 검색, 조작 또는 저장될 수 있는 데이터(218)를 포함할 수 있다. 메모리(214)는 하드 드라이브, 솔리드 스테이트 드라이브, 테이프 드라이브, 광학 스토리지, 메모리 카드, ROM, RAM, DVD, CD-ROM, 쓰기 가능 및 읽기 전용 메모리와 같이, 프로세서(212)에 의해 액세스가능한 정보를 저장할 수 있는 비일시적 컴퓨터 판독가능 매체의 유형일 수 있다. 프로세서(212)는 잘 알려진 프로세서 또는 덜 알려진 다른 유형의 프로세서일 수 있다. 대안적으로, 프로세서(212)는 GPU 또는 ASIC, 예를 들어 TPU와 같은 전용 제어기일 수 있다.

[0035] 명령어(216)는 컴퓨팅 디바이스 코드와 같이 직접 또는 스크립트와 같이 간접적으로 프로세서(212)에 의해 실행될 명령어의 세트일 수 있다. 이와 관련하여, 용어 "명령어", "단계" 및 "프로그램"은 본 명세서에서 상호교환적으로 사용될 수 있다. 명령어는 프로세서(212)에 의한 직접 프로세싱을 위해 오브젝트 코드 포맷으로 또는 스크립트 또는 요구에 따라 인터프리트되거나 미리 컴파일되는 독립적인 소스 코드 모듈의 집합을 포함하는 다른 유형의 컴퓨터 언어로 저장될 수 있다. 명령어의 기능, 방법 및 루틴은 앞의 예와 아래의 예시적 방법에서 더 자세히 설명된다. 명령어(216)는 본 명세서에 설명된 예시적 구성 중 임의의 것을 포함할 수 있다.

[0036] 데이터(218)는 명령어(216)에 따라 프로세서(212)에 의해 검색, 저장 또는 수정될 수 있다. 예를 들어, 시스템 및 방법은 특정한 데이터 구조에 의해 제한되지 않지만, 데이터(218)는 컴퓨팅 레지스터에, 복수의 상이한 필드 및 레코드를 갖는 테이블로서 관계형 또는 비-관계형 데이터베이스에 또는 JSON, YAML, 프로토 또는 XML 문서에 저장될 수 있다. 데이터(218)는 또한 이진 값, ASCII 또는 유니코드와 같은(그러나 이에 한정되지 않음) 컴퓨터 판독가능 포맷으로 포맷팅될 수 있다. 또한, 데이터(218)는 숫자, 설명 텍스트, 독점 코드, 포인터, 다른 네트워크 위치를 포함하여 다른 메모리에 저장된 데이터에 대한 참조 또는 관련 데이터를 계산하기 위해 함수에 의해 사용되는 정보와 같은 관련 정보를 식별하기에 충분한 정보를 포함할 수 있다.

[0037] 도 2는 프로세서(212) 및 메모리(214)가 동일한 블록 내에 있는 것으로 기능적으로 도시하지만, 프로세서(212) 및 메모리(214)는 실제로 동일한 물리적 하우징 내에 저장되거나 저장되지 않을 수 있는 다중 프로세서 및 메모리를 포함할 수 있다. 예를 들어, 명령어(216) 및 데이터(218) 중 일부는 이동식 CD-ROM에 저장될 수 있고 다른 것은 읽기 전용 컴퓨터 칩 내에 저장될 수 있다. 명령어 및 데이터의 일부 또는 전부는 프로세서(212)로부터 물리적으로 멀리 떨어져 있지만 여전히 액세스할 수 있는 위치에 저장될 수 있다. 유사하게, 프로세서(212)는 병렬로 동작하거나 동작하지 않을 수 있는 프로세서의 집합을 포함할 수 있다. 서버 컴퓨터(210, 220, 230, 240)는 서버 컴퓨터(210, 220, 230, 240)에 의해 실행되는 동작 및 프로그램에 대한 시간 측정에 사용될 수 있는, 타이밍 정보를 제공하는 하나 이상의 내부 클록을 각각 포함할 수 있다.

[0038] 서버 컴퓨터(210, 220, 230, 240)는 DAS(Direct Attached Storage), NAS(Network Attached Storage), SAN(Storage Area Network), 파이버 채널(FC), 이더넷을 통한 파이버 채널(FCoE), 혼합 아키텍처 네트워크 등을 포함하나 이에 한정되지 않는 다양한 아키텍처 및 기술을 구현할 수 있다. 일부 예에서, 서버 컴퓨터(210, 220, 230, 240)는 가상화된 환경일 수 있다.

[0039] 서버 컴퓨터(210, 220, 230, 240) 및 클라이언트 컴퓨터(250)는 각각 네트워크(290)의 하나의 노드에 있을 수 있고, 네트워크(290)의 다른 노드와 직접 및 간접적으로 통신할 수 있다. 예를 들어, 서버 컴퓨터(210, 220, 230, 240)는 네트워크(290)를 사용하여 클라이언트 컴퓨터(250)에서 실행되는 애플리케이션에 정보를 전송하도록 네트워크(290)를 통해 클라이언트 컴퓨터(250)와 통신할 수 있는 웹 서버를 포함할 수 있다. 서버 컴퓨터(210, 220, 230, 240)는 또한 클라이언트 컴퓨터(250)로 데이터를 수신, 프로세싱 및 전송하기 위해 네트워크(290)의 다른 노드와 정보를 교환할 수 있는 하나 이상의 로드 밸런싱 서버 팜의 컴퓨터일 수 있다. 도 2는 몇 대의 서버 컴퓨터(210, 220, 230, 240), 스토리지(280, 282) 및 데이터센터(260, 270)만 도시되어 있지만, 일반적인 시스템은 다수의 연결된 서버 컴퓨터, 다수의 스토리지 및/또는 각각이 네트워크(290)의 상이한 노드에 있는 다수의 데이터센터를 포함할 수 있음을 이해해야 한다.

[0040] 클라이언트 컴퓨터(250)는 또한 프로세서(252), 메모리(254), 명령어(256) 및 데이터(258)를 갖는 서버 컴퓨터(210, 220, 230, 240)와 유사하게 구성될 수 있다. 클라이언트 컴퓨터(250)는 CPU, 데이터 및 명령어를 저장하는 메모리(예: RAM 및 내부 하드 드라이브), 입력 및/또는 출력 디바이스, 센서, 클록 등과 같은 개인용 컴퓨팅 디바이스와 관련하여 일반적으로 사용되는 모든 컴포넌트를 가질 수 있다. 클라이언트 컴퓨터(250)는 풀 사이즈

개인용 컴퓨팅 디바이스를 포함할 수 있으며, 대안적으로 인터넷과 같은 네트워크를 통해 서버와 데이터를 무선으로 교환할 수 있는 모바일 컴퓨팅 디바이스를 포함할 수 있다. 예를 들어, 클라이언트 컴퓨터(250)는 데스크톱 또는 랩톱 컴퓨터, 모바일폰 또는 무선 지원 PDA, 태블릿 PC, 인터넷을 통해 정보를 얻을 수 있는 넷북 또는 웨어러블 컴퓨팅 디바이스와 같은 디바이스일 수 있다.

[0041] 클라이언트 컴퓨터(250)는 애플리케이션 인터페이스 모듈(251)을 포함할 수 있다. 애플리케이션 인터페이스 모듈(251)은 서버 컴퓨터(210, 220, 230, 240)와 같은 하나 이상의 서버 컴퓨터에 의해 이용 가능한 서비스에 액세스하는데 사용될 수 있다. 애플리케이션 인터페이스 모듈(251)은 서버와 클라이언트가 서로 통신하게 하는데 사용되는 서브 루틴, 데이터 구조, 오브젝트 클래스 및 기타 유형의 소프트웨어 컴포넌트를 포함할 수 있다. 일 양태에서, 애플리케이션 인터페이스 모듈(251)은 당업계에 공지된 여러 유형의 운영 체제와 함께 동작할 수 있는 소프트웨어 모듈일 수 있다. 메모리(254)는 애플리케이션 인터페이스 모듈(251)에 의해 액세스되는 데이터(258)를 저장할 수 있다. 데이터(258)는 또한 클라이언트 컴퓨터(250)에 연결될 수 있는 디스크, 테이프, SD 카드 또는 CD-ROM과 같은 이동식 매체에 저장될 수 있다.

[0042] 또한 도 2에 도시된 바와 같이, 클라이언트 컴퓨터(250)는 키보드, 마우스, 기계식 액추에이터, 소프트 액추에이터, 터치스크린, 마이크로폰, 센서 및/또는 기타 컴포넌트와 같은 하나 이상의 사용자 입력(253)을 포함할 수 있다. 클라이언트 컴퓨터(250)는 사용자 디스플레이, 터치스크린, 하나 이상의 스피커, 트랜스듀서 또는 기타 오디오 출력, 햅틱 인터페이스 또는 비-시각적 및 비-청각적 정보를 사용자에게 제공하는 기타 촉각 피드백과 같은 하나 이상의 출력 디바이스(255)를 포함할 수 있다. 또한, 도 2에서 하나의 클라이언트 컴퓨터(250)만이 도시되었지만, 일반적 시스템은 네트워크(290)의 다른 노드에 있는 다수의 클라이언트 컴퓨터를 제공할 수 있음을 이해해야 한다. 예를 들어, 시스템(200)의 서버 컴퓨터는 다수의 클라이언트 컴퓨터 상의 애플리케이션에 대한 워크로드를 실행할 수 있다.

[0043] 메모리(214)와 마찬가지로, 스토리지(280, 282)는 하드 드라이브, 메모리 카드, ROM, RAM, DVD, CD-ROM, 쓰기 가능 및 읽기 전용 메모리와 같이, 서버 컴퓨터(210, 220, 230, 240) 및 클라이언트 컴퓨터(250) 중 하나 이상에 의해 액세스 가능한 정보를 저장할 수 있는 임의의 유형의 컴퓨터화된 스토리지일 수 있다. 일부 예에서, 스토리지(280, 282)는 하나 이상의 영구 디스크("PD")를 포함할 수 있다. 또한, 스토리지(280, 282)는 데이터가 동일한 또는 상이한 지리적 위치에 물리적으로 위치될 수 있는 복수의 상이한 스토리지 디바이스에 저장되는 분산 스토리지 시스템을 포함할 수 있다. 스토리지(280, 282)는 도 2에 도시된 바와 같이 네트워크(290)를 통해 컴퓨팅 디바이스에 연결될 수 있고 및/또는 임의의 서버 컴퓨터(210, 220, 230, 240) 및 클라이언트 컴퓨터(250)에 직접 연결될 수 있다.

[0044] 서버 컴퓨터(210, 220, 230, 240) 및 클라이언트 컴퓨터(250)는 네트워크(290)를 통해서와 같이, 직접 및 간접 통신이 가능할 수 있다. 예를 들어, 인터넷 소켓을 사용하여, 클라이언트 컴퓨터(250)는 인터넷 프로토콜 스위트를 통해, 원격 서버 컴퓨터(210, 220, 230, 240)에서 동작하는 서비스에 연결할 수 있다. 서버 컴퓨터(210, 220, 230, 240)는 정보를 송수신하기 위한 개시 연결을 수락할 수 있는 리스닝 소켓을 설정할 수 있다. 네트워크(290) 및 중간 노드(intervening node)는 인터넷, 월드 와이드 웹, 인트라넷, 가상 사설망, 광역 네트워크, 로컬 네트워크, 하나 이상의 회사에 독점적인 통신 프로토콜을 사용하는 사설 네트워크, 이더넷, WiFi(예를 들어, 802.81, 802.81b, g, n 또는 기타 이러한 표준) 및 HTTP 및 이들의 다양한 조합을 포함한 다양한 구성 및 프로토콜을 포함할 수 있다. 이러한 통신은 모뎀(예: 전화 접속, 케이블 또는 광섬유) 및 무선 인터페이스와 같은 다른 컴퓨터와 데이터를 주고 받을 수 있는 디바이스에 의해 지원될 수 있다.

[0045] 도 3은 라이브 클러스터 마이그레이션이 발생할 수 있는 예시적 분산 시스템(300)을 도시하는 기능도이다. 분산 시스템(300)은 제1 클라우드(310) 및 제2 클라우드(320)를 포함한다. 도시된 바와 같이, 클라우드(310)는 데이터센터(260, 270)의 서버 컴퓨터(210, 220, 230, 240) 및 네트워크(290)에 연결된 스토리지(280, 282)를 포함할 수 있다. 클라이언트 컴퓨터(250)와 같은 하나 이상의 클라이언트 컴퓨터는 네트워크(290)에 연결되어 클라우드(310)에 의해 제공되는 서비스를 사용할 수 있다. 또한 도시된 바와 같이, 클라우드(320)는 데이터센터(330)와 같은 하나 이상의 데이터센터에 구성된 서버 컴퓨터(332, 334) 및 네트워크(390)에 연결된 스토리지(380)와 같은 하나 이상의 스토리지와 같은 컴퓨팅 디바이스를 유사하게 포함할 수 있다. 클라이언트 컴퓨터(350)와 같은 하나 이상의 클라이언트 컴퓨터는 네트워크(390)에 연결되어 클라우드(320)에 의해 제공되는 서비스를 사용할 수 있다. 도 3는 몇 대의 서버 컴퓨터, 데이터센터, 스토리지 및 클라이언트 컴퓨터만 도시되어 있지만, 일반적인 시스템은 다수의 연결된 서버 컴퓨터, 다수의 데이터센터, 다수의 스토리지 및/또는 각각이 네트워크의 상이한 노드에 있는 다수의 클라이언트 컴퓨터를 포함할 수 있음을 이해해야 한다.

[0046] 클라우드(310) 및 클라우드(320)는 상이한 클라우드 제공자에 의해 동작될 수 있다. 이와 같이, 클라우드(310) 및 클라우드(320)는 클라우드(310) 및 클라우드(320)에서 동작되는 클러스터가 상이한 소프트웨어 환경에서 실행되도록 상이한 구성을 가질 수 있다. 또한, 클라우드(310) 및 클라우드(320)에 의해 호스팅되는 클러스터는 임의의 스토리지 백엔드를 공유하거나 공유하지 않을 수 있으며, 동일한 네트워크에 연결되거나 동일한 물리적 위치에 있을 수 있다. 이와 같이, 클라우드(310) 및 클라우드(320) 상의 클러스터는 서로 리소스, 소프트웨어 컴포넌트 및/또는 구성을 수정하거나 액세스할 수 없을 수 있다. 일부 경우에, 클라우드(310) 및 클라우드(320) 중 하나 또는 둘 모두는 사용자에 의해 자체 관리될 수 있다.

[0047] 분산 시스템(300)에서의 라이브 클러스터 마이그레이션은 임의의 다양한 방식으로 발생할 수 있다. 예를 들어, 클러스터가 데이터센터(260)에서 실행되는 동안, 클라우드(310)에 대한 클라우드 제공자는 클라우드 제어 평면, 마스터 노드에서 실행되는 클러스터 제어 평면 또는 작업자 노드에 대한 소프트웨어 업그레이드를 도입할 수 있다. 이와 같이, 소프트웨어 업그레이드를 준수하는 데이터센터(260)에서 생성된 목적지 클러스터로 클러스터의 오브젝트에 대해 마이그레이션이 수행될 수 있다. 그러한 경우에, 마이그레이션은 동일한 데이터센터(260), 동일한 네트워크(290) 및 동일한 클라우드(310) 내에 있다.

[0048] 다른 예로, 라이브 클러스터 마이그레이션은 물리적 위치 간 이동을 포함할 수 있다. 예를 들어, 클라우드(310)에 대한 클라우드 제공자가 리소스를 재배치하거나 클러스터에서 실행되는 애플리케이션의 개발자가 다른 위치로 이동하기를 원할 수 있다. 이와 같이, 데이터센터(260)의 클러스터의 오브젝트를 데이터센터(270)에서 생성된 목적지 클러스터로 마이그레이션이 수행될 수 있다. 그러한 경우에, 마이그레이션은 여전히 동일한 네트워크(290) 및 동일한 클라우드(310) 내에 있을 수 있다.

[0049] 그러나 때로는 사용자가 자체 관리하거나 한 클라우드 운영자가 운영하는 하나의 클라우드로부터 다른 클라우드 운영자가 운영하는 다른 클라우드로 스위칭하고자 할 수 있다. 예를 들어, 클라우드(320) 상의 클러스터 내의 오브젝트에 대해 클라우드(310)에서 생성된 목적지 클러스터로의 라이브 마이그레이션이 수행될 수 있다. 클라우드를 변경하는 것 외에도, 이러한 마이그레이션은 일부 경우에 네트워크 변경 및/또는 지역 변경을 포함할 수 있다.

[0050] 아래의 예에서 추가로 설명되는 바와 같이, 클라우드 간의 마이그레이션을 위해, 클라우드(310) 및 클라우드(320) 중 하나 또는 둘 모두가 라이브 클러스터 마이그레이션을 수행하기 위한 기능으로 구성될 수 있다. 예를 들어, 클라우드(310) 및 클라우드(320) 둘 모두가 라이브 클러스터 마이그레이션을 수행하기 위한 기능을 포함하는 경우에, 이들 기능은 함께 라이브 클러스터 마이그레이션을 용이하게 할 수 있다. 클라우드(310)가 라이브 클러스터 마이그레이션을 수행하기 위한 기능을 포함하는 반면, 클라우드(320)는 라이브 클러스터 마이그레이션을 수행하기 위한 기능을 포함하지 않는 경우, 클라우드(310) 및 클라우드(310) 상의 마이그레이션 클러스터는 마이그레이션을 용이하게 하기 위해 추가 도구 및 방법을 사용할 수 있지만, 그러한 것들이 클라우드(320) 및 클라우드(320)의 마이그레이션 클러스터에 사용가능하지 않을 수 있다.

[0051] 도 4은 예시적 클러스터(400)를 도시하는 기능 다이어그램이다. 예를 들어, 개발자와 같은 사용자는 애플리케이션을 설계하고 도 2의 클라이언트 컴퓨터(250)와 같은 클라이언트 컴퓨터를 사용하여 애플리케이션에 대한 구성 데이터를 제공할 수 있다. 도 3의 클라우드(310)와 같은 클라우드에 의해 제공되는 컨테이너 오케스트레이션 아키텍처는 애플리케이션의 다양한 서비스를 컨테이너로 패키징하도록 구성될 수 있다. 컨테이너 오케스트레이션 아키텍처는 컨테이너에 리소스를 할당하고, 컨테이너가 제공하는 서비스를 로드 밸런싱하고, 컨테이너를 확장(예: 복제 및 삭제)하도록 구성할 수 있다.

[0052] 도 4에 도시된 바와 같이, 컨테이너 오케스트레이션 아키텍처는 마스터 노드(410)와 같은 하나 이상의 마스터 노드와 작업자 노드(420) 및 작업자 노드(430)와 같은 복수의 작업자 노드를 포함하는 클러스터(400)로 구성될 수 있다. 클러스터(400)의 각 노드는 물리적 머신 또는 가상 머신에서 실행될 수 있다. 클러스터(400)는 시스템(200)과 같은 분산 시스템에서 실행될 수 있다. 예를 들어, 클러스터(400)의 노드는 도 2에 도시된 데이터센터(260)의 하나 이상의 프로세서에서 실행될 수 있다. 마스터 노드(410)는 작업자 노드(420, 430)를 제어할 수 있다. 작업자 노드(420, 430)는 사용자 애플리케이션의 일부를 형성하는 컴퓨터 코드 및 프로그램 런타임의 컨테이너를 포함할 수 있다.

[0053] 또한 도시된 바와 같이, 일부 경우에, 컨테이너는 하나 이상의 포드로 더 구성될 수 있다. 예를 들어, 도 4에 도시된 바와 같이, 작업자 노드(420)는 컨테이너(421, 423, 425)를 포함할 수 있으며, 여기서 컨테이너(423 및 425)는 포드(427)로 구성되는 반면, 작업자 노드(430)는 컨테이너(431, 433, 435)를 포함할 수 있으며, 여기서 컨테이너(431 및 433)는 포드(437)로 구성된다. 작업자 노드의 컨테이너 및 포드에서 다양한 워크로드가 실행될

수 있고, 예를 들어 워크로드는 웹사이트 또는 애플리케이션 프로세스에 대한 콘텐츠를 제공할 수 있다. 포드는 애플리케이션 사용자 또는 웹사이트의 방문자와 같은 워크로드의 사용자로부터 네트워크 트래픽에 포드를 노출하는 "서비스"에 속할 수 있다. 하나 이상의 로드 밸런서가 트래픽, 예를 들어 서비스로부터의 요청을 클러스터(400)에서 실행되는 워크로드로 분산하도록 구성될 수 있다. 예를 들어, 트래픽은 클러스터(400)의 작업자 노드에서 포드간에 분산될 수 있다.

[0054] 또한, 작업자 노드(420)와 같은 일부 노드는 노드 풀(429)과 같은 노드 풀의 일부로서 논리적으로 구성될 수 있다. 예를 들어, 노드 풀은 메모리 크기, CPU/GPU 연결 등과 같은 하나 이상의 속성을 공유하는 노드의 그룹일 수 있다. 일부 경우에, 노드 풀의 모든 노드가 동일한 데이터센터, 데이터센터 내의 동일한 영역/구역 등일 수 있는, 클라우드의 동일한 위치에 위치될 수 있다.

[0055] 마스터 노드(410)는 작업자 노드(420, 430)의 워크로드 및 리소스를 관리하도록 구성될 수 있다. 이와 관련하여, 마스터 노드(410)는 클러스터의 제어 평면의 일부를 형성하는 다양한 소프트웨어 컴포넌트 또는 프로세스를 포함할 수 있다. 예를 들어, 도시된 바와 같이, 마스터 노드(410)는 서로 통신하는 API 서버(440), 데이터베이스(470), 제어기 관리자(480) 및 스케줄러(490)를 포함할 수 있다.

[0056] 마스터 노드(410)는 하나만 도시되어 있지만, 클러스터(400)는 복수의 마스터 노드를 추가로 포함할 수 있다. 예를 들어, 마스터 노드(410)는 복수의 마스터 노드를 생성하기 위해 복제될 수 있다. 클러스터(400)는 복수의 클러스터 제어 평면 프로세스를 포함할 수 있다. 예를 들어, 클러스터(400)는 복수의 API 서버, 복수의 데이터베이스 등을 포함할 수 있다. 이 경우, 복제 마스터 노드의 대다수와 같은 복제 마스터 노드의 쿼럼은 클러스터(400)가 클러스터(400)의 임의의 상태를 수정하는 것에 동의해야 한다. 또한, 클러스터(400)가 다수의 API 서버들 간에 API 호출과 같은 요청 할당을 위해 실행되는 클라우드에 하나 이상의 로드 밸런서가 제공될 수 있다. 복수의 마스터 노드는 하나 이상의 마스터 노드가 실패하더라도 클러스터(400)를 계속 관리함으로써 클러스터(400)의 성능을 개선할 수 있다. 일부 경우에, 복수의 마스터 노드는 서로 다른 물리적 및/또는 가상 머신으로 분산될 수 있다.

[0057] API 서버(440)는 사용자 애플리케이션 또는 작업자 노드에서 실행되는 워크로드로부터 들어오는 API 호출과 같은 요청을 수신하고, 이러한 API 호출을 처리하기 위한 워크로드를 실행하도록 작업자 노드(420, 430)를 관리하도록 구성될 수 있다. 도시된 바와 같이, API 서버(440)는 내장 리소스 서버(460) 및 확장 서버(462)와 같은 다중 서버를 포함할 수 있다. 또한 도시된 바와 같이, API 서버(440)는 들어오는 요청을 API 서버(440)의 적절한 서버로 라우팅하도록 구성된 애그리게이터(450)를 포함할 수 있다. 예를 들어, 사용자 애플리케이션에서 API 호출이 들어오면, 애그리게이터(450)는 API 호출이 클라우드의 내장 리소스에 의해 처리될 것인지 또는 확장인 리소스에 의해 처리될 것인지를 결정할 수 있다. 이 결정에 기초하여, 애그리게이터(450)는 API 호출을 내장 리소스 서버(460) 또는 확장 서버(462)로 라우팅할 수 있다.

[0058] API 서버(440)는 데이터베이스(470)에 저장된 오브젝트를 구성 및/또는 업데이트할 수 있다. API 서버(440)는 클러스터에 다른 API 서버를 포함하는, 클러스터의 다른 컴포넌트에 의해 이해, 서비스 및/또는 저장되기 위해 클러스터의 API 오브젝트가 반드시 준수해야 하는 형식을 포함할 수 있는 스키마에 따라 그렇게 할 수 있다. 오브젝트는 컨테이너, 컨테이너 그룹, 복제 컴포넌트 등에 대한 정보를 포함할 수 있다. 예를 들어, API 서버(440)는 클러스터(400) 내의 다양한 아이템의 상태 변경을 알리고 상기 변경에 기초하여 데이터베이스(470)에 저장된 오브젝트를 업데이트하도록 구성될 수 있다. 이와 같이, 데이터베이스(470)는 클러스터(400)의 전체 상태의 표시일 수 있는 클러스터(400)에 대한 구성 데이터를 저장하도록 구성될 수 있다. 예를 들어, 데이터베이스(470)는 다수의 오브젝트를 포함할 수 있고, 오브젝트는 의도 및 상태와 같은 하나 이상의 상태를 포함할 수 있다. 예를 들어, 사용자는 클러스터(400)에 대한 원하는 상태(들)와 같은 구성 데이터를 제공할 수 있다.

[0059] API 서버(440)는 클러스터(400)의 의도 및 상태를 제어기 관리자(480)에 제공하도록 구성될 수 있다. 제어기 관리자(480)는 원하는 상태(들)를 향해 클러스터(400)를 구동하기 위해 제어 루프를 실행하도록 구성될 수 있다. 이와 관련하여, 제어기 관리자(480)는 API 서버(440)를 통해 클러스터(400)의 노드에 의해 공유된 상태(들)를 감시하고, 현재 상태를 원하는 상태(들)로 이동시키기 위해 시도할 수 있다. 제어기 관리자(480)는 노드 관리(예: 노드 초기화, 노드에 대한 정보 획득, 응답하지 않는 노드에 대한 확인 등), 컨테이너 및 컨테이너 그룹의 복제 관리 등을 포함하는 임의의 다수의 기능을 수행하도록 구성될 수 있다.

[0060] API 서버(440)는 클러스터(400)의 의도 및 상태를 스케줄러(490)에 제공하도록 구성될 수 있다. 예를 들어, 스케줄러(490)는 워크로드가 가용 리소스를 초과하여 스케줄링되지 않도록 보장하기 위해 각 작업자 노드에서 리

소스 사용을 추적하도록 구성될 수 있다. 이를 위해, 스케줄러(490)에 리소스 요구 사항, 리소스 가용성 및 서비스 품질, 친화성/반친화성 요구 사항, 데이터 지역성 등과 같은 기타 사용자 제공 제약 및 정책 지시가 제공될 수 있다. 이와 같이, 스케줄러(490)의 역할은 리소스 공급을 워크로드 요구에 일치시키는 것일 수 있다.

[0061] API 서버(440)는 작업자 노드(420, 430)와 통신하도록 구성될 수 있다. 예를 들어, API 서버(440)는 데이터베이스(470)의 구성 데이터가 컨테이너(421, 423, 425, 431, 433, 435)와 같은 작업자 노드(420, 430)의 컨테이너의 데이터와 일치하도록 구성될 수 있다. 예를 들어, 도시된 바와 같이, API 서버(440)는 컨테이너 관리자(422, 432)와 같은 작업자 노드의 컨테이너 관리자와 통신하도록 구성될 수 있다. 컨테이너 관리자(422, 432)는 마스터 노드(410)로부터의 지시에 기초하여 컨테이너를 시작, 중지 및/또는 유지하도록 구성될 수 있다. 다른 예로서, API 서버(440)는 또한 프록시(424, 434)와 같은 작업자 노드의 프록시와 통신하도록 구성될 수 있다. 프록시(424, 434)는 네트워크 또는 다른 통신 채널을 통해 라우팅 및 스트리밍(TCP, UDP, SCTP 등)을 관리하도록 구성될 수 있다. 예를 들어, 프록시(424, 434)는 작업자 노드(420, 430) 간의 데이터 스트리밍을 관리할 수 있다.

[0062] 도 5는 라이브 마이그레이션에 관련된 두 클러스터의 컴포넌트 예를 도시한다. 도 5는 오브젝트가 마이그레이션될 소스 클러스터인 제1 클러스터(400)와 오브젝트가 마이그레이션될 목적지 클러스터인 제2 클러스터(500)를 도시한다. 도 5는 또한 클러스터(400) 및 클러스터(500) 둘 모두 복제된 마스터 노드를 가지는 것으로 도시하며, 따라서 클러스터(400) 및 클러스터(500)은 둘 모두 다중 API 서버(440, 442, 540, 542) 및 대응하는 애그리게이터(450, 452, 550, 552)와 함께 도시된다. 설명의 편의를 위해 2개의 복제본만이 도 5에 도시되어 있지만, 다수의 복제본 중 임의의 것이 생성될 수 있음을 이해해야 한다.

[0063] 목적지 클러스터(500)는 소스 클러스터(400)와 다른 환경에서 실행된다. 도 3과 관련하여 위에서 설명한 바와 같이, 상이한 환경은 상이한 소프트웨어 버전, 상이한 데이터센터의 물리적 위치, 상이한 네트워크, 상이한 클라우드의 상이한 클라우드 제어 평면 등일 수 있다. 소스 클러스터를 삭제하고 변경할 목적지 클러스터를 만드는 대신, 도 1a-b에 도시된 것과 같은 환경에서, 환경의 변경은 소스 클러스터(400)로부터 목적지 클러스터(500)로 다양한 오브젝트의 라이브 마이그레이션에 의해 수행될 수 있으며, 클러스터(400 및 500)는 모두 여전히 실행 중이다.

[0064] 라이브 마이그레이션 동안, 클러스터 제어 평면에 대한 요청은 소스 클러스터(400)와 목적지 클러스터(500) 사이에 할당될 수 있다. 예를 들어, API 호출과 같은 트래픽은 소스 클러스터(400)의 API 서버(440, 442)와 목적지 클러스터(500)의 API 서버(540, 542) 사이에 할당될 수 있다. 아래에서 상세히 설명하는 바와 같이, 이는 애그리게이터(450, 452, 550, 552)에 대한 수정(도 6 참조) 또는 API 트래픽을 인터셉트하는 컴포넌트를 추가(도 7 참조)함으로써 달성될 수 있다. 또한, 클러스터(400)로 라우팅되는 API 호출을 처리하기 위해, 클러스터(400)는 작업자 노드 및 오브젝트의 복제를 관리하는 것과 같이 클러스터(400)의 리소스를 관리하기 위해 제어기(580)를 실행할 수 있다. 유사하게, 클러스터(500)로 라우팅되는 API 호출을 처리하기 위해, 클러스터(500)는 제어기(582)를 실행하여 클러스터(500)의 리소스를 관리할 수 있다.

[0065] 또한 아래에서 상세히 설명되는 바와 같이, 클러스터(400 및 500) 간의 라이브 마이그레이션은 데이터베이스(470) 및 데이터베이스(570)에서 클러스터 제어 평면에 대해 저장된 오브젝트를 처리하는 것을 포함할 수 있다. 예를 들어, 클러스터(400 및 500)가 동일한 데이터센터에 있고 따라서 동일한 스토리지 백엔드를 공유하는 경우, 데이터베이스(470)와 데이터베이스(570)가 브릿징될 수 있다. 한편, 클러스터(400)와 클러스터(500)가 서로 다른 위치 또는 클라우드에 있어 서로의 스토리지 백엔드에 액세스할 수 없는 경우, 데이터베이스(470)와 데이터베이스(570)를 동기화해야 할 수 있다(도 8 참조).

[0066] 클러스터 제어 평면에 대한 마이그레이션 외에도, 소스 클러스터(400)에서 실행되는 워크로드(581) 및 목적지 클러스터에서 실행되는 워크로드(583)와 같이 클러스터에서 실행되는 워크로드에 대해 라이브 마이그레이션이 수행될 수 있다. 워크로드에 대한 API 호출과 같은 워크로드에 대한 요청은 예를 들어 글로벌로드 밸런서를 사용하여 소스 클러스터(400)와 목적지 클러스터(500) 사이에서 라우팅될 수 있다(도 9 참조). 또한 워크로드를 위한 스토리지 위치는 다른 위치 또는 다른 클라우드 간의 마이그레이션을 위해 변경되어야 할 수 있다(그림 10 참조).

[0067] 또한, 도 5에 도시된 바와 같이, 코디네이터(590)는 예를 들어 클라우드(310)에 대한 클라우드 제공자에 의해 제공될 수 있으며, 이는 라이브 마이그레이션을 구현하기 위한 다양한 규칙을 포함한다. 이와 관련하여, 마이그레이션이 클라우드(310)와 같이 동일한 클라우드 내에 있는 경우, 소스 클러스터(400)와 목적지 클러스터(500) 둘 모두 코디네이터(590)에 설정된 규칙에 기초하여 마이그레이션을 수행할 수 있다. 반면에, 마이그레이션이

클라우드(310) 및 클라우드(320)와 같은 두 개의 다른 클라우드 사이에서 있는 경우, 일부 경우에 코디네이터(590)와 동일한 클라우드에 있는 클러스터만이 코디네이터(590)에 설정된 규칙을 따를 수 있다. 예를 들어, 목적지 클러스터(500)는 클라우드(310) 상에 있을 수 있고 코디네이터(590)에서 설정된 규칙에 기초하여 라이브 마이그레이션을 수행할 수 있다; 소스 클러스터(400)는 자체 관리되거나 다른 클라우드에 의해 관리되는 클라우드(320)에 있을 수 있으며, 코디네이터(590)에서 설정된 규칙을 따르는데 필요한 기능을 갖지 않을 수 있다. 이와 같이, 클라우드(310)는 클라우드(320)로부터 또는 클라우드(320)로의 마이그레이션을 용이하게 하기 위한 추가 기능을 포함할 수 있다.

[0068] 클러스터 제어 평면의 라이브 마이그레이션과 관련하여, 도 6은 동일한 클라우드 내에서 라이브 마이그레이션하는 동안 두 클러스터의 제어 평면 간에 API 호출과 같은 요청을 라우팅하도록 구성된 클러스터 브릿징 애그리게이터의 예를 도시한다. 도 6는 오브젝트가 마이그레이션될 소스 클러스터인 제1 클러스터(400)와 오브젝트가 마이그레이션될 목적지 클러스터인 제2 클러스터(500)를 도시한다. 이 예에서, 소스 클러스터(400)와 목적지 클러스터(500)는 둘 모두 클라우드(310)와 같은 동일한 클라우드에서 호스팅된다. 도 6는 또한 클러스터(400) 및 클러스터(500) 둘 모두 복제된 마스터 노드를 가지는 것으로 도시하며, 따라서 클러스터(400) 및 클러스터(500)은 둘 모두 다중 API 서버(440, 442, 540, 542) 및 대응하는 클러스터 브릿징 애그리게이터(650, 652, 650, 652)와 함께 도시된다.

[0069] 트래픽 볼륨에 따라 다양한 API 서버들 간에 API 호출과 같은 들어오는 요청을 할당하도록 하나 이상의 로드 밸런서를 구성할 수 있다. 예를 들어, 로드 밸런서는 API 서버의 네트워크 주소와 같이 클러스터의 모든 API 서버들과 연관될 수 있다. 그러나, 로드 밸런서는 모든 API 호출을 보내기 위한 단일 네트워크 주소인 클러스터에 의해 실행되는 애플리케이션과 같은 클러스터의 클라이언트(들)를 제공하도록 구성될 수 있다. 예를 들어, 단일 네트워크 주소는 로드 밸런서에 할당된 네트워크 주소일 수 있다. 로드 밸런서가 들어오는 API 호출을 수신하면, 로드 밸런서는 트래픽 볼륨에 기초하여 API 호출을 라우팅할 수 있다. 예를 들어, 로드 밸런서는 클러스터의 API 서버간에 API 호출을 분할하고 API 서버의 네트워크 주소에 기초하여 API 호출을 보낼 수 있다.

[0070] 또한 도시된 바와 같이, 소스 클러스터(400) 및 목적지 클러스터(500)의 애그리게이터는 둘 모두 클러스터 브릿징 애그리게이터(650, 652, 654, 656)로 수정된다. 클러스터 브릿징 애그리게이터(650, 652, 654, 656)는 로드 밸런서(610)로부터 API 호출과 같은 들어오는 요청을 수신하고, 요청을 API 서버(440, 442, 540, 542)로 추가로 라우팅하도록 구성된다. 예를 들어, 클라우드(310)의 제어 평면은 예를 들어 코디네이터(590)를 통해 마이그레이션이 개시될 때 클러스터 브릿징 애그리게이터(650, 652, 654, 656)에 통지할 수 있다. 클러스터 브릿징 애그리게이터(650, 652, 654, 656)가 마이그레이션을 인식하면, 클러스터 브릿징 애그리게이터(650, 652, 654, 656)는 들어오는 API 호출이 소스 클러스터(400) 또는 목적지 클러스터(500)에 의해 처리되어야하는지 여부를 결정할 수 있다. 이 결정에 기초하여, 클러스터 브릿징 애그리게이터(650, 652, 654, 656)는 API 호출을 적절한 API 서버로 라우팅할 수 있다.

[0071] 예를 들어, API 호출이 소스 클러스터(400)의 클러스터 브릿징 애그리게이터(650)에 도착하면, 클러스터 브릿징 애그리게이터(650)는 API 호출이 소스 클러스터(400)의 API 서버 또는 목적지 클러스터(500)의 API 서버에 의해 처리되어야하는지 여부를 결정할 수 있다. 클러스터 브릿징 애그리게이터(650)가 API 호출이 소스 클러스터(400)의 API 서버에 의해 처리되어야한다고 결정하면, 클러스터 브릿징 애그리게이터(650)는 API 호출을 대응하는 API 서버(440)로 라우팅할 수 있다. 그렇지 않으면, 클러스터 브릿징 애그리게이터(650)는 API 호출을 목적지 클러스터(500)의 API 서버로 재-라우팅할 수 있다. 유사하게, API 호출이 목적지 클러스터(500)의 클러스터 브릿징 애그리게이터(654)에 도착하면, 클러스터 브릿징 애그리게이터(654)는 API 호출이 목적지 클러스터(500) 또는 소스 클러스터(400)에 의해 처리되어야하는지 여부를 결정할 수 있다. 클러스터 브릿징 애그리게이터(654)가 API 호출이 목적지 클러스터(500)에 의해 처리될 것이라고 결정하면, 클러스터 브릿징 애그리게이터(654)는 API 호출을 대응하는 API 서버(540)로 라우팅할 수 있다. 그렇지 않으면, 클러스터 브릿징 애그리게이터(654)는 API 호출을 소스 클러스터(400)의 API 서버로 라우팅할 수 있다. 소스 클러스터(400)의 API 서버와 목적지 클러스터(500)의 API 서버는 그들이 처리하는 오브젝트에 대해 서로 다른 스키마를 구현할 수 있기 때문에, API 트래픽 할당의 변경은 목적지 클러스터(500)의 스키마에 부합하는 오브젝트의 부분을 효과적으로 변경할 수 있다.

[0072] 클러스터 브릿징 애그리게이터(650, 652, 654, 656)는 임의의 많은 팩터에 기초하여 API 호출을 라우팅하거나 재-라우팅할 수 있다. 예를 들어, 라우팅은 포드, 서비스 등과 같은 리소스 유형에 기초할 수 있다. 예를 들어, 클러스터 브릿징 애그리게이터(650, 652)는 모든 포드에 대한 API 호출을 소스 클러스터(400)의 API 서버(440, 442)로 라우팅할 수 있고, 모든 서비스에 대한 API 호출을 목적지 클러스터(500)으로 재-라우팅할 수 있

다. 라우팅은 대안적으로 오브젝트 유형에 기초할 수 있다. 예를 들어, 클러스터 브릿징 애그리게이터(650, 652)는 포드 오브젝트에 대한 API 호출의 50%를 소스 클러스터(400)의 API 서버(440, 442)로 라우팅하고 나머지는 목적지 클러스터(500)로 재-라우팅할 수 있다. 또 다른 대안으로, 라우팅은 리소스의 물리적 위치에 기초할 수 있다. 예를 들어, 클러스터 브릿징 애그리게이터(650, 652)는 특정한 데이터센터의 포드에 대한 API 호출의 30%를 라우팅하고 나머지는 목적지 클러스터(500)로 재-라우팅할 수 있다. 다른 예시적 팩터는 사용자 에이전트, 사용자 계정, 사용자 그룹, 요청의 송신자의 위치 등을 포함할 수 있다. API 호출 라우팅을 위한 팩터는 클라우드(310)에 대한 클라우드 제공자에 의해 코디네이터(590)에서 설정될 수 있다.

[0073] 클러스터 브릿징 애그리게이터(650, 652, 654, 656)는 단계적 방식으로 API 호출을 라우팅하거나 재-라우팅할 수 있다. 예를 들어, 클러스터 브릿징 애그리게이터(654, 656)는 하나의 리소스 유형에 대한 API 호출을 한 단계에서 목적지 클러스터(500)의 API 서버(540, 542)로 라우팅하기 시작한 다음, 다른 리소스 유형에 대한 API 호출을 다음 단계에서 목적지 클러스터(500)의 API 서버(540, 542)에 포함하도록 변경할 수 있다. 대안적으로, 클러스터 브릿징 애그리게이터(654, 656)는 예를 들어 하나의 물리적 위치에 대한 API 호출을 한 단계에서 목적지 클러스터(500)의 API 서버(540, 542)로 라우팅하기 시작한 다음, 다른 물리적 위치에 대한 API 호출을 다음 단계에서 목적지 클러스터(500)의 API 서버(540, 542)로 라우팅하는 것을 포함하도록 변경할 수 있다. 다른 예로서, 클러스터 브릿징 애그리게이터(654, 656)는 API 호출을 API 서버(540, 542)로 증가하는 비율로 라우팅할 수 있다. 예를 들어, 포드 오브젝트의 10%에 대한 API 호출을 한 단계에서 목적지 클러스터(500)의 API 서버(540, 542)로 라우팅할 수 있고, 포드 오브젝트의 20%에 대한 API 호출을 다음 단계에서 목적지 클러스터(500)의 API 서버(540, 542)로 라우팅한다. API 호출 라우팅의 단계는 클라우드(310)에 대한 클라우드 제공자에 의해 코디네이터(590)에서 설정될 수 있다.

[0074] 요청을 라우팅할지 또는 재-라우팅할지를 결정하기 위해, 클러스터 브릿징 애그리게이터(650, 652, 654, 656)는 만들어질 할당에 대한 정보를 제공받을 수 있다. 예를 들어, 클러스터 브릿징 애그리게이터(650, 652, 654, 656)는 소스 클러스터(400) 및 목적지 클러스터(500)에 할당될 트래픽의 부분에 대한, 목적지 클러스터(500)의 데이터베이스(570)와 같은 하나 이상의 데이터베이스에 액세스하도록 구성될 수 있다. 이와 같이, API 호출이 예를 들어 클러스터 브릿징 애그리게이터(654)에 도착할 때, 클러스터 브릿징 애그리게이터(654)는 목적지 클러스터(500)에 할당될 API 호출의 부분(0<F<1)에 기초하여 API 호출에 대한 해시 값을 계산할 수 있다. 해시 값은 API 호출 소스의 IP 주소 및 API 호출의 메타데이터와 같은 API 호출의 다른 정보에 기초하여 추가로 계산될 수 있다. 이러한 정보는 위에서 설명한 단계적 롤아웃 프로세스와 관련된 리소스 유형, 오브젝트 유형, 물리적 위치 등을 결정하는데 사용될 수 있다. 일부 예에서, 해시 값은 0과 1 사이의 분수인 숫자 값 p로 해석될 수 있다. p<F이면, 클러스터 브릿징 애그리게이터(654)는 API 호출을 목적지 클러스터(500)로 라우팅할 수 있고, 그렇지 않으면 클러스터 브릿징 애그리게이터(654)는 API 호출을 소스 클러스터(400)로 라우팅할 수 있다. 해시 값에 기초한 결정은 마이그레이션에 관련하여 어떤 클러스터 브릿징 애그리게이터가 API 호출을 수신하더라도 다른 클러스터 브릿징 애그리게이터와 동일한 결정을 내리도록 결정적으로 정의될 수 있다. 따라서 API 호출을 두 번 이상 재-라우팅할 필요가 없다. 일부 경우에, 전술 한 단계적 롤아웃에서의 전환 동안, 상이한 부분 F가 설정될 수 있으며, 예를 들어 상이한 리소스, 상이한 물리적 위치 등이 있다.

[0075] 추가로, 클러스터 브릿징 애그리게이터는 두 클러스터 사이에 다른 리소스를 할당하도록 추가로 구성될 수 있다. 예를 들어, 목적지 클러스터(500)는 소스 클러스터(400)에 의해 사용되는 제어기와 비교하면 제어 루프를 실행하기 위한 상이한 제어기를 사용할 수 있다. 따라서 소스 클러스터의 제어기와 목적지 클러스터의 제어기 사이의 스위칭은 단계적 롤아웃에서도 수행될 수 있다. 예를 들어, 오브젝트에 일관되지 않은 변경이 이루어지지 않도록 제어기는 오브젝트를 조작하기 전에 잠금을 획득할 수 있다. 이와 같이, 클러스터 브릿징 애그리게이터(650, 652, 654, 656)는 소스 클러스터(400)의 제어기와 목적지 클러스터(500)의 제어기 사이에 제어기 잠금을 할당하도록 구성될 수 있다. 할당은 또한 미리 결정된 단계에서 수행될 수 있으며, 이는 또한 조용히 적용될 수 있다.

[0076] 도 6의 API 서버(440, 442, 540, 542) 및 클러스터 브릿징 애그리게이터(650, 652, 654, 656)는 함께 기본적으로 논리적 API 서비스를 형성한다. 따라서 이 논리적 API 서비스의 클라이언트는 이 논리적 API 서비스에 요청을 보낼 수 있으며, 요청은 다양한 클러스터 브릿징 애그리게이터에 의해 라우팅되고 다양한 API 서버에 의해 처리된다. 클라이언트에게는 가능한 대기 시간 외에는 눈에 띄는 차이가 없을 수 있다.

[0077] 그러나, 제1, 소스 클러스터(400) 및 제2, 목적지 클러스터(500)가 다른 클라우드에서 호스팅되는 경우, 소스 클러스터(400) 또는 목적지 클러스터(500) 중 하나에 클러스터 브릿징 애그리게이터가 제공되지 않을 수 있고, 도 7은 두 개의 서로 다른 클라우드 간에 라이브 클러스터 마이그레이션을 수행할 때 클러스터 제어 평면에 대

한 API 호출과 같은 요청을 인터셉트하는 추가 컴포넌트를 도시한다. 도시된 이 예에서, 목적지 클러스터(500)는 라이브 마이그레이션을 수행하도록 구성된 클라우드(310)에 있는 반면, 소스 클러스터(400)는 자체 관리되거나 라이브 마이그레이션을 수행하도록 구성되지 않은 다른 클라우드 제공자에 의해 관리되는 클라우드(320)에 있다. 이와 같이, 클라우드(310) 상의 목적지 클러스터(500)는 전술한 바와 같이 클러스터 브릿징 애그리게이터(654, 656)와 함께 제공되는 반면, 클라우드(320) 상의 소스 클러스터(400)에는 클러스터 사이에서 API 호출을 라우팅 및 재-라우팅할 수 없는 애그리게이터(450, 452)가 제공된다.

[0078] 여기서 두 개의 클러스터가 서로 다른 클라우드에 있기 때문에, API 호출과 같은 요청은 도 6에 도시된 바와 같이 동일한 로드 밸런서(610)를 통해 수신되지 않을 것이다. 오히려, API 호출은 IP 주소와 같은 서로 다른 네트워크 주소에 기초하여 소스 클러스터(400) 및 목적지 클러스터(500)의 클러스터 브릿징 애그리게이터로 라우팅될 것이다.

[0079] 또한 도 7에 도시된 바와 같이, 클러스터(400)는 클러스터 브릿징 애그리게이터를 포함하지 않기 때문에, 사이드카 컨테이너는 클러스터(400)에서 로컬적으로 API 서버로 향하는 API 호출과 같은 요청을 인터셉트하기 위해 클라우드(320)의 포드에 주입되고, 이들을 목적지 클러스터(500)의 클러스터 브릿징 애그리게이터(654, 656)로 재-라우팅한다. 예를 들어, 사이드카 컨테이너는 사용자가 클라우드(320)의 클라우드 제어 평면에 설치하는 확장에 의해 주입될 수 있다. 사이드카 컨테이너는 소스 클러스터(400)에서 실행되는 모든 워크로드 포드에 주입될 수 있다. 예를 들어 도시된 바와 같이, 사이드카 컨테이너(720)는 클러스터(400)에서 포드(710)에 주입된다. 사이드카 컨테이너(720)는 API 서버(440 또는 442)로 향하는 포드(710)에서 실행되는 워크로드(730)로부터의 API 호출을 가로 채고, 소스 클러스터(400)에 없는 클러스터 브릿징 애그리게이터를 시뮬레이션하도록 구성될 수 있다. 이러한 API 호출을 목적지 클러스터(500)의 클러스터 브릿징 애그리게이터(654, 656)로 리디렉션함으로써 이 시뮬레이션을 수행한다. 클러스터 브릿징 애그리게이터(654, 656)는 이러한 API 호출이 API 서버(540, 542)에 의해 로컬적으로 처리되어야하는지 또는 소스 클러스터의 API 서버(440, 442)로 다시 전송되어야하는지 여부를 결정할 수 있다. 클러스터 브릿징 애그리게이터(654, 656)는 도 6과 관련하여 위에서 논의된 바와 같이 결정을 내리고 그에 따라 API 호출을 라우팅할 수 있다.

[0080] 도 7의 API 서버(440, 442, 540, 542), 애그리게이터(450, 452), 사이드카 컨테이너(712), 클러스터 브릿징 애그리게이터(654, 656))는 함께 기본적으로 논리적 API 서비스를 형성한다. 따라서 이 논리적 API 서비스의 클라이언트는 이 논리적 API 서비스에 요청을 보낼 수 있으며, 요청은 사이드카 컨테이너(720)에 의해 인터셉트되고 및/또는 다양한 클러스터 브릿징 애그리게이터에 의해 라우팅되고, 다양한 API 서버에 의해 처리될 수 있다. 클라이언트에게는 가능한 대기 시간 외에는 눈에 띄는 차이가 없을 수 있다.

[0081] 전술한 바와 같이 사이드카 컨테이너를 주입하는 대안으로서, 요청을 인터셉트하고 재-라우팅하기 위해 다른 컴포넌트 또는 프로세스가 사용될 수 있다. 예를 들어, DNS(도메인 이름 서비스) 엔트리는 목적지 클러스터의 클러스터 브릿징 애그리게이터로 재-라우팅하기 위해 노드에 주입될 수 있다.

[0082] 도 5로 돌아가서, 클러스터 제어 평면에 대한 스토리지와 관련하여, 소스 클러스터(400) 및 목적지 클러스터(500)가 동일한 클라우드에 있고, 동일한 데이터센터 내에 있는 경우, 데이터베이스(570)는 데이터베이스(470)와 동일한 쿼럼에 조인할 수 있다. 이와 같이, 데이터베이스(470) 또는 데이터베이스(570)를 포함하는 데이터베이스의 쿼럼은 오브젝트가 데이터베이스의 쿼럼 중 어느 하나에 수정되거나 기록되기 전에 합의에 도달해야 한다. 예를 들어, 대부분의 데이터베이스 복제본이 변경에 동의하면 합의에 도달할 수 있다. 이렇게하면 데이터베이스(570) 및 데이터베이스(470)과 해당 복제본이 일관된 변경을 반영하는 것을 보장한다. 일부 예에서, 데이터베이스(570)는 처음에 데이터베이스 쿼럼의 비-투표 멤버로서 조인할 수 있고, 나중에 쿼럼의 투표 멤버가 될 수 있다.

[0083] 그러나, 소스 클러스터(400)와 목적지 클러스터(500)가 동일한 클라우드 또는 동일한 데이터센터에 있지 않으면, 데이터베이스(570)는 데이터베이스(470)의 쿼럼에 조인할 수 없다. 이와 같이, 도 8은 다른 클라우드 및/또는 영역의 클러스터에 대한 라이브 마이그레이션 동안 클러스터 제어 평면 스토리지 동기화의 예를 도시한다. 예를 들어, 제1 소스 클러스터(400)는 클라우드(320)에 있을 수 있고, 제2, 목적지 클러스터(500)는 클라우드(310)에 있을 수 있다. 다른 예로서, 목적지 클러스터(500)는 데이터센터(260)에 있을 수 있고, 소스 클러스터(400)는 데이터센터(270)에 있을 수 있다.

[0084] 컨테이너화된 환경에서 오브젝트의 일부 필드는 API 서버에 의해서만 수정될 수 있으며 그렇지 않으면 변경할 수 없다. 따라서, API 서버(440 또는 442)와 같은 소스 클러스터(400)의 API 서버에 의해 오브젝트의 변경 불가능한 필드가 작성되거나 수정되면, API 서버(540 또는 542)와 같은 목적지 클러스터(500)의 API 서버는 소스 클

러스터(400)의 데이터베이스(470)에 저장된 이러한 필드를 수정하지 못할 수 있다. 따라서 도시된 바와 같이, 예를 들어 클러스터 브릿징 애그리게이터(654)에서 새로운 오브젝트가 생성되거나 변경 불가능한 필드를 수정하도록 요청하는 API 호출이 들어오면, API 호출은 클러스터 브릿징 애그리게이터(654)에 의해 수정되고, 우선 애그리게이터(450)와 같은 소스 클러스터(400)에 송신될 수 있다. API 서버(440)는 수정된 API 호출에 따라 데이터베이스(470)에 저장된 오브젝트(810)를 생성 또는 수정할 수 있다.

[0085] 클러스터 브릿징 애그리게이터(654)는 그 다음 자신의 로컬 API 서버(540)를 사용하여 데이터베이스(570)에 오브젝트(820)로 도시된, 데이터베이스(470)에 오브젝트(810)의 자체 복사본을 생성할 수 있다. 예를 들어, 클러스터 브릿징 애그리게이터(654)는 소스 클러스터(400)의 API 서버(440)에 의해 선택된 값을 갖는 변경 불가능한 필드를 읽을 수 있고, 이들 값을 오브젝트(820)에 기록할 수 있다.

[0086] 일부 예에서, 클러스터 브릿징 애그리게이터(654, 656)는 API 호출자가 세계에 대한 일관된 뷰를 볼 수 있도록 해당 오브젝트에 대한 쓰기 동작이 진행중인 동안 오브젝트에 대한 읽기 전용 동작을 차단할 수 있다. 그렇지 않으면, 위에서 설명한 것처럼 이 마이그레이션 환경에서 쓰기는 다단계 프로세스일 수 있기 때문에 API 호출자가 수행된 변경의 일부만 관찰할 수 있다. 또한 API 호출자는 프로세스가 이러한 호출자에게 투명하도록 유지되어야 하는 API 서버의 동시성 모델에 대한 기대를 가지고 있다.

[0087] 다른 양태에서, 클러스터에서 실행되는 워크로드에 대해 마이그레이션을 수행할 수 있다. 도 9는 워크로드 마이그레이션 수행과 관련된 예시적 구성을 도시한다. 예를 들어, 노드(910, 912, 914)를 포함하는 노드 풀(429)과 함께 제1, 소스 클러스터(400)가 도시된다. 하나 이상의 포드가 도시된 포드(920) 및 포드(922)와 같은 클러스터(400)의 노드에서 실행될 수 있다. 클러스터(400)는 클러스터(400)의 워크로드에 트래픽을 할당하기 위한 로컬 로드 밸런서(930)를 더 포함할 수 있다. 예를 들어, 워크로드에 의해 서비스되는 웹사이트 또는 애플리케이션으로부터의 요청은 로컬 로드 밸런서(930)에 의해 수신될 수 있고, 로컬 로드 밸런서(930)는 이러한 요청을 노드 풀(429)의 다양한 포드 및 노드에 할당할 수 있다. 예를 들어, 클러스터(400)의 워크로드에 의해 서비스되는 웹사이트 또는 애플리케이션은 웹사이트 또는 애플리케이션을 로컬 로드 밸런서(930)의 네트워크 주소에 연관시키는 도메인 이름 서비스(DNS) 레코드로 구성될 수 있다.

[0088] 또한 도시된 바와 같이, 클러스터(400) 내의 워크로드는 제2, 목적지 클러스터(500)로 마이그레이션된다. 클러스터(500)는 어떤 노드도 갖지 않는 노드 풀(940)과, 클러스터(500)에서 포드 및 노드가 생성되면 들어오는 요청을 워크로드에 할당하기 위한 로컬 밸런서(970)로 초기화될 수 있다. 동일한 데이터센터 내 또는 데이터센터의 동일한 영역/구역 내와 같은 동일한 위치 내에서 클러스터(400)로부터 클러스터(500)로의 마이그레이션이 수행될 수 있고 또는 상이한 위치 사이에서 수행될 수 있다. 마이그레이션은 동일한 클라우드 내에서 또는 상이한 클라우드 간에 수행될 수 있다. 클러스터(400 및 500)는 단 하나의 노드 풀로 도시되었지만, 실제 예에서 클러스터(400 및 500)는 복수의 노드 풀을 포함할 수 있다. 클러스터가 노드를 아직 노드 풀로 그룹화하지 않은 경우, 마이그레이션 중에 각 노드가 자체 노드 풀로 처리되거나, 비슷한 크기의 노드가 함께 그룹화될 수 있다.

[0089] 목적지 클러스터(500)가 초기화되면, 노드 풀(940)의 크기가 점진적으로 증가할 수 있다. 예를 들어, 새 노드(950)가 노드 풀(940)에 할당될 수 있다. 새 노드(950)는 초기에 어떤 포드도 포함하지 않을 수 있다. 노드 풀(940)의 크기 증가에 응답하여, 이전 노드 풀(429)의 크기가 감소할 수 있다. 예를 들어, 이전 노드(910)가 삭제될 수 있다. 새 노드의 할당 및 이전 노드 제거는 코디네이터의 지시에 따라 클라우드 제공자가 수행할 수 있다.

[0090] 소스 클러스터(400) 및/또는 목적지 클러스터(500)의 클러스터 제어 평면은 노드(910)가 현재 누락되었음을 통지받을 수 있고, 노드(910)에 이전에 존재했던 모든 포드, 예를 들어 도시된 포드(920 및 922)를 분실된 것으로 등록할 수 있다. 이와 같이, 목적지 클러스터(500)의 클러스터 제어 평면은 새 노드 풀(940)에서 대체 포드를 생성할 수 있다. 예를 들어, 목적지 클러스터(500)의 제어기는 노드 풀(940)의 새 노드(950)가 용량을 가지고 있다고 결정할 수 있고, 새 노드(950)에서 도시된 교체 포드(960 및 962)와 같은 교체 포드를 생성할 수 있다. 따라서, 효과적으로 포드(920, 922)는 포드(960, 962)로서 제2 클러스터로 이동된다. 이것은 노드 풀(429)의 다른 노드에 대해 반복될 수 있는데, 예를 들어, 노드 풀(429)에 노드 및/또는 포드가 더 이상 존재하지 않을 때까지, 노드 풀(940)에서 도시된 노드(912, 914)에 대응하는 새로운 노드(952 및 954)를 생성하고, 누락된 포드를 교체할 수 있다.

[0091] 임의의 포드를 이동하기 전에 노드(910)를 삭제하고 노드(950)를 추가하는 것에 대한 대안으로서, 라이브 마이그레이션이 수행될 수 있다. 예를 들어, 새 노드(950)가 생성되면, 노드(910)는 새 포드가 노드(910)에서 스케줄링되는 것을 방지하도록 "연결"될 수 있다. 그 다음 노드(950)에서 새 포드(960)이 생성된다. 포드(920)의 상

태는 기록되어 포드(960)로 전송될 수 있다. 그 후, 포드(920)에서 프로세스의 실행이 일시 중지될 수 있다. 상태를 기록한 이후 포드(920)에 변경이 있는 경우, 이러한 변경은 포드(960)에도 복사될 수 있다. 일시 중지된 실행은 포드(960)에서 재개될 수 있다. 그 다음 포드(920)가 삭제될 수 있다. 이 라이브 마이그레이션 동안, 워크로드에 대한 요청과 같이 포드(920)로 향하는 트래픽은 포드(920)가 삭제될 때까지 포드(960)로 포워딩될 수 있다. 예를 들어, 로드 밸런서는 새로 생성된 포드(960)를 인식하기 전에 포드(920)로 요청을 보낼 수 있다. 이것은 포드가 남아 있지 않을 때까지 소스 클러스터(400)의 다양한 노드 및 노드 풀의 각 포드에 대해 반복될 수 있다.

[0092] 또한, 워크로드의 마이그레이션은 포드(Pod)의 마이그레이션 외에도, 포드가 속한 서비스의 마이그레이션을 포함할 수 있다. 서비스 마이그레이션은 포드 마이그레이션과 오버랩핑될 수 있다. 예를 들어, 목적지 클러스터(500)에서 하나 이상의 포드가 생성되면, 소스 클러스터(400)의 포드에 의해 이전에 처리된 서비스는 목적지 클러스터(500)의 포드에 의해 처리되도록 마이그레이션될 수 있다. 또한, 소스 클러스터(400)에 더 이상 서비스를 처리할 포드가 존재하지 않기 전에 서비스 마이그레이션이 완료되어야할 수 있다.

[0093] 이와 관련하여, 하나 이상의 글로벌 로드 밸런서가 생성될 수 있다. 예를 들어, 워크로드 노드 및 포드 마이그레이션이 시작되었지만 노드가 이동되기 전에, 소스 클러스터(400) 및 목적지 클러스터(500)는 소스 클러스터(400) 및 목적지 클러스터(500)에서 실행중인 워크로드로 요청을 라우팅하도록 구성된 하나 이상의 로드 밸런서와 각각 연관될 수 있다. 예를 들어, 도시된 바와 같이, 로컬 로드 밸런서(930) 및 로컬 로드 밸런서(970) 모두는 글로벌 로드 밸런서(980)와 연관될 수 있다. 따라서, 소스 클러스터(400) 및 목적지 클러스터(500)가 상이한 위치 또는 클라우드에 있는 경우, 글로벌 로드 밸런서(980)는 요청을 이러한 상이한 위치 또는 클라우드로 라우팅하도록 구성될 수 있다. 클러스터(400)의 워크로드에 의해 이전에 서비스된 웹사이트 또는 애플리케이션은 이전에 로컬 로드 밸런서(930) 대신에 글로벌 로드 밸런서(980)의 네트워크 주소에 웹사이트 또는 애플리케이션을 연관시키는 DNS 레코드로 구성될 수 있다. 이와 같이, 워크로드 노드 및 포드 마이그레이션이 시작되면, 웹사이트 또는 애플리케이션으로부터의 요청은 글로벌 로드 밸런서(980)를 통해 로컬 로드 밸런서(930 및 970) 모두로 라우팅될 수 있다.

[0094] 워크로드 노드 및 포드 마이그레이션이 완료되면, 로컬 로드 밸런서(970)와 글로벌 로드 밸런서(980) 간의 연관이 제거될 수 있다. 또한, 클러스터(400) 및 클러스터(500) 모두에 의해 이전에 서비스된 웹사이트 또는 애플리케이션은 웹사이트 또는 애플리케이션을 로컬 로드 밸런서(970)의 네트워크 주소에 연관시키는 DNS 레코드로 구성될 수 있다. 따라서, 이 시점부터, 로컬 로드 밸런서(970)는 웹사이트 또는 애플리케이션으로부터의 요청을 목적지 클러스터(500)에서 실행되는 워크로드로만 라우팅하도록 구성될 수 있다.

[0095] 또한 도 9에 도시된 워크로드 마이그레이션이 서로 다른 위치 또는 서로 다른 클라우드 간에 이루어지는 경우, 워크로드 스토리지의 라이브 마이그레이션을 수행해야할 수 있다. 도 10은 상이한 위치 또는 클라우드 간의 라이브 워크로드 스토리지 마이그레이션을 도시한다. 예를 들어, 라이브 워크로드 스토리지 마이그레이션은 도 9에 도시된 대로 포드의 마이그레이션과 동시에 발생할 수 있다. 컨테이너화된 환경을 위한 스토리지 시스템은 데이터를 저장하는 다양한 오브젝트를 포함할 수 있다. 예를 들어, 스토리지 시스템은 클라우드 제공자가 제공하는 영구 디스크와 참조를 포함하는 메타데이터 오브젝트를 포함할 수 있다. 예를 들어, 메타데이터 오브젝트를 사용하여 포드 또는 컨테이너용 영구 디스크를 설정하거나 '마운트'할 수 있다. 일부 예로서, 메타데이터 오브젝트는 영구 디스크의 데이터를 참조하는 영구 볼륨 및 영구 볼륨을 참조하고 컨테이너 또는 포드의 이러한 데이터 사용에 대한 정보를 저장하는 영구 볼륨 클레임을 포함할 수 있다.

[0096] 상이한 위치 또는 클라우드 간에 마이그레이션하는 경우, 메타데이터 오브젝트는 목적지 환경에 복사될 수 있지만, 영구 디스크는 목적지 환경에 복사되지 않을 수 있다. 따라서 워크로드에 대한 스토리지 시스템의 라이브 마이그레이션은 각 영구 디스크의 위치를 추적하고, 목적지 환경에서 메타데이터 오브젝트를 복제하고, COW(copy-on-write) 시스템을 사용하여 데이터를 복사하여 수행할 수 있다.

[0097] 예를 들어, 도시된 바와 같이, 제1, 소스 클러스터(400)에서 실행되는 동안 포드(920)는 영구 디스크(1012)를 참조할 수 있는 이미 존재하는 메타데이터 오브젝트(1010)를 가질 수 있다. 이러한 저장 오브젝트의 효과적인 사본을 만들기 위해, 헬퍼 포드(1030)가 소스 클러스터(400)에서 생성되고, 메타데이터 오브젝트(1010)에 연결될 수 있다. 이 헬퍼 포드(1030)는 포드(920)가 포드(960)로서 제2, 목적지 클러스터(500)로 마이그레이션된 후에 영구 디스크(1012)로부터 읽도록 구성될 수 있다.

[0098] 마이그레이션된 포드(960)는 목적지 클러스터(500)의 노드 및 새로 생성된 메타데이터 오브젝트(1020)에 연결되며, 이는 메타데이터 오브젝트(1010)의 복제본일 수 있다. 마이그레이션된 포드(960)의 메타데이터 오브젝트

(1020)가 영구 디스크(1012)에 대한 참조를 포함하는 것으로 결정될 수 있다. 마이그레이션된 포드(960)에 대한 스토리지를 설정하기 위해, 스토리지 드라이버(1050)는 영구 디스크(1012)가 상이한 클러스터에 있다고 결정할 수 있다. 이와 같이, 새 영구 디스크(1022)가 목적지 클러스터(500)에 생성될 수 있다.

[0099] 그러나, 새 영구 디스크(1022)에 직접 연결되는 대신, 포드(960)는 초기에 스토리지 드라이버(1050)를 통해 읽기 및/또는 쓰기를 수행할 수 있으며, 이는 포드(960) 및 메타데이터 오브젝트(1020)가 2개의 상이한 위치의 영구 디스크를 참조하고 있다고 결정할 수 있다. 예를 들어, 스토리지 드라이버(1050)는 도 9의 노드(910)에서 플러그인으로 실행될 수 있다. 스토리지 드라이버(1050)는 예를 들어 헬퍼 포드(1030)에 대한 네트워크 액세스를 통해 이전 영구 디스크(1012) 및 새 영구 디스크(1022) 모두에 액세스하도록 구성될 수 있다.

[0100] 예를 들어, 읽기 위해, 포드(960)는 새 영구 디스크(1022)로부터 읽기 위해 스토리지 드라이버(1050)를 사용할 수 있다. 추가적으로, 스토리지 드라이버(1050)는 또한 영구 디스크(1012)로부터 읽을 수 있는 헬퍼 포드(1030)를 호출할 수 있다.

[0101] 쓰기를 위해, 포드(960)는 또한 스토리지 드라이버(1050)를 통해 그렇게 할 수 있다. 스토리지 드라이버(1050)는 모든 기록을 영구 디스크(1022)로 보내도록 구성될 수 있다. 이렇게하면, 새로운 변경이 새 영구 디스크(1022)에 기록된다. 쓰기는 COW(copy-on-write)에 의해 수행될 수 있으며, 여기서 변경은 새 영구 디스크(1022)에 직접 기록되는 반면, 변경되지 않은 데이터는 이전 영구 디스크(1012)에서 복사된다.

[0102] 또한, 소스 클러스터(400)의 스토리지 오브젝트로부터 목적지 클러스터(500)로 모든 데이터를 점진적으로 이동시키기 위해 백그라운드에서 마이그레이션이 수행될 수 있다. 예를 들어, 네트워크가 사용 중이 아닐 때, 스토리지 드라이버(1050)는 영구 디스크(1012)로부터 데이터를 계속 읽을 수 있고, 이 데이터를 영구 디스크(1022)에 기록할 수 있다. 모든 데이터가 복사되면, 영구 디스크(1022)는 완전한 파일 시스템을 포함할 것이고, 포드(960)는 스토리지 드라이버(1050) 없이 영구 디스크(1022)에 직접 연결될 수 있다. 기존 영구 디스크(1012)는 삭제될 수 있다. 이 프로세스 동안, 포드(960)의 관점에서 볼 때, 가능한 지연 시간 외에 다른 차이는 없다.

[0103] 도 10은 포드와 영구 디스크 사이의 하나의 메타데이터 오브젝트를 도시하지만, 일부 예에서 참조 체인을 형성하는 서로를 참조하는 다수의 메타데이터 오브젝트가 있을 수 있다. 예를 들어, 포드는 영구 볼륨을 참조하는 영구 볼륨 클레임을 참조할 수 있으며, 영구 볼륨은 영구 디스크를 참조할 수 있다.

[0105] 예시적 방법들

[0106] 상기 설명된 예시적 시스템에 더하여, 예시적 방법이 이제 설명된다. 이러한 방법은 전술한 시스템, 그 수정 또는 상이한 구성을 갖는 임의의 다양한 시스템을 사용하여 수행될 수 있다. 다음 방법에 관련된 동작은 설명된 정확한 순서로 수행될 필요가 없음을 이해해야 한다. 오히려 다양한 동작이 다른 순서로 또는 동시에 처리될 수 있으며, 동작이 추가되거나 생략될 수 있다.

[0107] 예를 들어, 도 11a-c는 클러스터 제어 평면에 대한 라이브 클러스터 마이그레이션의 예를 보여주는 타이밍 다이어그램이다. 도 11a-c는 제1, 소스 클러스터의 소스 마스터 노드(1111), 제2, 목적지 클러스터의 목적지 마스터 노드(1112), 논리적 API 서비스(1113) 및 코디네이터(1114)에서 발생하는 다양한 액션을 도시한다. 소스 마스터 노드(1111) 및 목적지 마스터 노드(1112)는 도 4-7 중 임의의 것에 도시된 바와 같이 구성될 수 있다. 단지 하나의 소스 마스터 노드(1111)와 하나의 목적지 마스터 노드(1112)만이 도시되어 있지만, 도 4-7에 도시된 바와 같이, 소스 클러스터와 목적지 클러스터 중 하나 또는 둘 모두에 임의의 수의 마스터 노드가 있을 수 있다. 논리적 API 서비스(1113)는 하나 이상의 클러스터에 대한 API 서버의 쿼럼일 수 있으며, 여기에는 도 4-6에 도시된 애그리게이터 및/또는 클러스터 브릿징 애그리게이터 및/또는 도 7에 도시된 사이드카 컨테이너가 포함된다. 타이밍 다이어그램은 도 2 또는 도 3에 도시된 하나 이상의 프로세서와 같은 시스템에서 수행될 수 있다.

[0108] 도 11a를 참조하면, 처음에, 소스 클러스터의 소스 마스터 노드(1111)가 이미 클라우드에서 실행 중일 수 있다. 이와 같이, 소스 마스터 노드(1111)는 이미 PD에 연결되어 있고, 소스 마스터 노드(1111)의 API 서버(들)는 이미 논리적 API 서비스(1113)의 멤버(들)일 수 있다.

[0109] 어떤 시점에서, 클라우드의 클라우드 제공자 또는 사용자는 소프트웨어 업그레이드 도입, 다른 데이터센터로 이동, 다른 클라우드로/로의 이동 등과 같은 환경 변경을 개시할 수 있다. 클라우드 제공자는 코디네이터(1114)에서 환경 변경을 구현하기 위한 라이브 마이그레이션에 대한 규칙을 추가로 정의할 수 있고, 코디네이터(1114)는 규칙을 구현하도록 논리적 API 서비스(1113)에 지시할 수 있다. 예를 들어, 규칙은 워크로드 트래픽할당 및 마이그레이션 단계에 대한 팩터를 포함할 수 있다.

[0110] 환경 변경이 개시되면, 목적지 마스터 노드(1112)가 생성되고 PD에 연결될 수 있다. 소스 마스터 노드(1111)로서 일관된 변경을 유지하기 위해, 목적지 마스터 노드(1112)의 하나 이상의 데이터베이스가 소스 마스터 노드(1111)의 하나 이상의 데이터베이스(들)와 브릿지되거나 동기화될 수 있다. 예를 들어, 소스 마스터 노드(1111)와 목적지 마스터 노드(1112)가 동일한 클라우드 및 위치에 있는 경우, 목적지 마스터 노드(1112)의 데이터베이스(들)는 소스 마스터 노드(1111)의 데이터베이스(들)와 동일한 쿼럼에 조인할 수 있다. 소스 마스터 노드(1111)와 목적지 마스터 노드(1112)가 다른 클라우드 또는 위치에 있는 경우, 목적지 마스터 노드(1112)의 데이터베이스(들)는 도 8에 도시된 바와 같이 소스 마스터 노드(1111)의 데이터베이스(들)에 동기화될 수 있다.

[0111] 이 시점에서 목적지 마스터 노드(1112)는 실행을 시작할 수 있고, 소스 마스터 노드(1111)는 계속 실행된다. 따라서, 도 1a 및 1b에 도시된 프로세스와 비교하여 다운 타임이 감소되거나 제거된다. API 호출과 같은 클러스터 제어 평면에 대한 요청을 동시에 처리하기 위해, 목적지 마스터 노드(1112)의 API 서버(들)는 논리적 API 서비스(1113)에 조인할 수 있다. 예를 들어, 목적지 마스터 노드(1112)의 API 서버(들)는 도 6에 도시된 바와 같이 클러스터 브릿징 애그리게이터(들)를 통해 논리적 API 서비스(1113)에 조인할 수 있거나, 도 7에 도시된 바와 같이 사이드카 포드(들)가 생성될 수 있다.

[0112] 코디네이터(1114)가 목적지 마스터 노드(1112)의 API 서버(들)를 관찰하면, 코디네이터(1114)는 환경을 변경하기 위해 단계적 롤아웃을 시작할 수 있다. 도 11b를 계속하면, 타이밍 다이어그램은 소스 클러스터로부터 목적지 클러스터로의 API 트래픽의 단계적 롤아웃 예를 도시한다. 도시된 바와 같이, 코디네이터(1114)는 소스 마스터 노드(1111)의 API 서버(들)와 목적지 마스터 노드(1112)의 API 서버(들) 사이에 단계적 트래픽 할당을 구현하도록 논리적 API 서비스(1113)에 지시할 수 있다. API 트래픽 할당은 도 6에 도시된 바와 같이 클러스터 브릿징 애그리게이터(들) 및/또는 도 7에 도시된 바와 같이 하나 이상의 사이드카 컨테이너를 사용하여 구현될 수 있다. 소스 클러스터와 목적지 클러스터의 API 서버는 서로 다른 스키마를 기반으로 오브젝트를 처리할 수 있으므로, API 트래픽이 목적지 마스터 노드(1112)의 API 서버로 점점 더 많이 라우팅됨에 따라 목적지 환경의 오브젝트에 대한 목적지 스키마가 점진적으로 롤아웃된다.

[0113] 도 11b에 도시된 바와 같이, 롤아웃 단계 동안, 들어오는 API 호출은 논리적 API 서비스(1113)를 통해 목적지 마스터 노드(1112)의 API 서버(들) 및 소스 마스터 노드(1111)의 API 서버(들)로 라우팅될 수 있다. 코디네이터(1114)는 API 트래픽 할당의 미리 결정된 비율을 설정할 수 있다. 표시된 특정 예에서, 수신된 API 호출의 초기 1%는 목적지 마스터 노드(1112)의 API 서버(들)에 의해 처리될 수 있으며, 수신된 API 호출의 나머지 99%는 소스 마스터 노드 (1111)의 API 서버(들)에 의해 처리될 수 있다. 즉, 초기적으로 목적지 환경의 스키마에 따라 목적지 마스터 노드(1112)의 API 서버가 API 호출의 1%만 처리하고, 소스 환경의 스키마에 따라 나머지는 소스 마스터 노드(1111)의 API 서버에서 처리한다. API 트래픽을 미리 결정된 비율로 할당하는 것 외에 또는 그 대안으로, API 트래픽은 리소스 유형, 사용자, 네임스페이스, 오브젝트 유형 등과 같은 다른 기준에 따라 추가로 할당될 수 있다.

[0114] 롤아웃 프로세스 동안, 목적지 마스터 노드(1112)의 API 서버(들)에서의 활동이 모니터링될 수 있다. 예를 들어, 코디네이터(1114)는 API 서버, 제어기 관리자 등과 같은 클러스터 제어 평면 컴포넌트의 활동을 모니터링할 수 있다. 코디네이터(1114)는 문제가 있는 차이에 대해 소스 및 목적지 클러스터에 의해 처리되는 워크로드를 비교하는 것과 같이 워크로드를 추가로 모니터링할 수 있다. 이와 같이, 목적지 마스터 노드(1112)의 API 서버(들)에 의해 처리된 API 호출의 한 비율로 실패가 검출되지 않거나 적어도 마이그레이션 전에 소스 클러스터(400)에서 이미 발생하지 않은 추가 실패가 없는 경우, 그 다음 목적지 마스터 노드(1112)의 API 서버(들)로의 API 트래픽은 더 높은 비율로 증가될 수 있다. 예를 들어, 도시된 바와 같이, 목적지 마스터 노드(1112)의 API 서버(들)로 라우팅되는 API 호출은 1%에서 2%, 5%, 10% 등으로 증가할 수 있다. 그러나, 목적지 마스터 노드(1112)의 API 서버(들)에 의해 처리되는 API 호출의 비율에서 하나 이상의 실패가 검출되면, 실패는 더 많은 비율의 API 호출이 목적지의 API 서버(들)에 의해 처리되는 경우 더 많은 실패가 발생할 수 있다는 경고로 작용할 수 있다. 도 11에 표시된 것처럼 모든 API 트래픽을 소스 API 서버로 되돌리는 등 경고에 기초하여 적절한 조치가 취해질 수 있다.

[0115] 또한 도시된 바와 같이, 일부 인스턴스에서, 오브젝트에 뒤따르는 정확한 스키마와 같은 목적지 환경에 대한 정보를 포함하는 발견 문서는 목적지 마스터 노드(1112)의 API 서버가 들어오는 모든 API 호출을 처리한 후에만 사용자에게 제공될 수 있다. 예를 들어, 각 유형의 오브젝트가 목적지 클러스터에 의해 완전히 처리되면, 대응하는 오브젝트 유형에 대한 발견 문서의 섹션이 해당 유형의 오브젝트에 대한 목적지 스키마로 업데이트될 수 있다. 즉, 최종 사용자는 모든 오브젝트가 목적지 스키마에 기초하여 목적지 마스터 노드(1112)의 API 서버(들)에 의해 처리될 때까지 어떠한 환경 변경도 관찰할 수 없을 수 있다. 이 시점에서, 소스 마스터 노드(111

1)에 의해 수신된 API 트래픽이 더 이상 없으므로, 이전 스키마에 기초하여 소스 마스터 노드(1111)의 API 서버(들)에 의해 처리되는 오브젝트가 없다. 소스 마스터 노드(1111)의 제어 평면도 새 발견 문서를 관찰할 수 있으며, 스키마 마이그레이션이 완료되었다고 통지된다.

[0116] 코디네이터(1114)가 완료된 스키마 마이그레이션을 관찰하면, 코디네이터(1114)는 클러스터의 하나 이상의 다른 양태에 대한 단계적 롤아웃을 선택적으로 시작할 수 있다. 예를 들어, 그림 11c로 계속해서, 타이밍 다이어그램은 제어기에 대한 예시적 단계적 롤아웃을 도시한다. 일부 경우에, 환경 변경은 클러스터의 오브젝트를 작동시키는 제어기의 변경을 포함할 수 있다. 예를 들어, 목적지 환경에서 목적지 마스터 노드(1112)는 소스 마스터 노드(1111)에 의해 사용되는 제어기와 비교하여 제어 루프를 실행하기 위해 상이한 제어기를 사용할 수 있다.

이와 같이, 소스 마스터 노드(1111)의 제어기와 목적지 마스터 노드의 제어기 사이의 스위칭은 단계적 롤아웃에서도 수행될 수 있다. 예를 들어, 오브젝트에 일관되지 않은 변경이 이루어지지 않도록 제어기는 오브젝트를 조작하기 전에 잠금을 획득할 수 있다. 이와 같이, 코디네이터(1114)는 소스 클러스터의 제어기와 목적지 클러스터의 제어기 사이에 단계적 제어기 잠금 할당을 구현하도록 논리 API 서비스(1113)에 지시할 수 있다.

[0117] 따라서, 도 11c에 도시된 특정 예에서, 초기적으로 제어기 잠금의 1%만이 목적지 마스터 노드(1112)의 제어기에 제공되고, 나머지 제어기 잠금은 소스 마스터 노드(1111)의 제어기에 제공된다. API 서버의 롤아웃과 마찬가지로, 코디네이터(1114)는 목적지 마스터 노드(1112)의 제어기로의 스위칭으로 인한 실패에 대해 API 서버, 제어기 관리자 및/또는 워크로드와 같은 클러스터 제어 평면 컴포넌트의 활동을 모니터링할 수 있다. 실패가 검출되지 않거나, 적어도 마이그레이션 이전에 소스 클러스터(400)에서 이미 발생하지 않은 추가 실패가 없는 경우, 목적지 마스터 노드(1112)의 제어기에 제공되는 제어기 잠금의 비율은 점차 증가할 수 있다. 또한 1% 잠금에서 2% 잠금 할당으로 이동하는 것과 같이, 제어기 잠금 할당을 조정하는 동안, 두 제어기가 오브젝트를 조작하지 않도록 하기 위해, 제어기는 제어기가 이전 단계에서 이미 제어 중인 오브젝트에 대한 잠금을 유지하도록 구성할 수 있다. 결국, 모든 제어기 잠금이 목적지 마스터 노드(1112)의 제어기에 제공될 수 있으며, 그 시점에서 소스 마스터 노드(1111)에서 더 이상 제어기 활동은 없다.

[0118] 이 시점에서, 선택적으로 코디네이터(1114)는 임의의 다른 나머지 애드온을 스위칭할 수 있다. 예를 들어, 오브젝트는 소스 마스터 노드(1111)의 애드온 컴포넌트 대신, 목적지 마스터 노드(1112)의 애드온 컴포넌트에 의해 처리될 수 있다. 애드온 컴포넌트의 예는 대시 보드, DNS(Domain Name System) 서버 등과 같은 사용자 인터페이스를 포함할 수 있다. 선택적으로 애드온 컴포넌트는 API 서버 및 제어기에 대해 위에서 설명한대로 단계적 롤아웃에서 스위칭될 수 있다.

[0119] 소스 환경으로부터 목적지 환경으로의 롤아웃이 완료되면, 소스 마스터 노드(1111)에 대한 셧다운 프로세스가 시작될 수 있다. 예를 들어, 소스 마스터 노드(1111)와 목적지 마스터 노드(1112) 사이의 임의의 브릿징, 동기화 또는 데이터베이스 마이그레이션이 중지될 수 있다. 또한, PD는 소스 마스터 노드(1111)로부터 분리될 수 있고, 소스 마스터 노드(1111)는 삭제될 수 있다. 소스 마스터 노드(1111)가 파괴되면, 코디네이터(1114)는 성공적으로 완료된 마이그레이션을 클라우드에 고할 수 있다.

[0120] 클러스터 제어 평면의 마이그레이션 외에도, 워크로드에 대한 라이브 마이그레이션이 수행될 수 있다. 도 12는 한 환경에서 다른 환경으로 클러스터의 워크로드에 대한 예시적 라이브 마이그레이션을 도시하는 타이밍 다이어그램이다. 도 12는 제1, 소스 클러스터의 노드에 있는 이전 포드(1201), 제2, 목적지 클러스터의 노드에 생성된 새 포드(1202) 및 두 클러스터의 클러스터 제어 평면(1203)에서 발생하는 다양한 액션을 도시한다. 포드는 도 4 또는 9에 도시된 바와 같이, 작업자 노드에 구성될 수 있으며, 예를 들어 이전 포드(1201)는 소스 클러스터(400)의 노드(910)에 구성될 수 있고, 새 포드(1202)는 클러스터(500)의 노드(950)에 구성될 수 있다. 단 하나의 이전 포드(1201) 및 단 하나의 새 포드(1202)만을 포함하는 예시적 동작이 도시되어 있지만, 이러한 동작은 소스 클러스터 및 목적지 클러스터에 있는 임의의 수의 포드 쌍에 대해 수행될 수 있다. 제어 평면(1203)은 도 4-7에 도시된 것과 같은 목적지 클러스터 및 소스 클러스터 둘 모두의 제어 평면으로부터의 컴포넌트를 포함할 수 있다. 타이밍 다이어그램은 도 2 또는 도 3에 도시된 하나 이상의 프로세서와 같은 시스템에서 수행될 수 있다.

[0121] 도 12를 참조하면, 이전 포드(1201)가 여전히 소스 클러스터의 노드에서 실행되는 동안, 클러스터 제어 평면(1203)은 새 포드(1202)를 스케줄링할 수 있다. 예를 들어, 새로운 포드(1202)는 목적지 클러스터(500)의 제어기에 의해 스케줄링될 수 있다. 클러스터 제어 평면(1203)은 이전 포드(1201)의 상태를 기록한 다음, 이러한 상태를 새로운 포드(1202)로 전송할 수 있다. 클러스터 제어 평면(1203)은 이전 포드(1201)의 실행을 일시 중지할 수 있다. 클러스터 제어 평면(1203)은 이전 포드(1201)의 상태 변화를 복사할 수 있고, 이러한 변화를 새로운

포드(1202)로 전송할 수 있다. 클러스터 제어 평면(1203)은 포드(1202)의 실행을 재개할 수 있다.

[0122] 포드(1202)가 실행을 시작하면, 애플리케이션 또는 이전 포드(1201)로 향하는 웹 사이트로부터의 요청과 같은 네트워크 트래픽은 클러스터 제어 평면(1203)에 의해 새로운 포드(1202)로 포워딩될 수 있다. 예를 들어, 할당은 도 9와 관련하여 설명된 대로 글로벌 로드 밸런서에 의해 수행될 수 있다. 워크로드 마이그레이션이 완료되면, 이전 포드(1201)에 대한 연결이 닫힐 수 있다. 그 후, 이전 포드(1201)가 삭제될 수 있다. 더 나아가, 라이브 워크로드 마이그레이션 중에, 워크로드 스토리지의 라이브 마이그레이션이 도 10에 도시된 대로 수행될 수 있다. 예를 들어, 워크로드 스토리지의 라이브 마이그레이션은 요청을 워크로드로 라이브 마이그레이션하는 동안 수행될 수 있다.

[0123] 위에서 언급했듯이, 목적지 클러스터는 라이브 마이그레이션 도중 및/또는 이후에 실패가 있는지 모니터링할 수 있다. 이와 같이, 도 13은 라이브 마이그레이션의 성공 또는 실패 여부에 따라 취할 수 있는 추가 액션의 예를 도시한다. 도시된 바와 같이, 소스 환경으로부터 목적지 환경으로의 변경은 코디네이터(1114)에게 지시하는 클라우드 플랫폼(1311)에 의해 개시될 수 있다. 클라우드 플랫폼(1311)은 마이그레이션을 위해 하나 이상의 새로운 목적지 VM을 시작하도록 클라우드 제어 평면(1312)에 지시할 수 있다. 코디네이터(1114)가 클라우드 플랫폼(1311)으로 마이그레이션하는 동안 또는 그 이후에 실패를 보고하면, 클라우드 플랫폼(1311)은 코디네이터(1114)에게 마이그레이션을 중지 또는 일시 중지하도록 지시할 수 있다. 또한 검출된 실패에 대한 정보를 포함하는 출력이 생성될 수 있다. 예를 들어 정보가 클라우드 관리자, 사용자 등에게 디스플레이될 수 있다.

[0124] 대안적으로 또는 추가적으로, 클라우드 플랫폼(1311)은 코디네이터(1114)에게 목적지 환경으로부터 소스 환경으로 다시 변경을 시작하도록 지시할 수 있다. 롤백이 완료되면, 클라우드 플랫폼(1311)은 클라우드 제어 평면(1312)에 마이그레이션을 위해 생성된 목적지 VM을 삭제하도록 지시할 수 있다. 오류 보고, 진단 및 수정은 예를 들어 클라우드 플랫폼(1311)의 관리자에 의해 수행될 수 있다. 오류가 수정되면, 클라우드 플랫폼(1311)은 코디네이터(1114)에게 소스 환경으로부터 목적지 환경으로의 변경을 재-개시하도록 지시할 수 있다. 중요한 것은, 마이그레이션이 실패하고 롤백되더라도, 클러스터에서 실행되는 워크로드가 매우 사소한 중단 이상을 경험하지 않는다는 것이다.

[0125] 또한 도시된 바와 같이, 일부 경우에, 코디네이터(1114)는 성공적인 마이그레이션을 보고할 수 있다. 이러한 경우, 소스 VM(들)이 클라우드 플랫폼(1311)과 동일한 클라우드에 있다면, 클라우드 플랫폼(1311)은 클라우드 제어 평면(1312)에 소스 VM(들)을 삭제하도록 지시할 수 있다. 소스 VM(들)이 클라우드 플랫폼(1311)과 다른 클라우드에 있는 경우, 클라우드 플랫폼(1311)은 소스 VM(들)에 대해 아무 것도 할 수 없을 수 있다. 이 경우 사용자는 이러한 소스 VM을 삭제하도록 다른 클라우드에 지시해야 할 수 있다.

[0126] 도 13에 여러 가지 액션의 예가 도시되었지만, 모든 액션이 수행되어야 하는 것은 아니며 순서가 다를 수 있다. 예를 들어, 전체 롤백을 시작할지 또는 일부 실패를 수정하기 위해 마이그레이션을 일시 중지할지 여부는 실패의 심각도를 결정하거나 마이그레이션 전에 실패가 이미 존재했는지 여부에 기초할 수 있다. 또한 이와 관련하여, 실패의 보고, 진단 및 수정은 마이그레이션이 일시 중지된 후 추가 또는 대안으로 발생할 수 있으며, 목적지 VM은 삭제되지 않고 대신 오류가 수정되면 마이그레이션이 재개될 수 있도록 유지된다.

[0127] 도 14는 하나 이상의 프로세서(212, 222)와 같은 하나 이상의 프로세서에 의해 수행될 수 있는 흐름도(1400)이다. 예를 들어, 프로세서(212, 222)는 흐름도에 도시된 바와 같이 데이터를 수신하고 다양한 결정을 내릴 수 있다. 도 14는 제1 클러스터의 제어 평면으로부터 제2 클러스터의 제어 평면으로의 라이브 마이그레이션의 예를 도시한다. 도 14를 참조하면, 블록 1410에서, 하나 이상의 클러스터 제어 평면에 대한 요청이 수신되며, 상기 하나 이상의 클러스터 제어 평면은 상기 제1 클러스터의 제어 평면 및 제2 클러스터의 제어 평면을 포함할 수 있다. 블록 1420에서, 상기 수신된 요청의 미리 결정된 부분은 상기 제2 클러스터의 제어 평면에 할당되고, 상기 수신된 요청의 나머지 부분을 상기 제1 클러스터의 제어 평면에 할당될 수 있다. 블록 1430에서, 요청의 미리 결정된 부분은 제2 클러스터의 제어 평면을 사용하여 처리된다. 블록 1440에서, 요청의 미리 결정된 부분을 처리하는 동안, 제2 클러스터에 실패가 있는지 여부를 검출한다. 블록 1450에서, 상기 제2 클러스터에서 실패를 검출하지 못한 것에 기초하여, 수신된 모든 요청이 제2 클러스터의 제어 평면에 할당될 때까지 미리 결정된 단계에서 상기 제2 클러스터의 제어 평면에 할당된 상기 요청의 미리 결정된 부분을 증가시킨다.

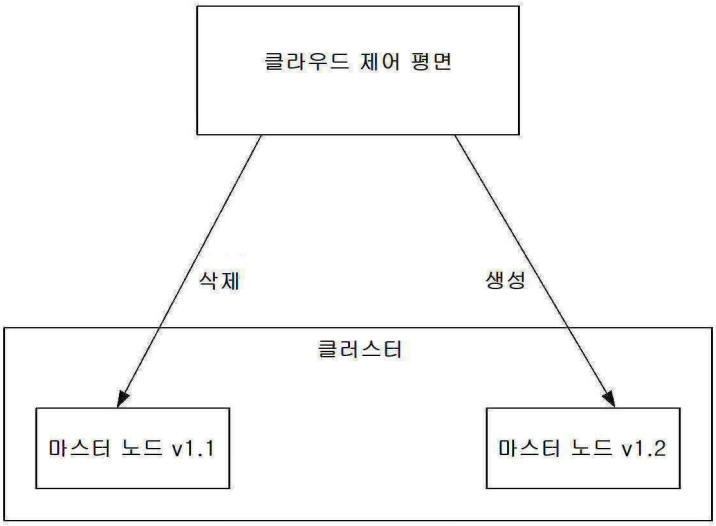
[0128] 이 기술은 클러스터를 업그레이드하거나 클러스터의 환경의 다른 양태를 수정하기 위해 점진적이고, 모니터링되는 롤아웃 프로세스를 제공하기 때문에 이점이 있다. 단계별 및 조용히 일부에 우선 적용되는(canaried) 롤아웃 프로세스는 문제가 발생할 경우 업그레이드를 중지할 수 있는 더 많은 기회를 제공하여 대규모 손해를 방지한다. 동시에 실행되는 소스 및 목적지 클러스터 간의 워크로드 트래픽 할당은 업그레이드 중에 다운타임을

줄이거나 제거할 수 있다. 또한 워크로드 트래픽 할당으로 인해, 클라이언트의 관점에서 라이브 마이그레이션 중에 하나의 클러스터만 존재하는 것처럼 보일 수 있다. 업그레이드가 실패한 경우, 성공적인 업그레이드가 완료되지 않는 한 소스 클러스터가 삭제되지 않으므로 시스템은 롤백 옵션을 제공한다. 이 기술은 또한 상이한 물리적 위치에 위치된 클러스터들 간 뿐만 아니라 클라우드 중 하나가 라이브 마이그레이션을 지원하지 않는 다른 클라우드에서 동작되는 클러스터들 간에 라이브 마이그레이션을 가능하게 하는 구성을 제공한다.

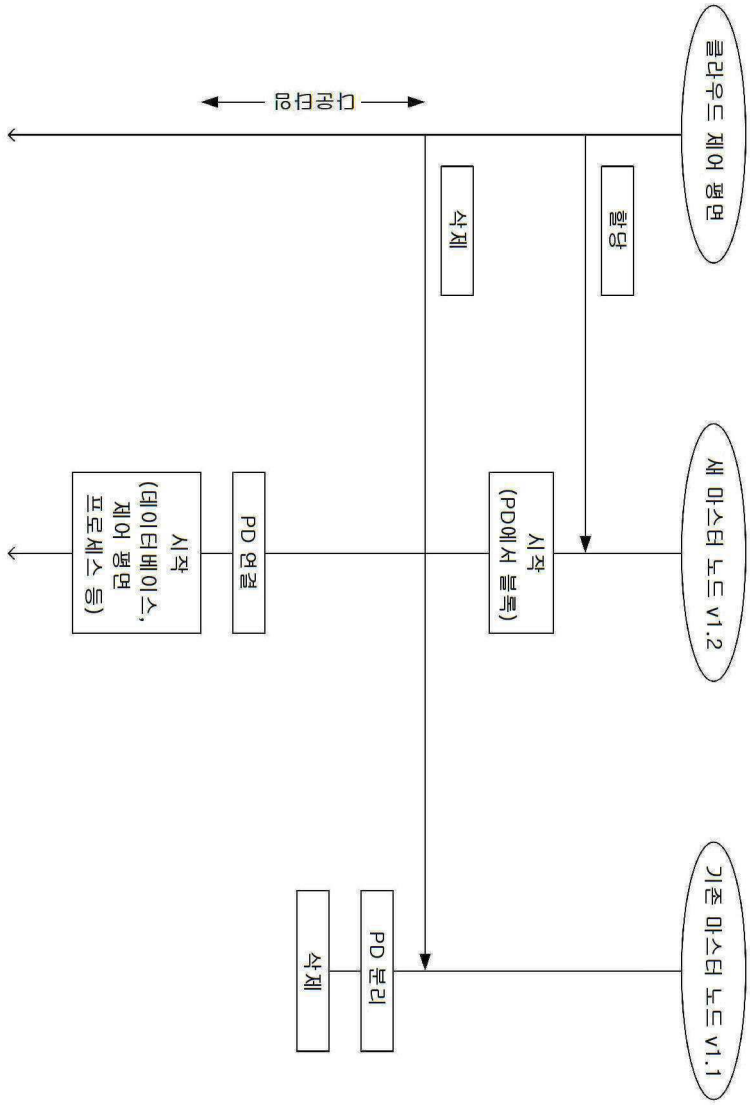
[0129] 달리 언급되지 않는 한, 전술한 대안적 예는 상호 배타적이지 않으며, 고유한 이점을 달성하기 위해 다양한 조합으로 구현될 수 있다. 상기 논의된 구성의 이러한 및 다른 변형 및 조합은 청구 범위에 의해 정의된 주제를 벗어나지 않고 활용될 수 있으므로, 실시예에 대한 전술한 설명은 청구 범위에 의해 정의된 주제의 제한이 아닌 예시의 방식으로 받아들여야 한다. 또한, "예를 들어", "포함하는" 등과 같이 표현된 문구 뿐만 아니라 본 명세서에 설명된 예시의 제공은 특정 예에 대한 청구 범위의 주제를 제한하는 것이 아니라 오히려 예는 많은 가능한 실시예 중 하나만을 예시하기 위한 것이라고 해석되어야 한다. 또한, 상이한 도면에서 동일한 참조 번호는 동일하거나 유사한 요소를 식별할 수 있다.

도면

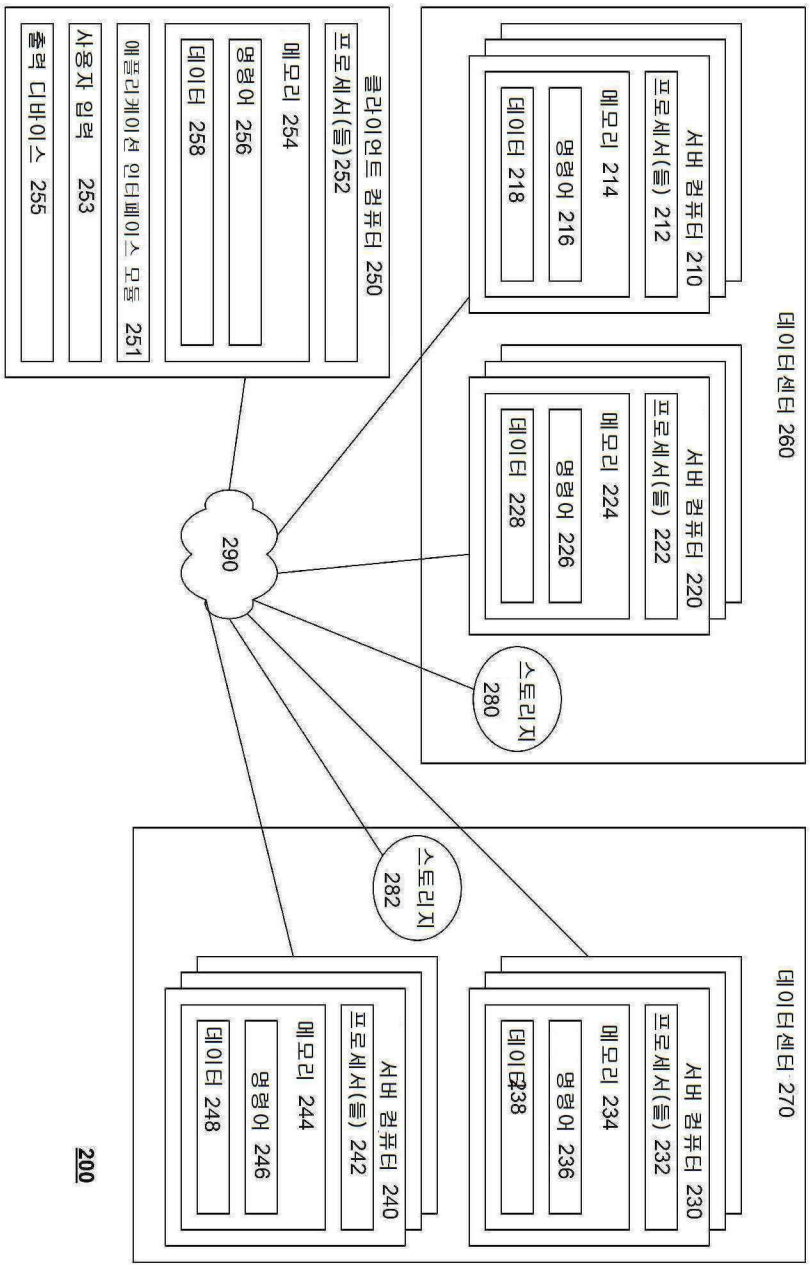
도면1a



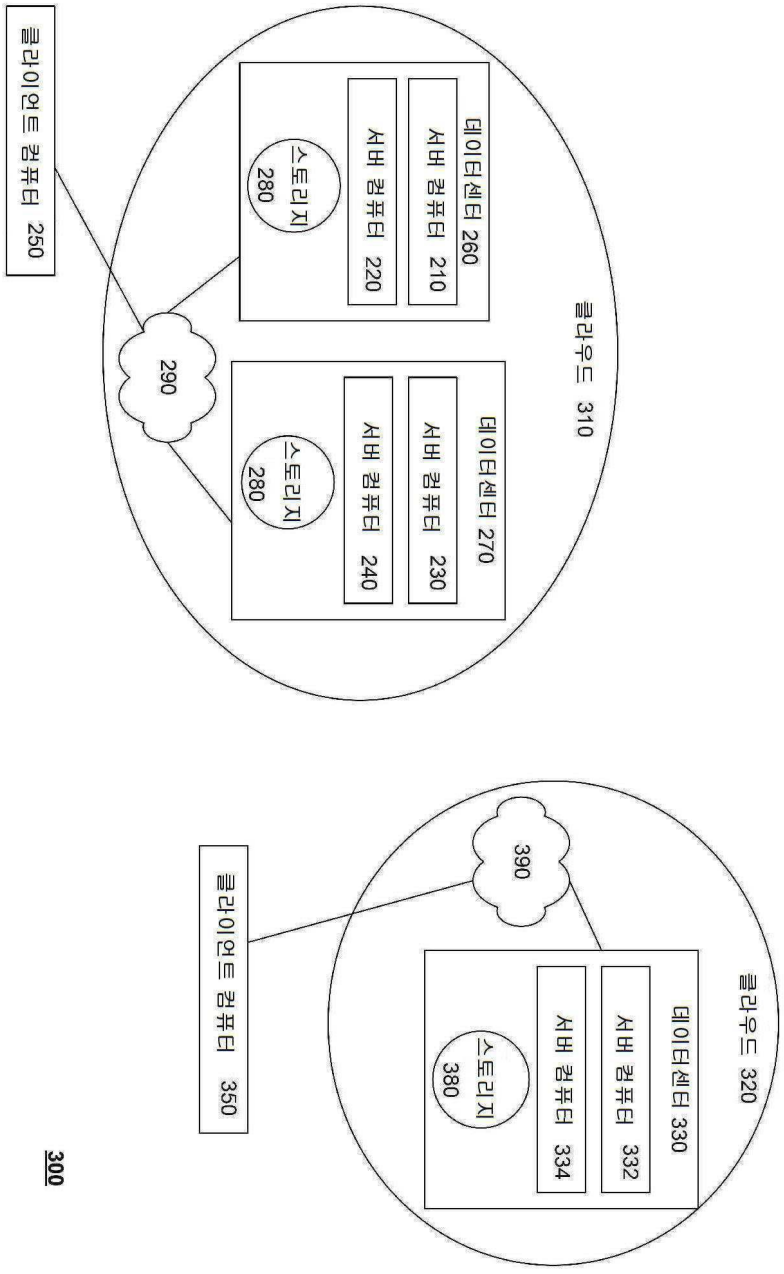
도면1b


클라우드 제어 평면
새 마스터 노드 v1.2
기존 마스터 노드 v1.1
다운타임
할당
삭제
시작
(PD에서 블록)
PD 연결
시작
(데이터베이스,
제어 평면
프로세스 등)
PD 분리
삭제

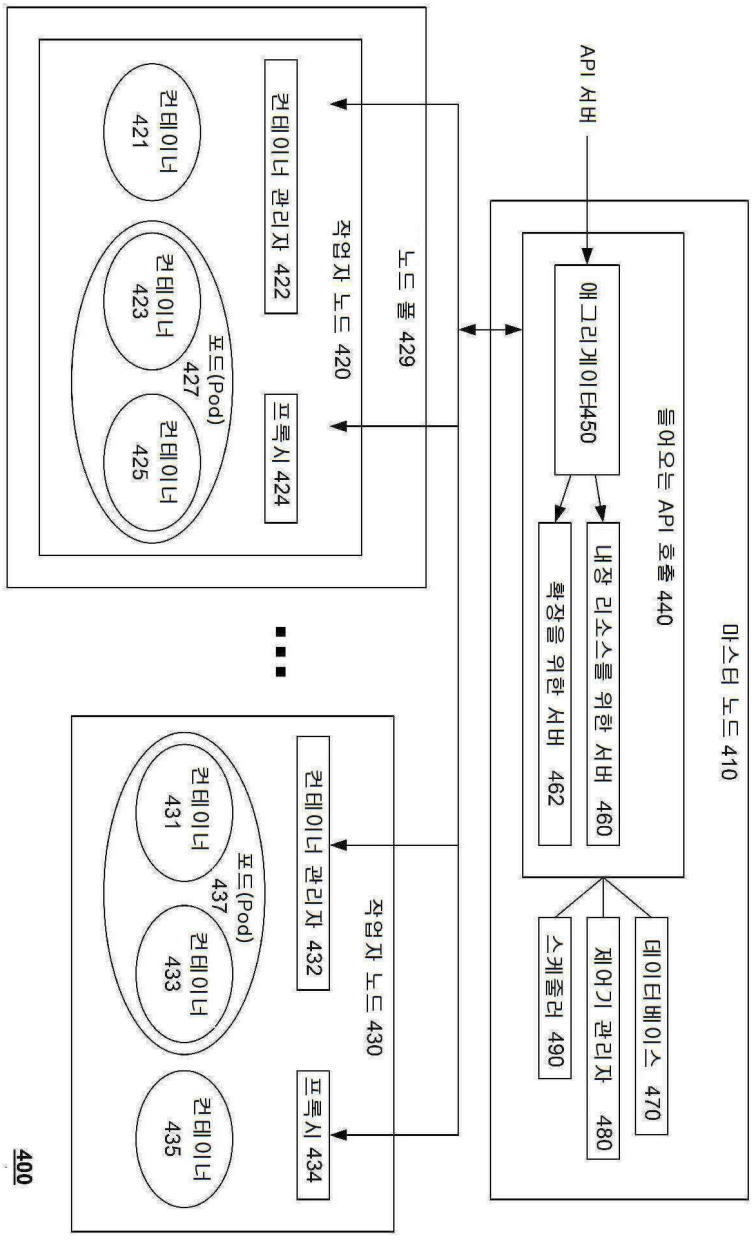
도면2


클라이언트 컴퓨터 250
프로세서(들)252
메모리 254
명령어 256
데이터 258
애플리케이션 인터페이스 모듈 251
사용자 입력 253
출력 디바이스 255
290
데이터센터 260
서버 컴퓨터 210
프로세서(들) 212
메모리 214
명령어 216
데이터 218
서버 컴퓨터 220
프로세서(들) 222
메모리 224
명령어 226
데이터 228
스토리지 280
데이터센터 270
서버 컴퓨터 230
프로세서(들) 232
메모리 234
명령어 236
데이터238
서버 컴퓨터 240
프로세서(들) 242
메모리 244
명령어 246
데이터 248
스토리지 282
200

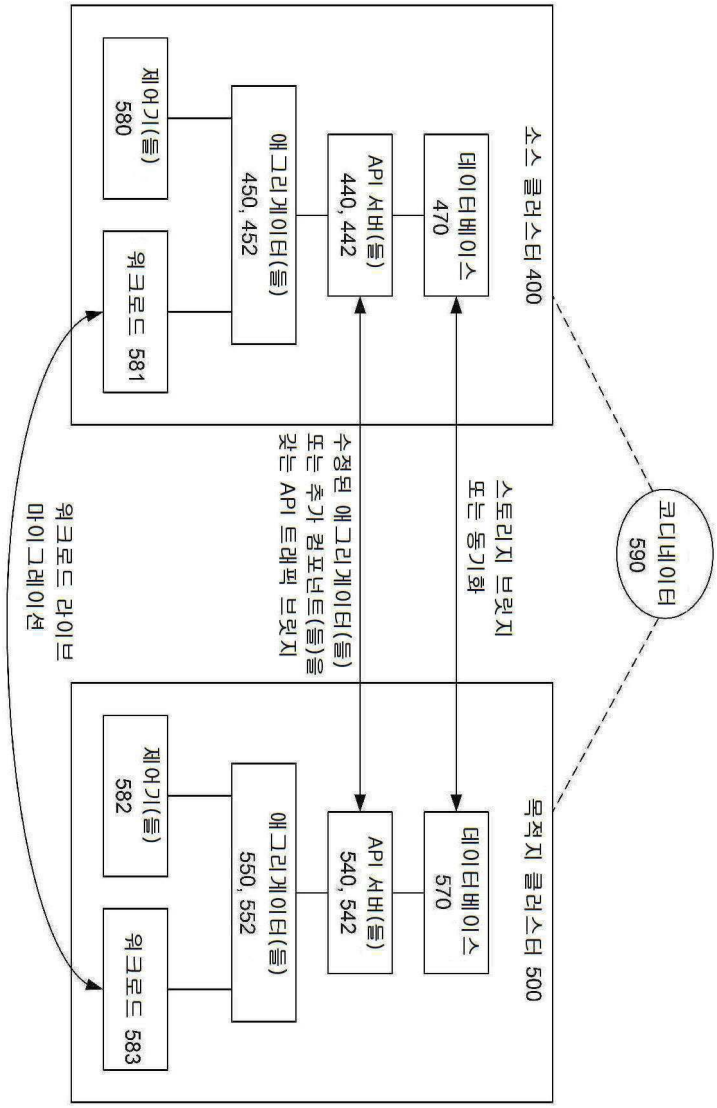
도면3


클라이언트 컴퓨터 250
290
클라우드 310
데이터센터 260
서버 컴퓨터 210
서버 컴퓨터 220
스토리지 280
데이터센터 270
서버 컴퓨터 230
서버 컴퓨터 240
스토리지 280
390
클라우드 320
데이터센터 330
서버 컴퓨터 332
서버 컴퓨터 334
스토리지 380
클라이언트 컴퓨터 350
300

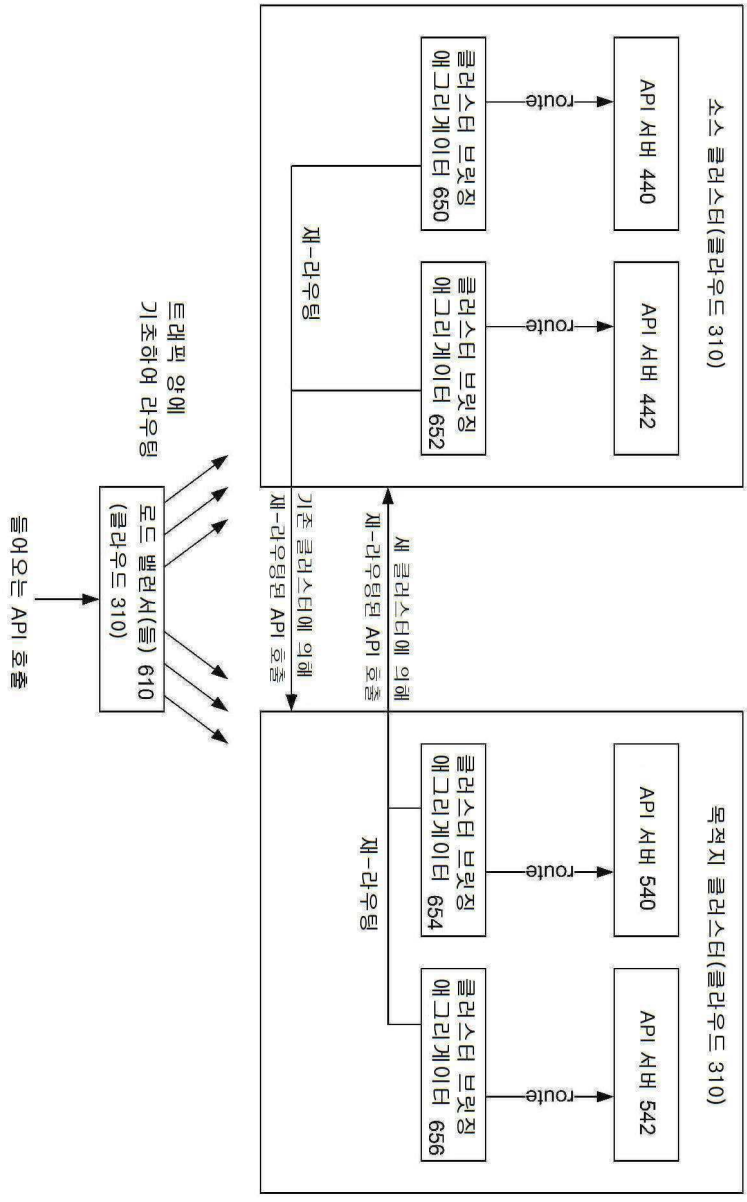
도면4


400
마스터 노드 410
들어오는 API 호출 440
애그리게이터450
내장 리소스를 위한 서버 460
확장을 위한 서버 462
스케줄러 490
제어기 관리자 480
데이터베이스 470
API 서버
노드 풀 429
작업자 노드 420
컨테이너 관리자 422
프록시 424
포드(Pod) 427
컨테이너 421
컨테이너 423
컨테이너 425
작업자 노드 430
컨테이너 관리자 432
프록시 434
포드(Pod) 437
컨테이너 431
컨테이너 433
컨테이너 435

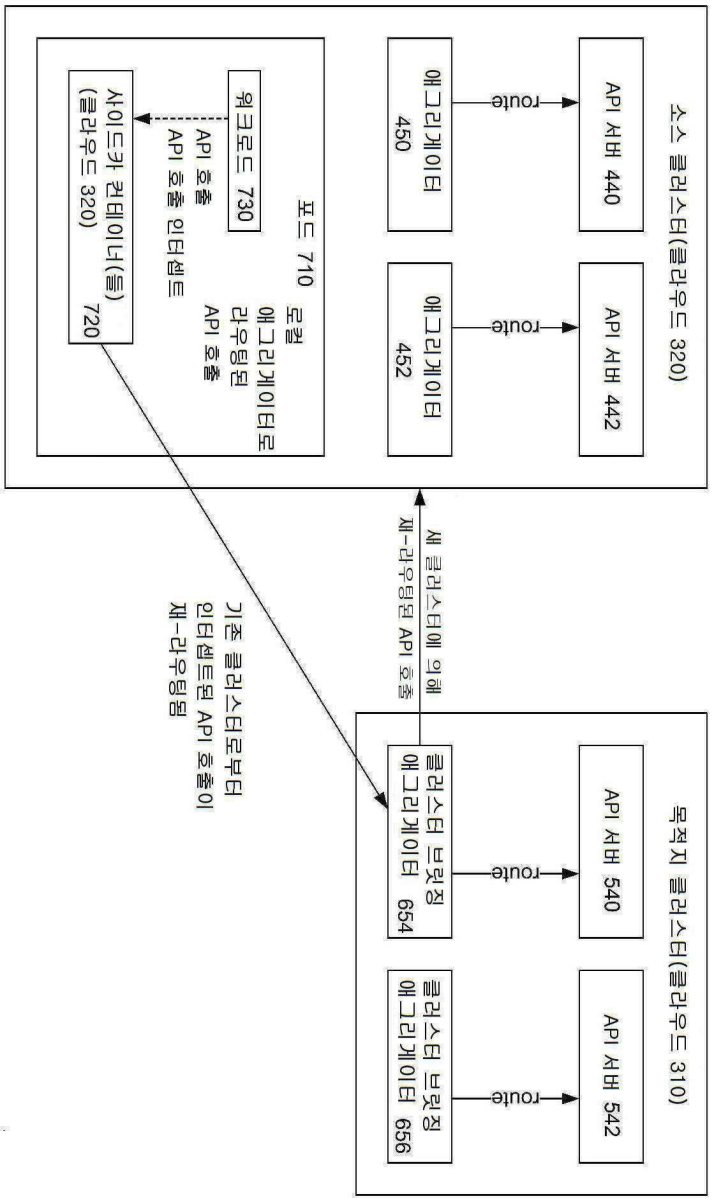
도면5


소스 클러스터 400
제어기(들) 580
워크로드 581
애그리게이터(들) 450, 452
API 서버(들) 440, 442
데이터베이스 470
수정된 애그리게이터(들) 또는 추가 컴포넌트(들)을 갖는 API 트래픽 브릿지
스토리지 브릿지 또는 동기화
코디네이터 590
워크로드 라이브 마이그레이션
목적지 클러스터 500
제어기(들) 582
워크로드 583
애그리게이터(들) 550, 552
API 서버(들) 540, 542
데이터베이스 570

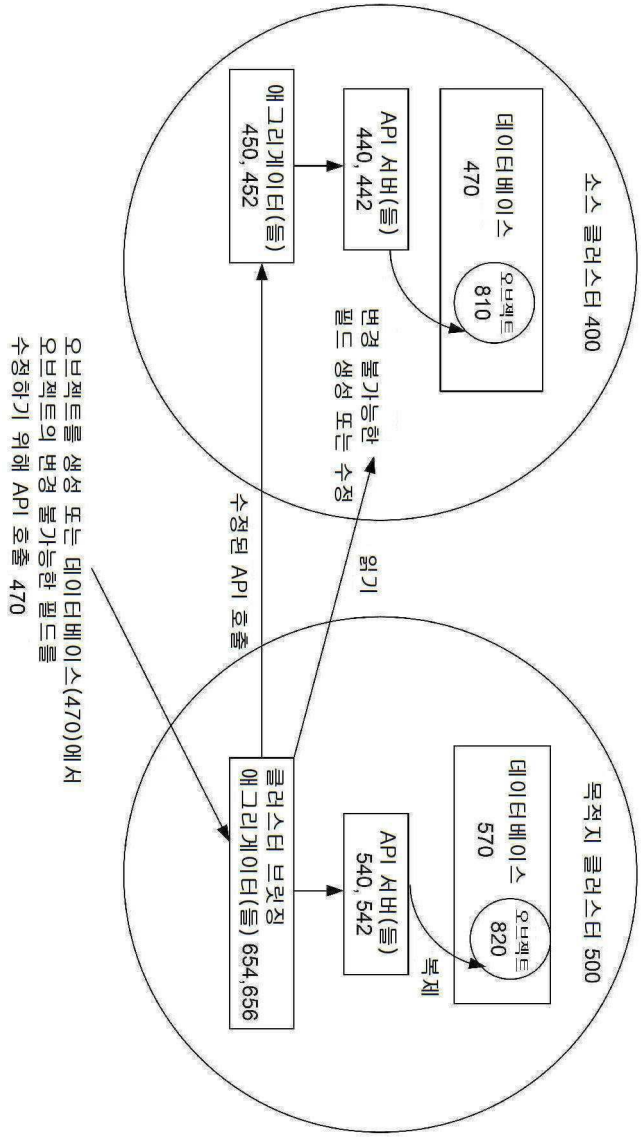
도면6


들어오는 API 호출
모드 밸런서(들) 610
(클라우드 310)
트래픽 양에
기초하여 라우팅
기존 클러스터에 의해
재-라우팅된 API 호출
새 클러스터에 의해
재-라우팅된 API 호출
재-라우팅
재-라우팅
소스 클러스터(클라우드 310)
클러스터 브릿징
애그리게이터 650
route
API 서버 440
클러스터 브릿징
애그리게이터 652
route
API 서버 442
목적지 클러스터(클라우드 310)
클러스터 브릿징
애그리게이터 654
route
API 서버 540
클러스터 브릿징
애그리게이터 656
route
API 서버 542

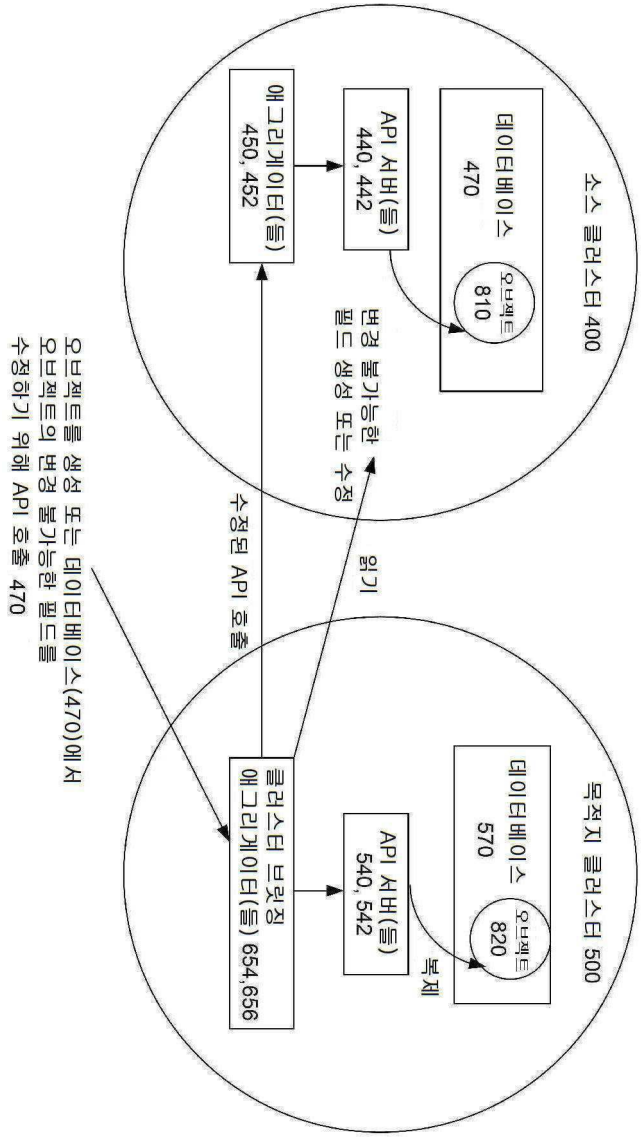
도면7


소스 클러스터(클라우드 320)
API 서버 440
route
애그리게이터 450
API 서버 442
route
애그리게이터 452
포드 710
로컬 애그리게이터로 라우팅된 API 호출
워크로드 730
API 호출
API 호출 인터셉트
사이드카 컨테이너(들) (클라우드 320) 720
새 클러스터에 의해 재-라우팅된 API 호출
기존 클러스터로부터 인터셉트된 API 호출이 재-라우팅됨
목적지 클러스터(클라우드 310)
클러스터 브릿징 애그리게이터 654
route
API 서버 540
클러스터 브릿징 애그리게이터 656
route
API 서버 542

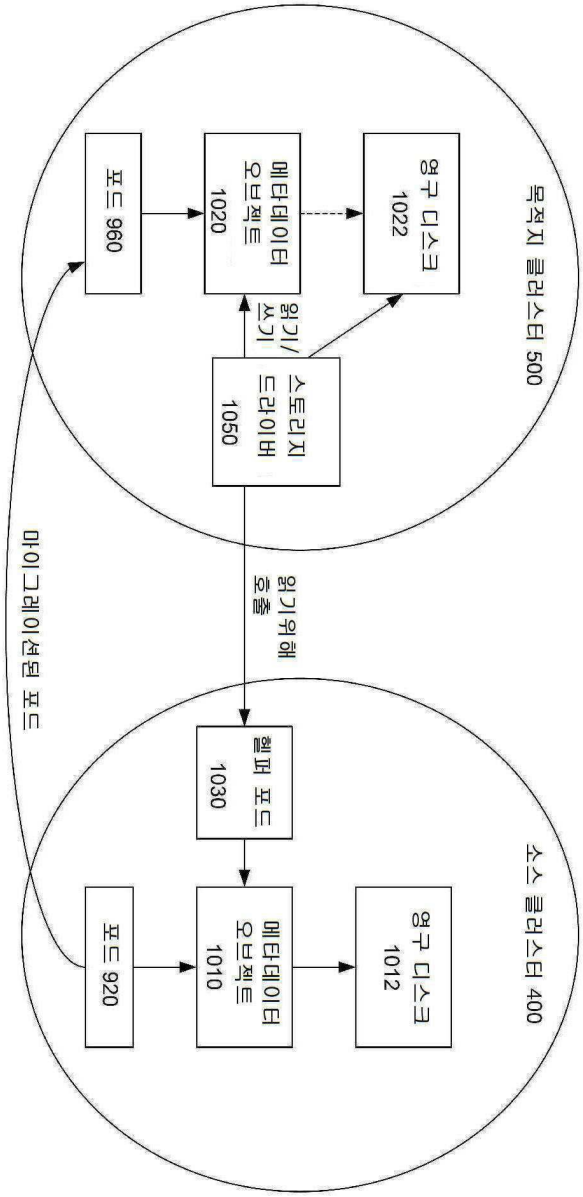
도면8


소스 클러스터 400
애그리게이터(들)
450, 452
API 서버(들)
440, 442
데이터베이스
470
오브젝트
810
변경 불가능한
필드 생성 또는 수정
수정된 API 호출
읽기
목적지 클러스터 500
클러스터 브릿징
애그리게이터(들) 654,656
API 서버(들)
540, 542
데이터베이스
570
오브젝트
820
복제
오브젝트를 생성 또는 데이터베이스(470)에서
오브젝트의 변경 불가능한 필드를
수정하기 위해 API 호출 470

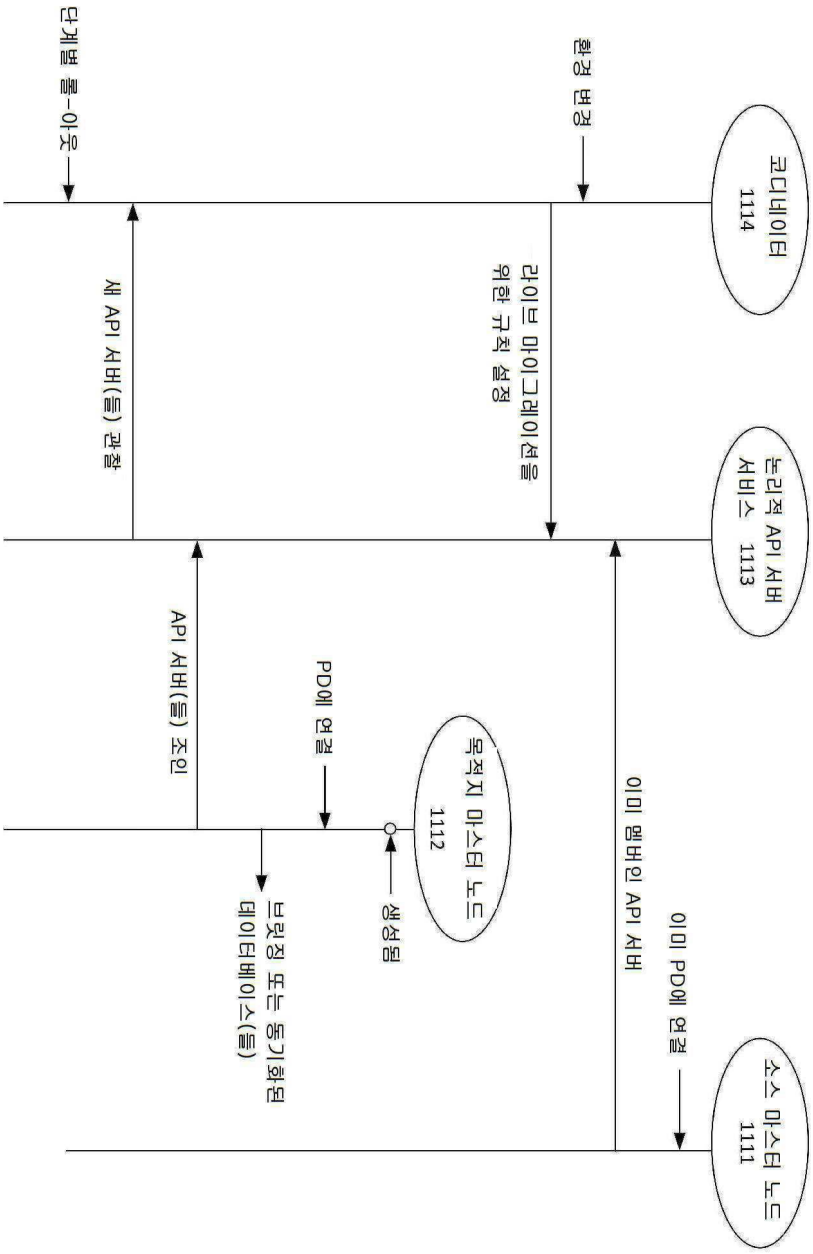
도면9


소스 클러스터 400
데이터베이스 470
오브젝트 810
API 서버(들) 440, 442
애그리게이터(들) 450, 452
변경 불가능한 필드 생성 또는 수정
수정된 API 호출
읽기
목적지 클러스터 500
데이터베이스 570
오브젝트 820
API 서버(들) 540, 542
클러스터 브릿징 애그리게이터(들) 654,656
복제
오브젝트를 생성 또는 데이터베이스(470)에서 오브젝트의 변경 불가능한 필드를 수정하기 위해 API 호출 470

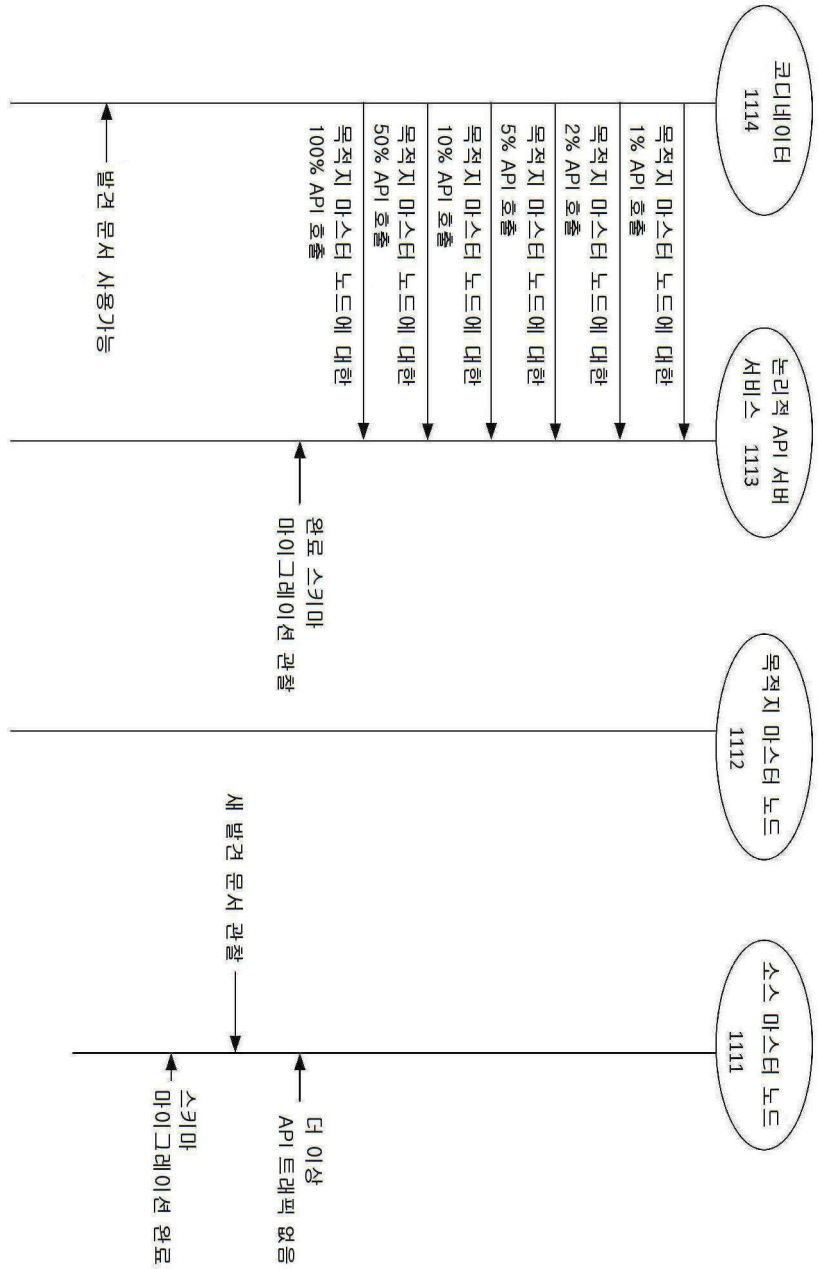
도면10


목적지 클러스터 500
포드 960
메타데이터 오브젝트 1020
영구 디스크 1022
읽기/쓰기
스토리지 드라이버 1050
읽기위해 호출
마이그레이션 포드
헬퍼 포드 1030
포드 920
메타데이터 오브젝트 1010
영구 디스크 1012
소스 클러스터 400

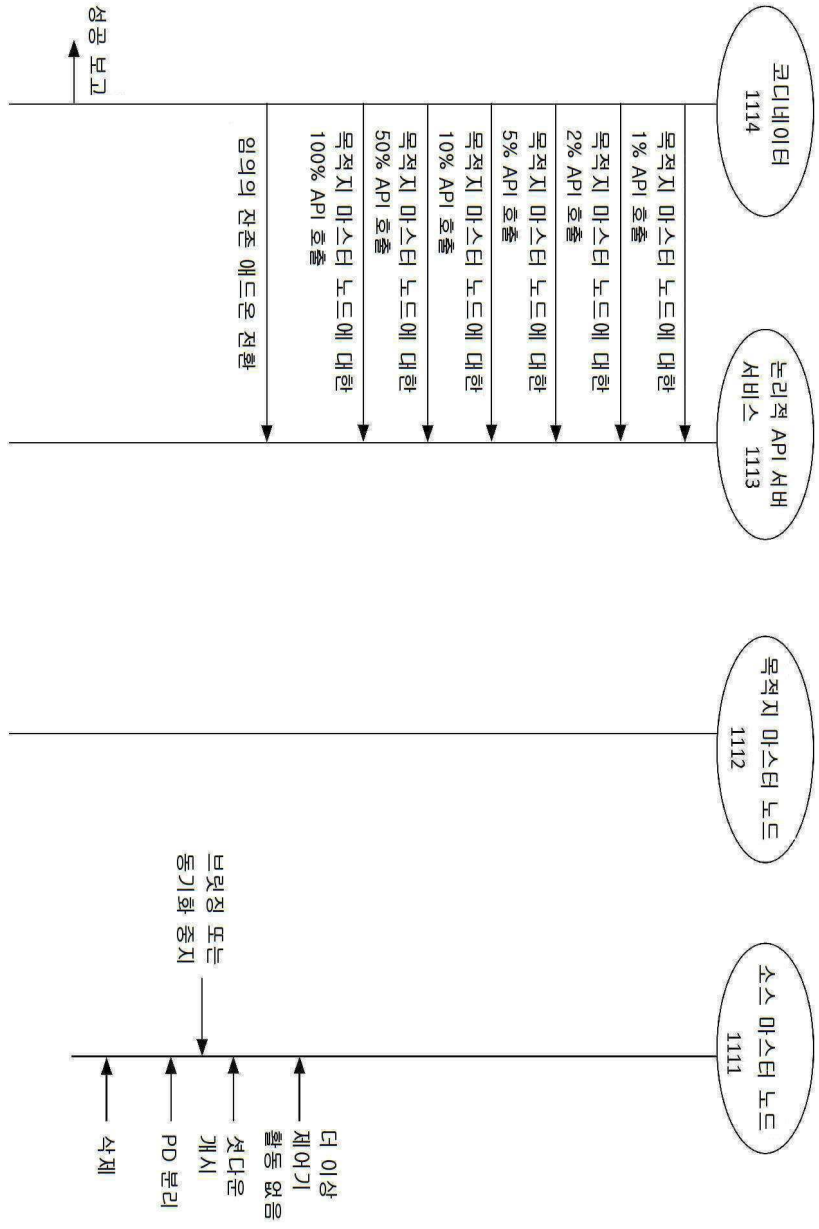
도면11a


코디네이터
1114
환경 변경
단계별 롤-아웃
논리적 API 서버
서비스 1113
라이브 마이그레이션을
위한 규칙 설정
새 API 서버(들) 관찰
목적지 마스터 노드
1112
생성됨
PD에 연결
API 서버(들) 조인
브릿징 또는 동기화된
데이터베이스(들)
소스 마스터 노드
1111
이미 PD에 연결
이미 멤버인 API 서버

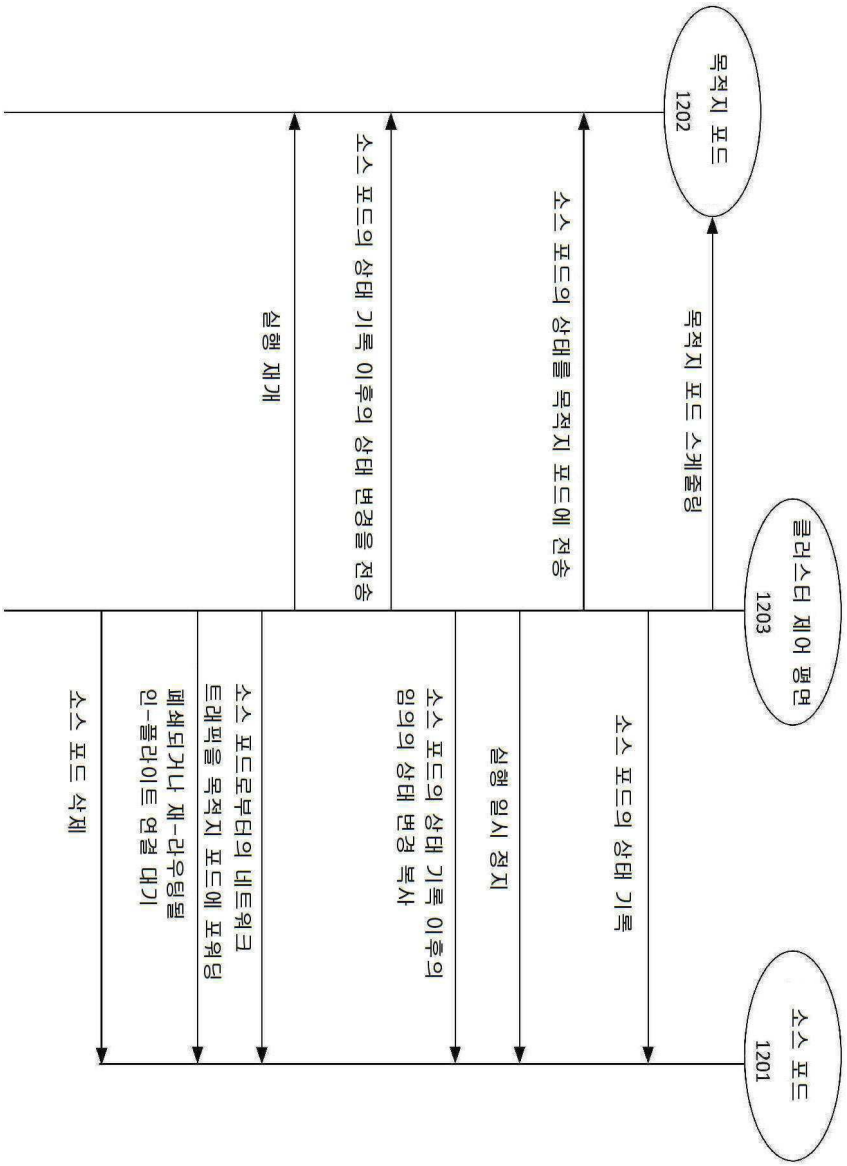
도면11b


코디네이터
1114
논리적 API 서버
서비스 1113
목적지 마스터 노드
1112
소스 마스터 노드
1111
목적지 마스터 노드에 대한
1% API 호출
목적지 마스터 노드에 대한
2% API 호출
목적지 마스터 노드에 대한
5% API 호출
목적지 마스터 노드에 대한
10% API 호출
목적지 마스터 노드에 대한
50% API 호출
목적지 마스터 노드에 대한
100% API 호출
발견 문서 사용가능
완료 스키마
마이그레이션 관찰
새 발견 문서 관찰
더 이상
API 트래픽 없음
스키마
마이그레이션 완료

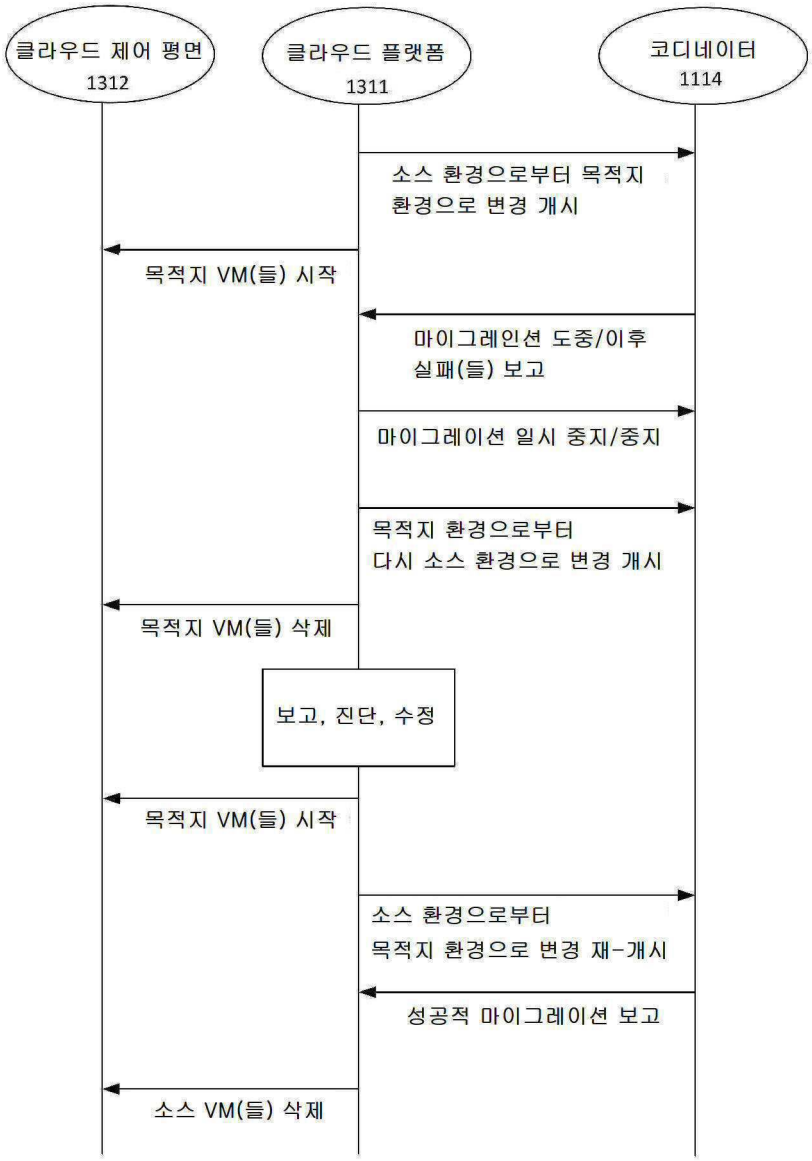
도면11c


코디네이터
1114
논리적 API 서버
서비스 1113
목적지 마스터 노드
1112
소스 마스터 노드
1111
성공 보고
목적지 마스터 노드에 대한
1% API 호출
목적지 마스터 노드에 대한
2% API 호출
목적지 마스터 노드에 대한
5% API 호출
목적지 마스터 노드에 대한
10% API 호출
목적지 마스터 노드에 대한
50% API 호출
목적지 마스터 노드에 대한
100% API 호출
임의의 잔존 에드온 전환
브릿징 또는
동기화 중지
더 이상
제어기
활동 없음
셧다운
개시
PD 분리
삭제

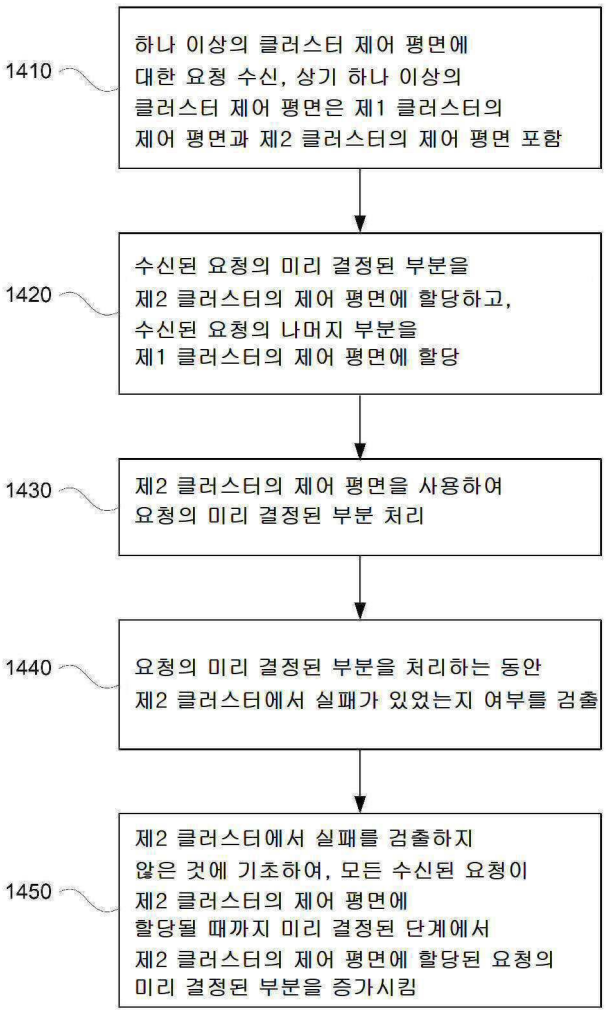
도면12


목적지 포드
1202
클러스터 제어 평면
1203
소스 포드
1201
목적지 포드 스케줄링
소스 포드의 상태를 목적지 포드에 전송
소스 포드의 상태 기록 이후의 상태 변경을 전송
실행 재개
소스 포드의 상태 기록
실행 일시 정지
소스 포드의 상태 기록 이후의 임의의 상태 변경 복사
소스 포드로부터의 네트워크 트래픽을 목적지 포드에 포워딩
폐쇄되거나 제-라우팅될 인-플라이트 연결 대기
소스 포드 삭제

도면13


클라우드 제어 평면
1312
클라우드 플랫폼
1311
코디네이터
1114
소스 환경으로부터 목적지
환경으로 변경 개시
목적지 VM(들) 시작
마이그레인션 도중/이후
실패(들) 보고
마이그레이션 일시 중지/중지
목적지 환경으로부터
다시 소스 환경으로 변경 개시
목적지 VM(들) 삭제
보고, 진단, 수정
목적지 VM(들) 시작
소스 환경으로부터
목적지 환경으로 변경 재-개시
성공적 마이그레이션 보고
소스 VM(들) 삭제

도면14


1410
하나 이상의 클러스터 제어 평면에 대한 요청 수신, 상기 하나 이상의 클러스터 제어 평면은 제1 클러스터의 제어 평면과 제2 클러스터의 제어 평면 포함
1420
수신된 요청의 미리 결정된 부분을 제2 클러스터의 제어 평면에 할당하고, 수신된 요청의 나머지 부분을 제1 클러스터의 제어 평면에 할당
1430
제2 클러스터의 제어 평면을 사용하여 요청의 미리 결정된 부분 처리
1440
요청의 미리 결정된 부분을 처리하는 동안 제2 클러스터에서 실패가 있었는지 여부를 검출
1450
제2 클러스터에서 실패를 검출하지 않은 것에 기초하여, 모든 수신된 요청이 제2 클러스터의 제어 평면에 할당될 때까지 미리 결정된 단계에서 제2 클러스터의 제어 평면에 할당된 요청의 미리 결정된 부분을 증가시킴

1400