(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2024-0132079
(43) 공개일자 2024년09월02일

(51) 국제특허분류(Int. Cl.)
G06F 9/50 (2018.01) *G06F 9/455* (2018.01)
H04L 41/12 (2022.01) *H04L 41/5051* (2022.01)
(52) CPC특허분류
G06F 9/5072 (2013.01)
G06F 9/45545 (2013.01)
(21) 출원번호 10-2024-7027042
(22) 출원일자(국제) 2022년12월20일
심사청구일자 없음
(85) 번역문제출일자 2024년08월12일
(86) 국제출원번호 PCT/US2022/082073
(87) 국제공개번호 WO 2023/136964
국제공개일자 2023년07월20일
(30) 우선권주장
63/298,685 2022년01월12일 미국(US)
18/050,392 2022년10월27일 미국(US)

(71) 출원인
오라클 인터내셔날 코포레이션
미국, 캘리포니아 94065, 레드우드 쇼어스 엠에스 5오피7, 오라클 파크웨이 500
(72) 발명자
브라르, 자그윈더
미국, 캘리포니아 94065, 레드우드 쇼어스 엠에스 5오피7, 오라클 파크웨이 500 오라클 인터내셔날 코포레이션 내
베커, 데이비드
미국, 캘리포니아 94065, 레드우드 쇼어스 엠에스 5오피7, 오라클 파크웨이 500 오라클 인터내셔날 코포레이션 내
코크마드, 해롤드 도넬
미국, 캘리포니아 94065, 레드우드 쇼어스 엠에스 5오피7, 오라클 파크웨이 500 오라클 인터내셔날 코포레이션 내
(74) 대리인
특허법인에이아이피

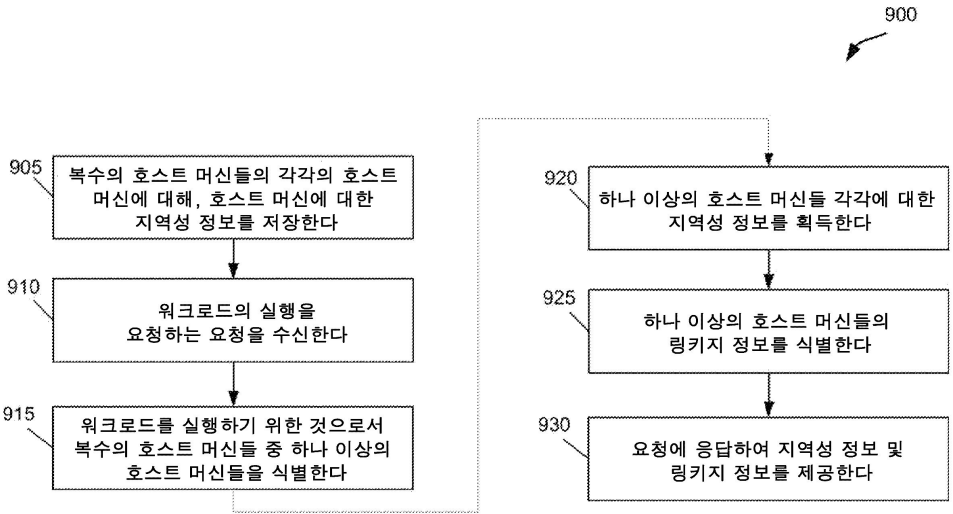
전체 청구항 수 : 총 20 항

(54) 발명의 명칭 **그래픽 프로세싱 유닛 워크로드들에 대한 물리적 토폴로지 네트워크 지역성 퍼블리싱**

(57) 요 약

그래픽 프로세싱 유닛 기반 워크로드들의 실행을 위해 클러스터 네트워크에 포함된 호스트 머신들의 지역성 정보를 사용하는 기술들이 본 명세서에서 논의된다. 복수의 호스트 머신들의 각각의 호스트 머신에 대해, 호스트 머신에 대한 지역성 정보가 내부에 저장된다. 호스트 머신에 대한 지역성 정보는 호스트 머신을 포함하는 랙을 식별한다. 워크로드의 실행을 요청하는 요청을 수신하는 것에 응답하여, 복수의 호스트 머신들 중 하나 이상의 호스트 머신들이 워크로드를 실행하기 위해 이용가능한 것으로서 식별된다. 하나 이상의 호스트 머신들 각각에 대해, 호스트 머신에 대한 지역성 정보가 획득된다. 또한, 하나 이상의 호스트 머신들의 링키지 정보가 식별된다. 하나 이상의 호스트 머신들의 지역성 정보 및 링키지 정보는 요청에 응답하여 제공된다.

대 표 도



(52) CPC특허분류
G06F 9/45558 (2013.01)
G06F 9/5027 (2013.01)
G06F 9/5083 (2013.01)
G06F 2009/45562 (2013.01)
G06F 2009/4557 (2019.08)
G06F 2009/45595 (2019.08)
G06F 2209/5022 (2013.01)

명 세 서

청구범위

청구항 1

방법으로서,

복수의 호스트 머신들의 각각의 호스트 머신에 대해, 상기 호스트 머신을 포함하는 랙을 식별하는 상기 호스트 머신에 대한 지역성 정보를 저장하는 단계; 및

워크로드의 실행을 요청하는 요청을 수신하는 것에 응답하여:

상기 복수의 호스트 머신들 중 하나 이상의 호스트 머신들을 상기 워크로드를 실행하기 위해 이용가능한 것으로서 식별하는 단계,

상기 하나 이상의 호스트 머신들 각각에 대해, 상기 호스트 머신에 대한 상기 지역성 정보를 획득하는 단계,

상기 하나 이상의 호스트 머신들의 링키지 정보를 식별하는 단계, 및

상기 요청에 응답하여 상기 하나 이상의 호스트 머신들의 상기 지역성 정보 및 상기 링키지 정보를 제공하는 단계를 포함하는, 방법.

청구항 2

제1항에 있어서,

상기 호스트 머신에 대한 상기 지역성 정보는 상기 호스트 머신을 포함하는 상기 랙의 랙 식별자를 포함하는 정보를 포함하는, 방법.

청구항 3

제1항에 있어서,

상기 호스트 머신의 상기 지역성 정보는 상기 랙과 연관된 탑-오브-랙(top-of-rack) 스위치의 식별자를 포함하는, 방법.

청구항 4

제1항에 있어서,

상기 하나 이상의 호스트 머신들의 상기 링키지 정보는 상기 하나 이상의 호스트 머신들에 의해 형성된 논리적 토폴로지에 대응하는, 방법.

청구항 5

제4항에 있어서,

상기 하나 이상의 호스트 머신들에 의해 형성된 상기 논리적 토폴로지는 링 토폴로지인, 방법.

청구항 6

제1항에 있어서,

상기 호스트 머신에 대한 상기 지역성 정보는 인스턴스 메타데이터 서비스로부터 획득되며, 상기 지역성 정보는 상기 호스트 머신과 연관된 네트워크 가상화 디바이스를 통해 상기 인스턴스 메타데이터 서비스에 의해 상기 호스트 머신에 저장되는, 방법.

청구항 7

제1항에 있어서,

상기 방법은,

고객에 의해, 상기 하나 이상의 호스트 머신들의 상기 지역성 정보 및 상기 링키지 정보를 수신하는 단계;

상기 고객에 의해, 상기 워크로드를 실행하기 위해 상기 하나 이상의 호스트 머신들의 서브세트를 선택하는 단계 - 상기 선택하는 단계는 상기 워크로드와 연관된 하나 이상의 제약들에 기초하여 수행됨 -; 및

상기 하나 이상의 호스트 머신들의 상기 서브세트에서 상기 워크로드를 실행하는 단계를 더 포함하는, 방법.

청구항 8

제7항에 있어서,

하나 이상의 제약들은 워크로드와 연관된 레이턴시 임계치에 대응하는 제1 제약 및 상기 워크로드의 실행 시의 희망되는 가용성의 정도와 연관된 제2 제약을 포함하는, 방법.

청구항 9

제8항에 있어서,

상기 하나 이상의 호스트 머신들의 상기 서브세트를 선택하는 단계는 상기 하나 이상의 호스트 머신들의 상기 링키지 정보에 더 기초하는, 방법.

청구항 10

하나 이상의 프로세서들에 의해 실행될 때 하기의 단계들을 수행하게 하는 컴퓨터-실행가능 명령어들을 저장하는 하나 이상의 컴퓨터 판독가능 비-일시적 매체로서, 상기 단계들은:

복수의 호스트 머신들의 각각의 호스트 머신에 대해, 상기 호스트 머신을 포함하는 랙을 식별하는 상기 호스트 머신에 대한 지역성 정보를 저장하는 단계; 및

워크로드의 실행을 요청하는 요청을 수신하는 것에 응답하여:

상기 복수의 호스트 머신들 중 하나 이상의 호스트 머신들을 상기 워크로드를 실행하기 위해 이용가능한 것으로서 식별하는 단계,

상기 하나 이상의 호스트 머신들 각각에 대해, 상기 호스트 머신에 대한 상기 지역성 정보를 획득하는 단계,

상기 하나 이상의 호스트 머신들의 링키지 정보를 식별하는 단계, 및

상기 요청에 응답하여 상기 하나 이상의 호스트 머신들의 상기 지역성 정보 및 상기 링키지 정보를 제공하는 단계를 포함하는, 컴퓨터-실행가능 명령어들을 저장하는 하나 이상의 컴퓨터 판독가능 비-일시적 매체.

청구항 11

제10항에 있어서,

상기 호스트 머신에 대한 상기 지역성 정보는 상기 호스트 머신을 포함하는 상기 랙의 랙 식별자를 포함하는 정보를 포함하는, 컴퓨터-실행가능 명령어들을 저장하는 하나 이상의 컴퓨터 판독가능 비-일시적 매체.

청구항 12

제10항에 있어서,

상기 호스트 머신의 상기 지역성 정보는 상기 랙과 연관된 탑-오브-랙 스위치의 식별자를 포함하는, 컴퓨터-실행가능 명령어들을 저장하는 하나 이상의 컴퓨터 판독가능 비-일시적 매체.

청구항 13

제10항에 있어서,

상기 하나 이상의 호스트 머신들의 상기 링키지 정보는 상기 하나 이상의 호스트 머신들에 의해 형성된 논리적 토폴로지에 대응하는, 컴퓨터-실행가능 명령어들을 저장하는 하나 이상의 컴퓨터 판독가능 비-일시적 매체.

청구항 14

제13항에 있어서,

상기 하나 이상의 호스트 머신들에 의해 형성된 상기 논리적 토폴로지는 링 토폴로지인, 컴퓨터-실행가능 명령어들을 저장하는 하나 이상의 컴퓨터 판독가능 비-일시적 매체.

청구항 15

제10항에 있어서,

상기 호스트 머신에 대한 상기 지역성 정보는 인스턴스 메타데이터 서비스로부터 획득되며, 상기 지역성 정보는 상기 호스트 머신과 연관된 네트워크 가상화 디바이스를 통해 상기 인스턴스 메타데이터 서비스에 의해 상기 호스트 머신에 저장되는, 컴퓨터-실행가능 명령어들을 저장하는 하나 이상의 컴퓨터 판독가능 비-일시적 매체.

청구항 16

제10항에 있어서,

하나 이상의 프로세서들에 의해 실행될 때 하기의 단계들을 수행하게 하는 컴퓨터-실행가능 명령어들을 더 포함하며, 상기 단계들은:

고객에 의해, 상기 하나 이상의 호스트 머신들의 상기 지역성 정보 및 상기 링키지 정보를 수신하는 단계;

상기 워크로드를 실행하기 위해 상기 하나 이상의 호스트 머신들의 서브세트를 선택하는 단계 - 상기 선택하는 단계는 상기 워크로드와 연관된 하나 이상의 제약들에 기초하여 수행됨 -; 및

상기 하나 이상의 호스트 머신들의 상기 서브세트에서 상기 워크로드를 실행하는 단계를 포함하는, 컴퓨터-실행가능 명령어들을 저장하는 하나 이상의 컴퓨터 판독가능 비-일시적 매체.

청구항 17

제16항에 있어서,

하나 이상의 제약들은 워크로드와 연관된 레이턴시 임계치에 대응하는 제1 제약 및 상기 워크로드의 실행 시의 희망되는 가용성의 정도와 연관된 제2 제약을 포함하는, 컴퓨터-실행가능 명령어들을 저장하는 하나 이상의 컴

퓨터 판독가능 비-일시적 매체.

청구항 18

컴퓨팅 디바이스로서,

하나 이상의 프로세서들; 및

명령어들을 포함하는 메모리를 포함하며, 상기 명령어들은 상기 하나 이상의 프로세서들을 이용하여 실행될 때, 상기 컴퓨팅 디바이스가 적어도:

복수의 호스트 머신들의 각각의 호스트 머신에 대해, 상기 호스트 머신을 포함하는 랙을 식별하는 상기 호스트 머신에 대한 지역성 정보를 저장하고; 그리고

워크로드의 실행을 요청하는 요청을 수신하는 것에 응답하여:

상기 복수의 호스트 머신들 중 하나 이상의 호스트 머신들을 상기 워크로드를 실행하기 위해 이용가능한 것으로서 식별하며,

상기 하나 이상의 호스트 머신들 각각에 대해, 상기 호스트 머신에 대한 상기 지역성 정보를 획득하고,

상기 하나 이상의 호스트 머신들의 링키지 정보를 식별하며, 그리고

상기 요청에 응답하여 상기 하나 이상의 호스트 머신들의 상기 지역성 정보 및 상기 링키지 정보를 제공하게 하는, 컴퓨팅 디바이스.

청구항 19

제18항에 있어서,

상기 호스트 머신에 대한 상기 지역성 정보는 상기 호스트 머신을 포함하는 상기 랙의 랙 식별자를 포함하는 정보를 포함하는, 컴퓨팅 디바이스.

청구항 20

제18항에 있어서,

상기 호스트 머신의 상기 지역성 정보는 상기 랙과 연관된 탑-오브-랙 스위치의 식별자를 포함하는, 컴퓨팅 디바이스.

발명의 설명

기 술 분 야

[0001] 관련 출원들에 대한 상호 참조들

[0002] 본 출원은 2022년 01월 12일자로 출원된 미국 가특허 출원 제63/298,685호 및 2022년 10월 27일자로 출원된 미국 정규 특허출원 제18/050,392호에 대한 우선권을 주장한다. 전술한 출원들의 전체 내용들은 모든 목적들을 위하여 그 전체가 본원에 참조로서 통합된다.

[0003] 기술분야

[0004] 본 개시내용은 그래픽 프로세싱 유닛 기반 워크로드(workload)들의 실행 시에 클라우드 인프라스트럭처의 원격 직접 메모리 액세스(remote direct memory access; RDMA)-가능 클러스터 네트워크에 포함된 호스트 머신들의 지역성(locality) 정보를 사용하는 것에 관한 것이다.

배 경 기 술

[0005] 조직들은 사내(on premise) 하드웨어 및 소프트웨어의 구매, 업데이트, 유지 관리 비용을 감소시키기 위해 비즈니스 애플리케이션들 및 데이터베이스들을 클라우드로 계속해서 이동시키고 있다. 고성능 컴퓨터 애플리케이션들은 특정 성과 또는 결과들 달성하기 위해 이용가능한 모든 컴퓨팅 파워를 지속적으로 소비한다. 이러한 애플리케이션들은, 오늘날 상품 클라우드들을 구성하는 가상화된 인프라스트럭처에서 공급이 부족한 자원들 - 전용 네트워크 성능, 고속 스토리지, 높은 컴퓨팅 능력들, 및 상당한 양의 메모리를 필요로 한다.

[0006] 클라우드 인프라스트럭처 제공자들은 이러한 애플리케이션들의 요건들을 해결하기 위해 더 새롭고 더 빠른 그래픽 프로세싱 유닛(graphical processing unit; GPU)들을 제공한다. GPU 워크로드는 전형적으로 하나 이상의 호스트 머신들에서 실행된다. 전형적으로, 이러한 워크로드들은 예측된 레벨의 스루풋을 달성할 수 없다. 이러한 문제에 대한 하나의 요인은 흐름 엔트로피(flow entropy), 예를 들어, 등가 비용 다중-경로(equal cost multi-path; ECMP) 흐름 엔트로피의 결여이다. 또한, 이러한 문제는, 호스트 머신들(즉, 호스트들)이 그들의 로컬 네트워크 이웃에 있는 다른 호스트들과 관계없이 트래픽을 교환한다는 사실에 의해 더 악화된다. 본 명세서에서 논의되는 실시예들은 이러한 그리고 다른 문제들을 다룬다.

발명의 내용

[0007] 본 개시내용은 전반적으로 그래픽 프로세싱 유닛 기반 워크로드들의 실행 시에 클라우드 인프라스트럭처의 RDMA-가능 클러스터 네트워크에 포함된 호스트 머신들의 지역성 정보를 사용하는 것에 관한 것이다. 하나 이상의 프로세서들에 의해 실행가능한 프로그램들, 코드, 또는 명령어들을 저장하는 비-일시적인 컴퓨터-판독가능 저장 매체, 방법들, 시스템들, 및 유사한 것을 포함하는 다양한 실시예들에 본 명세서에서 설명된다. 이러한 예시적인 실시예들은 본 개시내용을 제한하거나 또는 정의하도록 언급되는 것이 아니라 본 개시내용의 이해를 보조하기 위한 예들을 제공하도록 언급된다. 추가적인 실시예들이 상세한 설명에서 논의되며, 추가적인 설명이 상세한 설명에서 제공된다.

[0008] 본 개시내용의 일 실시예는 방법에 관한 것으로서, 방법은: 복수의 호스트 머신들의 각각의 호스트 머신에 대해, 상기 호스트 머신을 포함하는 랙을 식별하는 호스트 머신에 대한 지역성 정보를 저장하는 단계; 및 워크로드의 실행을 요청하는 요청을 수신하는 것에 응답하여: 복수의 호스트 머신들 중 하나 이상의 호스트 머신들을 워크로드를 실행하기 위해 이용가능한 것으로서 식별하는 단계; 하나 이상의 호스트 머신들 각각에 대해, 호스트 머신에 대한 상기 지역성 정보를 획득하는 단계; 하나 이상의 호스트 머신들의 링키지 정보를 식별하는 단계; 및 요청에 응답하여 하나 이상의 호스트 머신들의 지역성 정보 및 링키지 정보를 제공하는 단계를 포함한다.

[0009] 본 개시내용의 일 양태는, 하나 이상의 데이터 프로세서들 및 하나 이상의 데이터 프로세서들 상에서 실행될 때 컴퓨팅 디바이스가 본 명세서에 개시된 하나 이상의 방법들 중 부분 또는 전부를 수행하게끔 하는 명령어들을 포함하는 비-일시적 컴퓨터 판독가능 저장 매체를 포함하는 컴퓨팅 디바이스를 제공한다.

[0010] 본 개시내용의 다른 양태는, 비-일시적 기계-판독가능 저장 매체에 유형적으로 구현되며, 하나 이상의 데이터 프로세서들이 본 명세서에서 개시된 하나 이상의 방법들의 부분 또는 전부를 수행하게끔 하도록 구성된 명령어들을 포함하는 컴퓨터-프로그램 제품을 제공한다.

[0011] 다른 특징들 및 실시예들과 함께 전술한 내용이 다음의 명세서, 청구항들, 및 첨부된 도면들을 참조할 때 더 명백해질 것이다.

도면의 간단한 설명

[0012] 본 개시내용의 특징들, 실시예들 및 이점들은 첨부된 도면들을 참조하여 다음의 상세한 설명을 읽을 때 더 잘 이해된다.

도 1은 특정 실시예들에 따른, 클라우드 서비스 제공자 인프라스트럭처에 의해 호스팅되는 가상 또는 오버레이 클라우드 네트워크를 보여주는 분산형 환경의 하이 레벨 도면이다.

도 2는 특정 실시예들에 따른, CSPI 내의 물리적 네트워크 내의 물리적 구성요소들의 단순화된 아키텍처 도면을 도시한다.

도3은 특정 실시예들에 따른, 호스트 머신이 다수의 네트워크 가상화 디바이스(network virtualization device; NVD)에 연결되는 CSPI 내의 예시적인 배열을 도시한다.

도 4는 특정 실시예들에 따른, 다중테넌시(multitenancy)를 지원하기 위한 I/O 가상화를 제공하기 위한 NVD과 호스트 머신 사이의 연결성을 도시한다.

도 5는 특정 실시예들에 따른, CSPI에 의해 제공되는 물리적 네트워크의 단순화된 블록도를 도시한다.

도 6은 특정 실시예들에 따른 클로스(CLOS) 네트워크 배열을 통합하는 클라우드 인프라스트럭처의 단순화된 블록도를 도시한다.

도 7은 특정 실시예들에 다른 클라우드 인프라스트럭처 내에 포함된 랙(rack)의 예시적인 구성을 예시한다.

도 8a는 특정 실시예들에 따른 지역성 정보 없이 구성된 논리적 토폴로지를 도시한다.

도 8b는 특정 실시예들에 따른 지역성 정보로 구성된 다른 논리적 토폴로지를 도시한다.

도 9는 특정 실시예들에 따른 요청을 프로비저닝하는 데 수행되는 단계들을 도시하는 예시적인 흐름도를 예시한다.

도 10은 특정 실시예들에 따른 요청을 프로세싱하는 데 수행되는 단계들을 도시하는 예시적인 흐름도를 예시한다.

도 11은 특정 실시예들에 따른 고객의 워크로드 요청을 프로비저닝하는 데 수행되는 단계들을 도시하는 예시적인 흐름도를 예시한다.

도 12는 적어도 일 실시예에 따른 서비스 시스템으로서 클라우드 인프라스트럭처를 구현하기 위한 하나의 패턴을 예시하는 블록도이다.

도 13은 적어도 일 실시예에 따른, 서비스 시스템으로서 클라우드 인프라스트럭처를 구현하기 위한 다른 패턴을 예시하는 블록도이다.

도 14는 적어도 일 실시예에 따른, 서비스 시스템으로서 클라우드 인프라스트럭처를 구현하기 위한 다른 패턴을 예시하는 블록도이다.

도 15는 적어도 일 실시예에 따른 서비스 시스템으로서 클라우드 인프라스트럭처를 구현하기 위한 다른 패턴을 예시하는 블록도이다.

도 16은 적어도 일 실시예에 따른 예시적인 컴퓨터 시스템을 예시하는 블록도이다.

발명을 실시하기 위한 구체적인 내용

[0013] 다음의 설명에서, 설명의 목적들을 위해, 특정 세부사항들이 특정 실시예들의 완전한 이해를 제공하기 위해 기술된다. 그러나 다양한 실시예들은 이러한 특정 세부사항들 없이 실시될 수 있음이 명백할 것이다. 도면들 및 설명은 제한적인 것으로 의도되지 않는다. 단어 "예시적인"은 "예, 예증, 또는 예시로서 기능함"을 의미하도록 본 명세서에서 사용된다. "예시적인" 것으로서 본 명세서에서 설명된 임의의 실시예 또는 설계는 다른 실시예들 또는 설계들에 비해 반드시 바람직하거나 또는 유리한 것으로서 해석되지는 않아야 한다.

[0014] 클라우드 인프라스트럭처의 예시적인 아키텍처

[0015] 용어 클라우드 서비스는 일반적으로, CSP에 의해 제공되는 시스템들 및 인프라스트럭처(클라우드 인프라스트럭처)를 사용하여, 클라우드 서비스 제공자(cloud services provider; CSP)에 의해 (예를 들어, 가입 모델을 통해) 주문형으로 사용자들 또는 고객들에게 이용가능하게 되는 서비스를 지칭하기 위해 사용된다. 전형적으로, CSP의 인프라스트럭처를 구성하는 서버들 및 시스템들은 고객의 자체적인 사내 서버들 및 시스템들로부터 분리된다. 따라서, 고객들은, 서비스들에 대한 하드웨어 및 소프트웨어 자원들을 구매할 필요 없이 CSP에 의해 제공되는 클라우드 서비스들 자체를 이용할 수 있다. 클라우드 서비스들은, 고객이 서비스들을 제공하기 위해 사용되는 인프라스트럭처를 조달하는 데 투자할 필요 없이 가입 고객에게 애플리케이션들 및 컴퓨팅 자원들에 대한 쉽고 스케일링가능한 액세스를 제공하도록 설계된다.

[0016] 다양한 유형들의 클라우드 서비스들을 제공하는 몇몇 클라우드 서비스 제공자들이 있다. 서비스형 소프트웨어(Software as a Service; SaaS), 서비스형 플랫폼(Platform as a Service; PaaS), 서비스형 인프라스트럭처(Infrastructure as a Service; IaaS), 및 다른 것들을 포함하여 다양하고 상이한 유형들 또는 모델들의 클라우드 서비스들이 있다.

[0017] 고객은 CSP에 의해 제공되는 하나 이상의 클라우드 서비스들에 가입할 수 있다. 고객은 개인, 조직, 기업, 및 유사한 것과 같은 임의의 엔티티일 수 있다. 고객이 CSP에 의해 제공되는 서비스들에 가입하거나 또는 등록할 때, 테넌시 또는 계정이 해당 고객에 대해 생성된다. 그러면, 고객은, 이러한 계정을 통해, 계정과 연관된 가입된 하나 이상의 클라우드 자원들에 액세스할 수 있다.

[0018] 이상에서 언급된 바와 같이, 서비스형 인프라스트럭처(infrastructure as a service; IaaS)는 클라우드 컴퓨팅 서비스의 하나의 특정 유형이다. IaaS 모델에서, CSP는, 그들 자신의 맞춤화가능 네트워크들을 구축하고 고객 자원들을 배포하기 위해 고객들에 의해 사용될 수 있는 인프라스트럭처(클라우드 서비스 제공자 인프라스트럭처 또는 CSPI로 지칭됨)를 제공한다. 따라서 고객의 자원들 및 네트워크들은 CSP에 의해 제공된 인프라스트럭처에 의해 분산형 환경에서 호스팅된다. 이는, 고객의 자원들 및 네트워크들이 고객에 의해 제공된 인프라스트럭처에 의해 호스팅되는 전통적인 컴퓨팅과는 상이하다.

[0019] CSPI는, 서브스트레이트(substrate) 네트워크 또는 언더레이 네트워크로도 지칭되는 물리적 네트워크를 형성하는 다양한 호스트 머신들, 메모리 자원들 및 네트워크 자원들을 포함하여 상호연결된 고성능 컴퓨팅 자원들을 포함할 수 있다. CSPI 내의 자원들은, 하나 이상의 지리적 지역들에 걸쳐 분산될 수 있는 하나 이상의 데이터 센터들에 걸쳐 분산될 수 있다. 가상 소프트웨어는 가상화된 분산형 환경을 제공하기 위해 이러한 물리적 자원들에 의해 실행될 수 있다. 가상화는 물리적 네트워크 위에 오버레이 네트워크(소프트웨어-기반 네트워크, 소프트웨어-정의형 네트워크, 또는 가상 네트워크로 알려짐)를 생성한다. CSPI인 물리적 네트워크는, 물리적 네트워크의 상단 상에 하나 이상의 오버레이 또는 가상 네트워크들을 생성하기 위한 기본 기반을 제공한다. 가상 또는 오버레이 네트워크들은 하나 이상의 가상 클라우드 네트워크(virtual cloud network; VCN)들을 포함할 수 있다. 가상 네트워크들은, 물리적 네트워크의 상단 상에서 실행될 수 있는 네트워크 추상화의 계층들을 생성하기 위해 소프트웨어 가상화 기술들(예를 들어, 하이퍼바이저(hypervisor)들, 네트워크 가상화 디바이스(network virtualization device; NVD)들(예를 들어, smartNIC들)에 의해 수행되는 기능들, 탑-오브-랙(top-of-rack; TOR) 스위치들, NVD에 의해 수행되는 하나 이상의 기능들을 구현하는 스마트 TOR들, 및 다른 메커니즘들)을 사용하여 구현된다. 가상 네트워크들은, 피어-투-피어 네트워크들, IP 네트워크들, 및 다른 것들을 포함하여 다수의 형태들을 취할 수 있다. 가상 네트워크들은 전형적으로 계층-3 IP 네트워크들 또는 계층-2 VLAN들이다. 가상 또는 오버레이 네트워킹의 이러한 방법은 흔히 가상 또는 오버레이 계층-3 네트워킹으로 지칭된다. 가상 네트워크들에 대해 개발된 프로토콜들의 예들은, IP-인-IP(IP-in-IP)(또는 일반 라우팅 캡슐화(Generic Routing Encapsulation; GRE), 가상 확장가능 LAN(VXLAN - IETF RFC 7348), 가상 사설 네트워크(Virtual Private Network; VPN)들(예를 들어, MPLS 계층-3 가상 사설 네트워크들(RFC 4364)), VMware의 NSX, GENEVE(Generic Network Virtualization Encapsulation), 및 다른 것들을 포함한다.

[0020] IaaS에 대해, CSP에 의해 제공된 인프라스트럭처(CSPI)는 공용 네트워크(예를 들어, 인터넷)를 통해 가상화된 컴퓨팅 자원들을 제공하도록 구성될 수 있다. IaaS 모델에서, 클라우드 컴퓨팅 서비스 제공자는 인프라스트럭처 구성요소들(예를 들어, 서버들, 저장 디바이스들, 네트워크 노드들(예를 들어, 하드웨어), 배포 소프트웨어, 플랫폼 가상화(예를 들어, 하이퍼바이저 계층), 또는 유사한 것)을 호스팅할 수 있다. 일부 경우들에서, IaaS 제공자는 또한 이러한 인프라스트럭처 구성요소들에 수반되는 다양한 서비스들(예를 들어, 청구, 모니터링, 로깅, 보안, 로드 밸런싱 및 클러스팅, 등)을 공급할 수 있다. 따라서, 이러한 서비스들이 정책-구동될 수 있기 때문에, IaaS 사용자들은 애플리케이션 가용성 및 성능을 유지하기 위한 로드 밸런싱을 구동하기 위한 정책들을 구현할 수 있다. CSPI는, 고객들이 고도로 이용가능한 호스팅되는 분산형 환경에서 광범위한 애플리케이션 및 서비스들을 구축하고 실행하는 것을 가능하게 하는 상보적인 클라우드 서비스들의 세트 및 인프라스트럭처를 제공한다. CSPI는, 고객의 사내 네트워크로부터와 같이 다양한 네트워킹된 위치들로부터 안전하게 액세스가능한 유연한 가상 네트워크에서 고성능 컴퓨팅 자원들과 능력들 및 스토리지 용량을 제공한다. 고객이 CSP에 의해 제공되는 IaaS 서비스들에 가입하거나 또는 등록할 때, 해당 고객에 대해 생성된 테넌시는, 고객이 그들의 클라우드 자원들을 생성하고, 조직하며, 관리할 수 있는 CSPI 내의 안전하고 격리된 파티션이다.

[0021] 고객들은 CSPI에 의해 제공되는 컴퓨팅, 메모리, 및 네트워킹 자원들을 사용하여 그들 자체의 가상 네트워크들을 구축할 수 있다. 컴퓨팅 인스턴스들과 같은 하나 이상의 고객 자원들 또는 워크로드들은 이러한 가상 네트워크들 상에 배포될 수 있다. 예를 들어, 고객은, 가상 클라우드 네트워크(virtual cloud network; VCN)들로 지칭되는 하나 또는 다수의 맞춤화가능 사설 가상 네트워크(들)를 구축하기 위해 CSPI에 의해 제공되는 자원들을 사용할 수 있다. 고객은 고객 VCN 상에 컴퓨팅 인스턴스들과 같은 하나 이상의 고객 자원들을 배포할 수 있다. 컴퓨팅 인스턴스들은 가상 머신들, 베어 메탈(bare metal) 인스턴스들, 및 유사한 것의 형태를 취할 수 있다. 따라서, CSPI는, 고객들이 고도로 이용가능한 가상 호스팅형 환경에서 광범위한 애플리케이션 및 서비스들을 구축

하고 실행하는 것을 가능하게 하는 상보적인 클라우드 서비스들의 세트 및 인프라스트럭처를 제공한다. 고객은 CSPI에 의해 제공된 기초 물리적 자원들을 관리하거나 또는 제어하는 것이 아니라, 운영 체제들, 스토리지, 및 배포된 애플리케이션에 대한 제어; 및 가능하게는 선택 네트워킹 구성요소들(예를 들어, 방화벽들)의 제한된 제어를 갖는다.

[0022] CSP는, 고객들 및 네트워크 관리자들이 CSP 자원들을 사용하여 클라우드에 배포되는 자원들을 구성하고, 액세스하며, 관리하는 것을 가능하게 하는 콘솔을 제공할 수 있다. 특정 실시예들에서, 콘솔은, CSPI에 액세스하고 이를 관리하기 위해 사용될 수 있는 웹-기반 사용자 인터페이스를 제공한다. 일부 구현예들에서, 콘솔은 CSP에 의해 제공되는 웹-기반 애플리케이션이다.

[0023] CSPI는 단일-테넌시 또는 다중-테넌시 아키텍처들을 지원할 수 있다. 단일 테넌시 아키텍처에서, 소프트웨어(예를 들어, 애플리케이션, 데이터베이스) 또는 하드웨어 구성요소(예를 들어, 호스트 머신 또는 서버)는 단일 고객 또는 테넌트를 서비스한다. 다중-테넌시 아키텍처에서, 소프트웨어 또는 하드웨어 구성요소는 다수의 고객들 또는 테넌트들을 서비스한다. 따라서, 다중-테넌시 아키텍처에서, CSPI 자원들은 다수의 고객들 또는 테넌트들 사이에서 공유된다. 다중-테넌시 상황에서, 각각의 테넌트의 데이터가 격리되고 다른 테넌트들에 비가시적으로 남아 있는 것을 보장하기 위해 CSPI 내에서 예방 조치들이 취해지고 세이프가드들이 제위치에 마련된다.

[0024] 물리적 네트워크에서, 네트워크 엔드포인트("엔드포인트")는, 물리적 네트워크에 연결되며 이것이 연결된 네트워크와 앞뒤로 통신하는 컴퓨팅 디바이스 또는 시스템을 의미한다. 물리적 네트워크 내의 네트워크 엔드포인트는 근거리 네트워크(Local Area Network; LAN), 광역 네트워크(Wide Area Network; WAN), 또는 다른 유형의 물리적 네트워크에 연결될 수 있다. 물리적 네트워크 내의 전통적인 엔드포인트들의 예들은, 모뎀들, 허브들, 브리지들, 스위치들, 라우터들, 및 다른 네트워킹 디바이스들, 물리적 컴퓨터들(또는 호스트 머신들), 및 유사한 것을 포함한다. 물리적 네트워크 내의 각각의 물리적 디바이스는 디바이스와 통신하기 위해 사용될 수 있는 고정된 네트워크 어드레스를 갖는다. 이러한 고정된 네트워크 어드레스는 계층-2 어드레스(예를 들어, MAC 어드레스), 고정된 계층-3 어드레스(예를 들어, IP 어드레스), 및 유사한 것일 수 있다. 가상화된 환경 또는 가상 네트워크에서, 엔드포인트들은, 물리적 네트워크의 구성요소들에 의해 호스팅되는(예를 들어, 물리적 호스트 머신들에 의해 호스팅되는) 가상 머신들과 같은 다양한 가상 엔드포인트들을 포함할 수 있다. 가상 네트워크 내의 이러한 엔드포인트들은 오버레이 계층-2 어드레스들(예를 들어, 오버레이 MAC 어드레스들) 및 오버레이 계층-3 어드레스들(예를 들어, 오버레이 IP 어드레스들)과 같은 오버레이 어드레스들에 의해 어드레싱된다. 네트워크 오버레이들은, 네트워크 관리자들이 (예를 들어, 가상 네트워크에 대한 제어 평면을 구현하는 소프트웨어를 통해) 소프트웨어 관리를 사용하여 네트워크 엔드포인트들과 연관된 오버레이 어드레스들을 이동시키는 것을 가능하게 함으로써 유연성을 가능하게 한다. 따라서, 물리적 네트워크와는 달리, 가상 네트워크에서, 오버레이 어드레스(예를 들어, 오버레이 IP 어드레스)는 네트워크 관리 소프트웨어를 사용하여 하나의 엔드포인트로부터 다른 것으로 이동될 수 있다. 가상 네트워크가 물리적 네트워크의 상단 상에 구축되기 때문에, 가상 네트워크 내의 구성요소들 사이의 통신들은 가상 네트워크 및 기초 물리적 네트워크 둘 모두를 수반한다. 이러한 통신들을 용이하게 하기 위해, CSPI의 구성요소들은, 가상 네트워크 내의 오버레이 어드레스들은 서브스트레이트 네트워크 내의 실제 물리적 어드레스들에 매핑하거나 또는 그 반대인 매핑들을 학습하고 저장하도록 구성된다. 그런 다음, 이러한 매핑들은 통신들을 용이하게 하기 위해 사용된다. 고객 트래픽은 가상 네트워크에서의 라우팅을 용이하게 하기 위해 캡슐화된다.

[0025] 따라서, 물리적 어드레스들(예를 들어, 물리적 IP 어드레스들)은 물리적 네트워크들 내의 구성요소들과 연관되며, 오버레이 어드레스들(예를 들어, 오버레이 IP 어드레스들)은 가상 네트워크들 내의 엔티티들과 연관된다. 물리적 IP 어드레스들 및 오버레이 IP 어드레스들 둘 모두는 실제 IP 어드레스들의 유형들이다. 이들은 가상 IP 어드레스들과는 별개이며, 여기서 가상 IP 어드레스는 다수의 실제 IP 어드레스들에 매핑된다. 가상 IP 어드레스는 가상 IP 어드레스와 다수의 실제 IP 어드레스들 사이의 1-대-다 매핑을 제공한다.

[0026] 클라우드 인프라스트럭처 또는 CSPI는 전세계의 하나 이상의 지역들 내의 하나 이상의 데이터 센터들에서 물리적으로 호스팅된다. CSPI는 물리적 또는 서브스트레이트 네트워크 내의 구성요소들 및 물리적 네트워크 구성요소들의 상단 상에 구축된 가상 네트워크 내에 있는 가상화된 구성요소들(예를 들어, 가상 네트워크들, 컴퓨팅 인스턴스들, 가상 머신들, 등)을 포함할 수 있다. 특정 실시예들에서, CSPI는 렐름(realm)들, 지역들 및 가용성 도메인들로 조직되고 호스팅된다. 지역(region)은 전형적으로 하나 이상의 데이터 센터들을 포함하는 국부화된 지리적 영역(area)이다. 지역들은 일반적으로 서로 독립적이며, 예를 들어, 국가들에 걸쳐 또는 심지어 대륙들에 걸쳐 먼 거리로 분리될 수 있다. 예를 들어, 제1 지역은 호주에 있을 수 있고, 다른 지역은 일본에 있을 수 있으며, 또 다른 지역은 인도에 있을 수 있는 등이다. CSPI 자원들은, 각각의 지역이 CSPI 자원들의 그 자체의

독립적인 서브세트를 갖도록 지역들 사이에서 분할된다. 각각의 지역은, 컴퓨팅 자원들(예를 들어, 베어 메탈 서버들, 가상 머신, 컨테이너들 및 관련된 인프라스트럭처, 등); 저장 자원들(예를 들어, 블록 볼륨 스토리지, 파일 스토리지, 객체 스토리지, 아카이브 스토리지); 네트워킹 자원들(예를 들어, 가상 클라우드 네트워크(VCN)들, 로드 밸런싱 자원들, 사내 네트워크들에 대한 연결들), 데이터베이스 자원들; 에지 네트워킹 자원들(예를 들어, DNS); 및 액세스 관리 및 모니터링 자원들, 및 다른 것들과 같은 코어 인프라스트럭처 서비스들 및 자원들의 세트를 제공할 수 있다. 각각의 지역은 일반적으로 이를 렐름 내의 다른 지역들에 연결하는 다수의 경로들을 갖는다.

[0027] 일반적으로, 애플리케이션은, 근처의 자원들이 거리가 있는 자원들을 사용하는 것보다 더 빠르기 때문에, 애플리케이션이 가장 많이 사용되는 지역에 배포된다(즉, 해당 지역과 연관된 인프라스트럭처 상에 배포된다). 애플리케이션들은 또한, 법적 관할권들, 과세 영역들, 및 다른 사업적 또는 사회적 기준, 및 유사한 것에 대한 다양한 요건들을 충족시키기 위해, 대규모 기상 시스템들 또는 지진들과 같은 지역-전체(region-wide) 이벤트들의 위험을 완화하기 위한 중복성과 같은 다양한 이유들로 상이한 지역들에 배포될 수 있다.

[0028] 지역 내의 데이터 센터들은 가용성 도메인(availability domain; AD)들로 추가로 조직되고 세분화될 수 있다. 가용성 도메인은 지역 내에 위치된 하나 이상의 데이터 센터들에 대응할 수 있다. 지역은 하나 이상의 가용성 도메인들로 구성될 수 있다. 이러한 분산형 환경에서, CSPI 자원들은 가상 클라우드 네트워크(virtual cloud network; VCN)와 같이 지역-특정 또는 컴퓨팅 인스턴스와 같이 가용성-도메인 특정일 수 있다.

[0029] 지역 내의 AD들은 서로 격리되고, 장애에 내성이 있으며, 동시에 고장이 날 가능성이 매우 낮도록 구성된다. 이는, 지역 내의 하나의 AD에서의 장애가 동일한 지역 내의 다른 AD들의 가용성에 영향을 줄 가능성이 낮도록 네트워킹, 물리적 케이블들, 케이블 경로들, 케이블 진입 지점들과 같은 중요 인프라스트럭처 자원들을 공유하지 않는 AD들에 의해 달성된다. 동일한 지역 내의 AD들은 저 레이턴시, 고 대역폭 네트워크에 의해 서로 연결될 수 있으며, 이는, 다른 네트워크들(예를 들어, 인터넷, 고객의 사내 네트워크들, 등)에 대한 고-가용성 연결성을 제공하고 고-가용성 및 재해 복구 둘 모두를 위해 복제된 시스템들을 다수의 AD들에 구축하는 것을 가능하게 한다. 클라우드 서비스들은 고 가용성을 보장하고 자원 장애에 대해 보호하기 위해 다수의 AD들을 사용한다. IaaS 제공자들에 의해 제공되는 인프라스트럭처가 성장함에 따라, 더 많은 지역들 및 AD들에 추가적인 용량이 추가될 수 있다. 가용성 도메인들 사이의 트래픽은 일반적으로 암호화된다.

[0030] 특정 실시예들에서, 지역들은 렐름들로 그룹화된다. 렐름은 지역들의 논리적 집합이다. 렐름들은 서로 격리되며 어떠한 데이터도 공유하지 않는다. 동일한 렐름 내의 지역들은 서로 통신할 수 있지만, 상이한 렐름들 내의 지역들은 그럴 수 없다. CSP를 갖는 고객의 테넌시 또는 계정은 단일 렐름에 존재하며, 해당 렐름에 속한 하나 이상의 지역들에 걸쳐 분산될 수 있다. 전형적으로, 고객이 IaaS 서비스에 가입할 때, 테넌시 또는 계정은 렐름 내의 고객-지정 지역("홈" 지역으로 지칭됨) 내의 해당 고객에 대해 생성된다. 고객은 렐름 내의 하나 이상의 다른 지역들에 걸쳐 고객의 테넌시를 확장할 수 있다. 고객은, 고객의 테넌시가 존재하는 렐름에 있지 않는 지역들에 액세스할 수 없다.

[0031] IaaS 제공자는 다수의 렐름들을 제공할 수 있으며, 각각의 렐름은 고객들 또는 사용자들의 특정 세트에 맞춰 제공된다. 예를 들어, 상업적 렐름은 상업적 고객들에 대해 제공될 수 있다. 다른 예로서, 렐름은 해당 국가 내의 고객들에 대해 특정 국가에 대해 제공될 수 있다. 또 다른 예로서, 정부 렐름은 정부 및 유사한 것에 대해 제공될 수 있다. 예를 들어, 정부 렐름은 특정 정부에 맞춰 제공될 수 있으며, 상업적 렐름보다 높은 보안 레벨을 가질 수 있다. 예를 들어, 오라클 클라우드 인프라스트럭처(Oracle Cloud Infrastructure; OCI)는 현재 상업적 영역(region)들에 대한 렐름 및 정부 클라우드 영역들에 대한 2개의 렐름들(예를 들어, FedRAMP 인가됨 및 IL5 인가됨)을 제공한다.

[0032] 특정 실시예들에서, AD는 하나 이상의 장애 도메인(fault domain)들로 세분될 수 있다. 장애 도메인은 안티-어피티니(anti-affinity)를 제공하기 위한 AD 내의 인프라스트럭처 자원들을 그룹화한 것이다. 장애 도메인들은, 인스턴스들이 단일 AD 내의 동일한 물리적 하드웨어 상에 있지 않도록 컴퓨팅 인스턴스들의 분산을 가능하게 한다. 이는 안티-어피니티로 알려져 있다. 장애 도메인은 단일 장애 지점을 공유하는 하드웨어 구성요소들(컴퓨터들, 스위치들, 등)의 세트를 의미한다. 컴퓨팅 풀은 장애 도메인들로 논리적으로 분할된다. 이로 인해, 하나의 장애 도메인에 영향을 주는 하드웨어 장애 또는 컴퓨터 하드웨어 유지보수는 다른 장애 도메인들 내의 인스턴스들에 영향을 주지 않는다. 실시예에 의존하여, 각각의 AD에 대한 장애 도메인들의 수는 변화할 수 있다. 예를 들어, 특정 실시예들에서, 각각의 AD는 3개의 장애 도메인들을 포함한다. 장애 도메인들은 AD 내에서 논리적 데이터 센터로서 역할한다.

[0033] 고객이 IaaS 서비스에 가입할 때, CSPI로부터의 자원들은 고객에 대해 프로비저닝되고 고객의 텐넌시와 연관된다. 고객은 사설 네트워크들을 구축하고 이러한 네트워크들 상에 자원들을 배포하기 위해 이러한 프로비저닝된 자원들을 사용할 수 있다. CSPI에 의해 클라우드 내에서 호스팅되는 고객 네트워크들은 가상 클라우드 네트워크(virtual cloud network; VCN)들로서 지칭된다. 고객은 고객에 대해 할당된 CSPI 자원들을 사용하여 하나 이상의 가상 클라우드 네트워크(VCN)들을 셋업할 수 있다. VCN은 가상 또는 소프트웨어 정의형 사설 네트워크이다. 고객의 VCN에 배포된 고객 자원들은 컴퓨팅 인스턴스들(예를 들어, 가상 머신들, 베어-메탈 인스턴스들) 및 다른 자원들을 포함할 수 있다. 이러한 컴퓨팅 인스턴스들은 애플리케이션들, 로드 밸런서들, 데이터베이스들, 및 유사한 것과 같은 고객 워크로드들을 나타낼 수 있다. VCN 상에 배포된 컴퓨팅 인스턴스는, 인터넷과 같은 공용 네트워크를 통해, 공용 액세스가능 엔드포인트들("공용 엔드포인트들"), 동일한 VCN 또는 다른 VCN들(예를 들어, 고객의 다른 VCN들, 또는 고객에게 속하지 않은 VCN들) 내의 다른 인스턴스들, 고객의 사내 데이터 센터들 또는 네트워크들, 서비스 엔드포인트들, 및 다른 유형들의 엔드포인트들과 통신할 수 있다.

[0034] CSP는 CSPI를 사용하여 다양한 서비스들을 제공할 수 있다. 일부 경우들에서, CSPI의 고객들은 자신들이 서비스 제공자처럼 행동하고 CSPI 자원들을 사용하여 서비스들을 제공할 수 있다. 서비스 제공자는, 식별 정보(예를 들어, IP 어드레스, DNS 명칭 및 포트)에 의해 특징지어지는 서비스 엔드포인트를 노출할 수 있다. 고객의 자원(예를 들어, 컴퓨팅 인스턴스)는 해당 특정 서비스에 대해 서비스에 의해 노출된 서비스 엔드포인트에 액세스함으로써 특정 서비스를 소비할 수 있다. 이러한 서비스 엔드포인트들은 일반적으로, 인터넷과 같은 공용 통신 네트워크를 통해 엔드포인트들과 연관된 공인 IP 어드레스들을 사용하여 사용자들에 의해 공개적으로 액세스가능한 엔드포인트들이다. 공개적으로 액세스가능한 네트워크 엔드포인트들은 때때로 공용 엔드포인트들로도 지칭된다.

[0035] 특정 실시예들에서, 서비스 제공자는 서비스에 대한 엔드포인트(때때로 서비스 엔드포인트로 지칭됨)를 통해 서비스를 노출할 수 있다. 그런 다음, 서비스의 고객들은 서비스에 액세스하기 위해 이러한 서비스 엔드포인트를 사용할 수 있다. 특정 구현예들에서, 서비스에 대해 제공되는 서비스 엔드포인트는 해당 서비스를 소비하려고 하는 다수의 고객들에 의해 액세스될 수 있다. 다른 구현예들에서, 해당 고객만이 해당 전용 서비스 엔드포인트를 사용하여 서비스에 액세스할 수 있도록 전용 서비스 엔드포인트가 고객에 대해 제공될 수 있다.

[0036] 특정 실시예들에서, VCN이 생성될 때, VCN은 사설 오버레이 클래스리스 인터-도메인 라우팅(Classless Inter-Domain Routing; CIDR) 어드레스 공간과 연관되며, 이는 VCN에 할당된 사설 오버레이 IP 어드레스들의 범위이다(예를 들어, 10.0/16). VCN은 연관된 서브넷들, 루트 테이블들, 및 게이트웨이들을 포함한다. VCN은 단일 지역 내에 상주하지만 지역의 가용성 도메인들 중 하나 또는 그 이상 또는 전부에 걸쳐 있을 수 있다. 게이트웨이는, VCN에 대해 구성되고 VCN으로 그리고 이로부터 VCN 외부의 하나 이상의 엔드포인트들로의 트래픽의 통신을 가능하게 하는 가상 인터페이스이다. 하나 이상의 상이한 유형들의 게이트웨이들이 상이한 유형들의 엔드포인트들로의 그리고 이로부터의 통신을 가능하게 하기 위해 VCN에 대해 구성될 수 있다.

[0037] VCN은 하나 이상의 서브넷들과 같은 하나 이상의 서브-네트워크들로 세분될 수 있다. 따라서, 서브넷은 VCN 내에 생성될 수 있는 구성의 유닛 또는 서브디비전(subdivision)이다. VCN은 하나 또는 다수의 서브넷들을 가질 수 있다. VCN 내의 각각의 서브넷은, 해당 VCN 내의 다른 서브넷들과 중첩하지 않으며 VCN의 어드레스 공간 내의 어드레스 공간 서브넷을 나타내는 오버레이 IP 어드레스들의 연속적인 범위(예를 들어, 10.0.0.0/24 및 10.0.1.0/24)와 연관된다.

[0038] 각각의 컴퓨팅 인스턴스는, 컴퓨팅 인스턴스가 VCN의 서브넷에 참여하는 것을 가능하게 하는 가상 네트워크 인터페이스 카드(virtual network interface card; VNIC)와 연관된다. VNIC는 물리적 네트워크 인터페이스 카드(Network Interface Card; NIC)의 논리적 표현이다. 일반적으로, VNIC은 엔티티(예를 들어, 컴퓨팅 인스턴스, 서비스)와 가상 네트워크 사이의 인터페이스이다. VNIC은 서브넷에 존재하고, 하나 이상의 연관된 IP 어드레스들 및 연관된 보안 규칙들 또는 정책들을 갖는다. VNIC은 스위치 상의 계층-2 포트와 동등하다. VNIC은 컴퓨팅 인스턴스에 그리고 VCN 내의 서브넷에 접속된다. 컴퓨팅 인스턴스와 연관된 VNIC는 컴퓨팅 인스턴스가 VCN의 서브넷의 부분이 되는 것을 가능하게 하고, 컴퓨팅 인스턴스가 컴퓨팅 인스턴스와 동일한 서브넷 상에 있는 엔드포인트들, VCN 내의 상이한 서브넷들 내의 엔드포인트들, 또는 VCN 외부의 엔드포인트들과 통신하는 것(예를 들어, 패킷들을 전송하고 수신하는 것)을 가능하게 한다. 따라서, 컴퓨팅 인스턴스와 연관된 VNIC은, 컴퓨팅 인스턴스가 VNIC 내부 및 외부의 엔드포인트들과 연결하는 방법을 결정한다. 컴퓨팅 인스턴스에 대한 VNIC이 생성되며, 이는, 컴퓨팅 인스턴스가 생성되고 VCN 내의 서브넷에 추가될 때 해당 컴퓨팅 인스턴스와 연관된다. 컴퓨팅 인스턴스들의 세트를 포함하는 서브넷에 대해, 서브넷은 컴퓨팅 인스턴스들의 세트에 대응하는 VNIC들을 포함하

며, 각각의 VNIC는 컴퓨팅 인스턴스들의 세트 내의 컴퓨팅 인스턴스에 접속된다.

[0039] 각각의 컴퓨팅 인스턴스에는 컴퓨팅 인스턴스와 연관된 VNIC를 통해 사설 오버레이 IP 어드레스가 할당된다. 이러한 사설 오버레이 IP 어드레스는, 컴퓨팅 인스턴스가 생성되고 컴퓨팅 인스턴스로의 그리고 이로부터의 트래픽을 라우팅하기 위해 사용될 때 컴퓨팅 인스턴스와 연관된 VNIC에 할당된다. 주어진 서브넷 내의 모든 VNIC들은 동일한 루트 테이블, 보안 리스트들, 및 DHCP 옵션들을 사용한다. 이상에서 설명된 바와 같이, VCN 내의 각각의 서브넷은, 해당 VCN 내의 다른 서브넷들과 중첩하지 않으며 VCN의 어드레스 공간 내의 어드레스 공간 서브넷을 나타내는 오버레이 IP 어드레스들의 연속적인 범위(예를 들어, 10.0.0.0/24 및 10.0.1.0/24)와 연관된다. VCN의 특정 서브넷 상의 VNIC에 대해, VNIC에 할당된 사설 오버레이 IP 어드레스는 서브넷에 대해 할당된 오버레이 IP 어드레스들의 연속적인 범위로부터의 어드레스이다.

[0040] 특정 실시예들에서, 컴퓨팅 인스턴스에는 선택적으로, 사설 오버레이 IP 어드레스에 추가하여, 예를 들어, 공용 서브넷에 있는 경우 하나 이상의 공인 IP 어드레스들과 같은 추가적인 오버레이 IP 어드레스들이 할당될 수 있다. 이러한 다수의 어드레스들은 동일한 VNIC 상에 할당되거나 또는 컴퓨팅 인스턴스와 연관된 다수의 VNIC들에 걸쳐 할당된다. 그러나, 각각의 인스턴스는 인스턴스 런칭 동안 생성되고 인스턴스에 할당된 오버레이 사설 IP 어드레스와 연관된 1차(primary) VNIC를 가지며 - 이러한 1차 VNIC는 제거될 수 없다. 2차 VNIC들로 지칭되는 추가적인 VNIC들이 1차 VNIC와 동일한 가용성 도메인 내의 기존 인스턴스에 추가될 수 있다. 모든 VNIC들은 인스턴스와 동일한 가용성 도메인에 있다. 2차 VNIC는 1차 VNIC와 동일한 VCN 내의 서브넷에 있을 수 있거나, 또는 동일한 VCN 또는 상이한 VCN 내에 있는 상이한 서브넷에 있을 수 있다.

[0041] 컴퓨팅 인스턴스에는 선택적으로, 컴퓨팅 인스턴스가 공용 서브넷 내에 있는 경우 공인 IP 어드레스가 할당될 수 있다. 서브넷은 서브넷이 생성되는 시점에 공용 서브넷 또는 사설 서브넷으로 지정될 수 있다. 사설 서브넷은, 서브넷 내의 자원들(예를 들어, 컴퓨팅 인스턴스들) 및 연관된 VNIC들이 공인 오버레이 IP 어드레스들을 가질 수 없다는 것을 의미한다. 공용 서브넷은, 서브넷 내의 자원들 및 연관된 VNIC들이 공인 IP 어드레스들을 가질 수 있다는 것을 의미한다. 고객은 지역 또는 렐름 내의 단일 가용성 도메인에 또는 다수의 가용성 도메인들에 걸쳐 존재하도록 서브넷을 지정할 수 있다.

[0042] 이상에서 설명된 바와 같이, VCN은 하나 이상의 서브넷들로 세분될 수 있다. 특정 실시예들에서, VCN에 대해 구성된 가상 라우터(Virtual Router; VR)(VCN VR 또는 단지 VR로 지칭됨)는 VCN의 서브넷들 사이의 통신들을 가능하게 한다. VCN 내의 서브넷에 대해, VR은, 서브넷(즉, 해당 서브넷 상의 컴퓨팅 인스턴스들)이 VCN 내의 다른 서브넷들 상의 엔드포인트들 및 VCN 외부의 다른 엔드포인트들과 통신하는 것을 가능하게 하는 해당 서브넷에 대한 논리적 게이트웨이를 나타낸다. VCN VR은, VCN 내의 VNIC들과 VCN과 연관된 가상 게이트웨이들("게이트웨이들") 사이에서 트래픽을 라우팅하도록 구성된 논리적 엔티티이다. 게이트웨이들은 도 1과 관련하여 아래에서 추가로 설명된다. VCN VR은 계층-3/IP 계층 개념이다. 일 실시예에서, VCN에 대해 하나의 VCN VR이 있으며, 여기서 VCN VR은 잠재적으로 IP 어드레스들에 의해 어드레싱되는 무한한 수의 포트들을 가지고, 여기서 하나의 포트는 VCN의 각각의 서브넷에 대한 것이다. 이러한 방식으로, VCN VR는, VCN VR이 접속된 VCN 내의 각각의 서브넷에 대해 상이한 IP 어드레스를 갖는다. VR은 또한 VCN에 대해 구성된 다양한 게이트웨이들에 연결된다. 특정 실시예들에서, 서브넷에 대한 오버레이 IP 어드레스 범위로부터의 특정 오버레이 IP 어드레스는 해당 서브넷에 대한 VCN VR의 포트에 대해 예약된다. 예를 들어, 각각 연관된 어드레스 범위들 10.0/16 및 10.1/16을 갖는 2개의 서브넷들을 갖는 VCN을 고려해 본다. 어드레스 범위 10.0/16을 갖는 VCN 내의 제1 서브넷에 대해, 이러한 범위로부터의 어드레스는 해당 서브넷에 대한 VCN VR의 포트에 대해 예약된다. 일부 경우들에서, 그 범위로부터의 제1 IP 어드레스는 VCN VR에 대해 예약될 수 있다. 예를 들어, 오버레이 IP 어드레스 범위 10.0/16을 갖는 서브넷에 대해, IP 어드레스 10.0.0.1이 해당 서브넷에 대한 VCN VR의 포트에 대해 예약될 수 있다. 어드레스 범위 10.1/16를 갖는 동일한 VCN 내의 제2 서브넷에 대해, VCN VR은 IP 어드레스 10.1.0.1을 갖는 해당 제2 서브넷에 대한 포트를 가질 수 있다. VCN VR는 VCN 내의 각각의 서브넷에 대해 상이한 IP 어드레스를 갖는다.

[0043] 일부 다른 실시예들에서, VCN 내의 각각의 서브넷은, VR과 연관된 예약된 또는 디폴트 IP 어드레스를 사용하여 서브넷에 의해 어드레스가능한 그 자체의 연관된 VR을 가질 수 있다. 예약된 또는 디폴트 IP 어드레스는, 예를 들어, 해당 서브넷과 연관된 IP 어드레스들의 범위로부터의 제1 IP 어드레스일 수 있다. 서브넷 내의 VNIC들은 이러한 디폴트 또는 예약된 IP 어드레스를 사용하여 서브넷과 연관된 VR과 통신(예를 들어, 패킷들을 전송하고 수신)할 수 있다. 이러한 실시예에서, VR은 해당 서브넷에 대한 진입/진출(ingress/egress) 포인트이다. VCN 내의 서브넷과 연관된 VR은 VCN 내의 다른 서브넷들과 연관된 다른 VR들과 통신할 수 있다. VR들은 또한 VCN과 연관된 게이트웨이들과 통신할 수 있다. 서브넷에 대한 VR 기능은 서브넷 내의 VNIC에 대한 VNIC 기능성을 실행하

는 하나 이상의 NVD들 상에서 실행 중이거나 또는 이들에 의해 실행된다.

[0044] 루트 테이블들, 보안 규칙들, 및 DHCP 옵션들은 VCN에 대해 구성될 수 있다. 루트 테이블들은 VCN에 대한 가상 루트 테이블들이며, 트래픽을 VCN 내의 서브넷들로부터 특별하게 구성된 인스턴스들 또는 게이트웨이들을 통해 VCN 외부의 목적지들로 라우팅하기 위한 규칙들을 포함한다. VCN의 루트 테이블들은, 패킷들이 VCN으로 그리고 이로부터 포워딩/라우팅되는 방법을 제어하도록 맞춤화될 수 있다. DHCP 옵션들은, 인스턴스들이 부팅(boot up)될 때 인스턴스들에 자동으로 제공되는 구성 정보를 의미한다.

[0045] VCN에 대해 구성된 보안 규칙들은 VCN에 대한 오버레이 방화벽 규칙들을 나타낸다. 보안 규칙들은 진입 및 진출 규칙들을 포함하며, (예를 들어, 프로토콜 및 포트에 기초하여) VCN 내의 인스턴스들 안팎으로 허용된 트래픽의 유형들을 지정한다. 고객은, 주어진 규칙이 스테이트풀(stateful)인지 또는 스테이트리스인지 여부를 선택할 수 있다. 예를 들어, 고객은, 소스 CIDR 0.0.0.0/0, 및 목적지 TCP 포트 22를 갖는 스테이트풀 진입 규칙을 셋업함으로써 임의의 장소로부터 인스턴스들의 세트로의 인커밍 SSH 트래픽을 허용할 수 있다. 보안 규칙들은 네트워크 보안 그룹들 또는 보안 리스트들을 사용하여 구현될 수 있다. 네트워크 보안 그룹은 해당 그룹 내의 자원들에만 적용되는 보안 규칙들의 세트로 구성된다. 반면, 보안 리스트는, 보안 리스트를 사용하는 임의의 서브넷 내의 모든 자원들에 적용되는 규칙들을 포함한다. VCN에는 디폴트 보안 규칙들과 함께 디폴트 보안 리스트가 제공될 수 있다. VCN에 대해 구성된 DHCP 옵션들은, 인스턴스들이 부팅될 때 VCN 내의 인스턴스들에 자동으로 제공되는 구성 정보를 제공한다.

[0046] 특정 실시예들에서, VCN에 대한 구성 정보는 VCN 제어 평면에 의해 결정되고 저장된다. VCN에 대한 구성 정보는, 예를 들어, 다음에 관한 정보를 포함할 수 있다: VCN과 연관된 어드레스 범위, VCN 내의 서브넷들 및 연관된 정보, VCN과 연관된 하나 이상의 VR들, VCN 내의 컴퓨팅 인스턴스 및 연관된 VNIC들, VCN과 연관된 다양한 가상화 네트워크 기능들(예를 들어, VNIC들, VR들, 게이트웨이들)을 실행하는 NVD들, VCN에 대한 상태 정보, 및 다른 VCN-관련 정보. 특정 실시예들에서, VCN 배포(Distribution) 서비스는 VCN 제어 평면 또는 이의 부분들에 의해 저장된 구성 정보를 NVD들로 퍼블리싱(publish)한다. 배포된 정보는 VCN 내의 컴퓨팅 인스턴스로 그리고 이로부터 패킷들을 포워딩하기 위해 NVD들에 의해 저장되고 사용되는 정보(예를 들어, 포워딩 테이블들, 라우팅 테이블들, 등)를 업데이트하기 위해 사용될 수 있다.

[0047] 특정 실시예들에서, VCN들 및 서브넷들의 생성은 VCN 제어 평면(Control Plane; CP)에 의해 핸들링되며, 컴퓨팅 인스턴스들의 런칭은 컴퓨팅 제어 평면에 의해 핸들링된다. 컴퓨팅 제어 평면은 컴퓨팅 인스턴스에 대한 물리적 자원을 할당하는 것을 담당하고, 그런 다음 VNIC들을 생성하여 이를 컴퓨팅 인스턴스에 접속시키기 위해 VCN 제어 평면을 호출한다. VCN CP는 또한, 패킷 포워딩 및 라우팅 기능들을 수행하도록 구성된 VCN 데이터 평면으로 VCN 데이터 매핑들을 전송한다. 특정 실시예들에서, VCN CP는, VCN 데이터 평면에 업데이트들을 제공하는 것을 담당하는 배포 서비스를 제공한다. VCN 제어 평면의 예들은 또한 도 12, 도 13, 도 14 및 도 15(참조 부호 1216, 1316, 1416, 및 1516 참조)에 도시되며 이하에서 설명된다.

[0048] 고객은 CSPI에 의해 호스팅되는 자원들을 사용하여 하나 이상의 VCN들을 생성할 수 있다. 고객 VCN 상에 배포된 컴퓨팅 인스턴스는 상이한 엔드포인트들과 통신할 수 있다. 이러한 엔드포인트들은 CSPI에 의해 호스팅되는 엔드포인트들 및 CSPI 외부의 엔드포인트들을 포함할 수 있다.

[0049] CSPI 사용하여 클라우드-기반 서비스를 구현하기 위한 다양하고 상이한 아키텍처들이 도 1, 도 2, 도 3, 도 4, 도 5, 도 12, 도 13, 도 14, 및 도 15에 도시되며 이하에서 설명된다. 도 1은 특정 실시예들에 따른, CSPI에 의해 호스팅되는 오버레이 또는 고객 VCN을 보여주는 분산형 환경(100)의 하이 레벨 도면이다. 도 1에 도시된 분산형 환경은 오버레이 네트워크 내에 다수의 구성요소들을 포함한다. 도 1에 도시된 분산형 환경(100)은 단지 일 예이며, 청구되는 실시예들의 범위를 과도하게 제한하려고 의도되지 않는다. 많은 변형들, 대안들, 및 수정들이 가능하다. 예를 들어, 일부 구현예들에서, 도 1에 도시된 분산형 환경은 도 1에 도시된 것보다 더 많거나 또는 더 적은 시스템들 또는 구성요소들을 가질 수 있거나, 2개 이상의 시스템들을 결합할 수 있거나, 또는 시스템들의 상이한 구성 또는 배열을 가질 수 있다.

[0050] 도 1에 도시된 예에 도시된 바와 같이, 분산형 환경(100)은, 고객들이 가입할 수 있고 그들의 가상 클라우드 네트워크(VCN)들을 구축하기 위해 사용할 수 있는 서비스들 및 자원들을 제공하는 CSPI(101)를 포함한다. 특정 실시예들에서, CSPI(101)는 가입 고객들에게 IaaS 서비스들을 제공한다. CSPI(101) 내의 데이터 센터들은 하나 이상의 지역들로 조직될 수 있다. 하나의 예시적인 지역인 "지역 US"(102)가 도 1에 도시된다. 고객은 지역(102)에 대해 고객 VCN(104)을 구성하였다. 고객은 VCN(104) 상에 다양한 컴퓨팅 인스턴스들을 배포할 수 있으며, 여기서 컴퓨팅 인스턴스들은 가상 머신들 또는 베어 메탈 인스턴스들을 포함할 수 있다. 인스턴스들의 예들은 애

플리케이션들, 데이터베이스, 로드 밸런서들, 및 유사한 것을 포함한다.

[0051] 도 1에 도시된 실시예에서, 고객 VCN(104)는 2개의 서브넷들, 즉, "서브넷-1" 및 "서브넷-2"를 포함하며, 각각의 서브넷은 그 자체의 CIDR IP 어드레스 범위를 갖는다. 도 1에서, 서브넷-1에 대한 오버레이 IP 어드레스 범위는 10.0/16이며, 서브넷-2에 대한 어드레스 범위는 10.1/16이다. VCN 가상 라우터(105)는, VCN(104)의 서브넷들 사이의 그리고 VCN 외부의 다른 엔드포인트들과의 통신들을 가능하게 하는 VCN에 대한 논리적 게이트웨이를 나타낸다. VCN VR(105)은, VCN(104) 내의 VNIC들과 VCN(104)과 연관된 게이트웨이들 사이에서 트래픽을 라우팅하도록 구성된다. VCN VR(105)은 VCN(104)의 각각의 서브넷에 대한 포트를 제공한다. 예를 들어, VR(105)은 서브넷-1에 대한 IP 어드레스 10.0.0.1를 갖는 포트 및 서브넷-2에 대한 IP 어드레스 10.1.0.1을 갖는 포트를 제공할 수 있다.

[0052] 다수의 컴퓨팅 인스턴스들은 각각의 서브넷 상에 배포될 수 있으며, 여기서 컴퓨팅 인스턴스들은 가상 머신 인스턴스들 및/또는 베어 메탈 인스턴스들일 수 있다. 서브넷 내의 컴퓨팅 인스턴스들은 CSPI(101) 내의 하나 이상의 호스트 머신들에 의해 호스팅될 수 있다. 컴퓨팅 인스턴스는 컴퓨팅 인스턴스와 연관된 VNIC를 통해 서브넷에 참여한다. 예를 들어, 도 1에 도시된 바와 같이, 컴퓨팅 인스턴스(C1)는 컴퓨팅 인스턴스와 연관된 VNIC를 통해 서브넷-1의 부분이다. 유사하게, 컴퓨팅 인스턴스(C2)는 C2와 연관된 VNIC를 통해 서브넷-1의 부분이다. 유사한 방식으로, 가상 머신 인스턴스들 또는 베어 메탈 인스턴스들일 수 있는 다수의 컴퓨팅 인스턴스들은 서브넷-1의 부분일 수 있다. 그 연관된 VNIC를 통해, 각각의 컴퓨팅 인스턴스에는 사설 오버레이 IP 어드레스 및 MAC 어드레스가 할당된다. 예를 들어, 도 1에서, 컴퓨팅 인스턴스(C1)는 10.0.0.2의 오버레이 IP 어드레스 및 M1의 MAC 어드레스를 가지며, 반면 컴퓨팅 인스턴스(C2)는 10.0.0.3의 사설 오버레이 IP 어드레스 및 M2의 MAC 어드레스를 갖는다. 컴퓨팅 인스턴스들(C1 및 C2)을 포함하여 서브넷-1 내의 각각의 컴퓨팅 인스턴스는, 서브넷-1에 대한 VCN VR(105)의 포트에 대한 IP 어드레스인 IP 어드레스 10.0.0.1를 사용하는 VCN VR(105)에 대한 디폴트 루트를 갖는다.

[0053] 서브넷-2는, 가상 머신 인스턴스들 및/또는 베어 메탈 인스턴스들을 포함하여 그 위에 배포된 다수의 컴퓨팅 인스턴스들을 가질 수 있다. 예를 들어, 도 1에 도시된 바와 같이, 컴퓨팅 인스턴스들(D1 및 D2)은 개별적인 컴퓨팅 인스턴스들과 연관된 VNIC들을 통해 서브넷-2의 부분이다. 도 1에 도시된 실시예에서, 컴퓨팅 인스턴스(D1)는 10.1.0.2의 오버레이 IP 어드레스 및 MM1의 MAC 어드레스를 가지며, 반면 컴퓨팅 인스턴스(D2)는 10.1.0.3의 사설 오버레이 IP 어드레스 및 MM2의 MAC 어드레스를 갖는다. 컴퓨팅 인스턴스들(D1 및 D2)을 포함하여 서브넷-2 내의 각각의 컴퓨팅 인스턴스는, 서브넷-2에 대한 VCN VR(105)의 포트에 대한 IP 어드레스인 IP 어드레스 10.1.0.1를 사용하는 VCN VR(105)에 대한 디폴트 루트를 갖는다.

[0054] VCN A(104)는 또한 하나 이상의 로드 밸런서들을 포함할 수 있다. 예를 들어, 로드 밸런서는 서브넷에 대해 제공될 수 있으며, 서브넷 상의 다수의 컴퓨팅 인스턴스들에 걸쳐 트래픽을 로드 밸런싱하도록 구성될 수 있다. 로드 밸런서는 또한 VCN 내의 서브넷들에 걸쳐 트래픽을 로드 밸런싱하기 위해 제공될 수 있다.

[0055] VCN(104) 상에 배포된 특정 컴퓨팅 인스턴스는 다양하고 상이한 엔드포인트들과 통신할 수 있다. 이러한 엔드포인트들은 CSPI(200)에 의해 호스팅되는 엔드포인트들 및 CSPI(200) 외부의 엔드포인트들을 포함할 수 있다. CSPI(101)에 의해 호스팅되는 엔드포인트들을 다음을 포함할 수 있다: 특정 컴퓨팅 인스턴스와 동일한 서브넷 상의 엔드포인트(예를 들어, 서브넷-1에서 2개의 컴퓨팅 인스턴스들 사이의 통신들); 동일한 VCN 내에 있지만 상이한 서브넷 상의 엔드포인트(예를 들어, 서브넷-1 내의 컴퓨팅 인스턴스와 서브넷-2 내의 컴퓨팅 인스턴스 사이의 통신); 동일한 지역 내의 상이한 VCN 내의 엔드포인트(예를 들어, 동일한 지역(106 또는 110)) 내의 VCN 내의 엔드포인트와 서브넷-1 내의 컴퓨팅 인스턴스 사이의 통신들 또는 동일한 지역 내의 서비스 네트워크(110) 내의 엔드포인트와 서브넷-1 내의 컴퓨팅 인스턴스 사이의 통신들); 또는 상이한 지역 내의 VCN 내의 엔드포인트(예를 들어, 상이한 지역(108) 내의 VCN 내의 엔드포인트와 서브넷-1 내의 컴퓨팅 인스턴스 사이의 통신들). CSPI(101)에 의해 호스팅되는 서브넷 내의 컴퓨팅 인스턴스는 또한 CSPI(101)에 의해 호스팅되지 않는(즉, CSPI(101) 외부에 있는) 엔드포인트들과 통신할 수 있다. 이러한 외부 엔드포인트들은 고객의 사내 네트워크(116) 내의 엔드포인트들, 다른 원격 클라우드 호스팅형 네트워크들(118) 내의 엔드포인트들, 인터넷과 같은 공용 네트워크를 통해 액세스가능한 공용 엔드포인트들(114), 및 다른 엔드포인트들을 포함한다.

[0056] 동일한 서브넷 상의 컴퓨팅 인스턴스들 사이의 통신은 소스 컴퓨팅 인스턴스 및 목적지 컴퓨팅 인스턴스와 연관된 VNIC들을 사용하여 가능하게 된다. 예를 들어, 서브넷-1 내의 컴퓨팅 인스턴스(C1)는 서브넷-1 내의 컴퓨팅 인스턴스(C2)로 패킷들을 전송하기를 원할 수 있다. 소스 컴퓨팅 인스턴스에서 발원하며 그 목적지가 동일한 서브넷 내의 다른 컴퓨팅 인스턴스인 패킷에 대해, 패킷은 먼저 소스 컴퓨팅 인스턴스와 연관된 VNIC에 의해 프

로세싱된다. 소스 컴퓨팅 인스턴스와 연관된 VNIC에 의해 수행되는 프로세싱은, 패킷 헤더들로부터 패킷에 대한 목적지 정보를 결정하는 단계, 소스 컴퓨팅 인스턴스와 연관된 VNIC에 대해 구성된 임의의 정책들(예를 들어, 보안 리스트들)을 식별하는 단계, 패킷에 대한 다음 홉(hop)을 결정하는 단계, 필요에 따라 임의의 패킷 캡슐화/캡슐화해제(decapsulation) 기능들을 수행하는 단계, 그런 다음 그 의도된 목적지로의 패킷들의 통신을 가능하게 하려는 목적으로 패킷을 다음 홉으로 포워딩/라우팅하는 단계를 포함할 수 있다. 목적지 컴퓨팅 인스턴스가 소스 컴퓨팅 인스턴스와 동일한 서브넷에 있을 때, 소스 컴퓨팅 인스턴스와 연관된 VNIC은 목적지 컴퓨팅 인스턴스와 연관된 VNIC를 식별하고 패킷을 프로세싱을 위해 해당 VNIC로 포워딩하도록 구성된다. 그런 다음, 목적지 컴퓨팅 인스턴스와 연관된 VNIC가 실행되고 패킷을 목적지 컴퓨팅 인스턴스로 포워딩한다.

[0057] 서브넷 내의 컴퓨팅 인스턴스로부터 동일한 VCN 내의 상이한 서브넷 내의 엔드포인트로 통신될 패킷에 대해, 통신은 소스 및 목적지 컴퓨팅 인스턴스들과 연관된 VNIC들 및 VCN VR에 의해 가능하게 된다. 예를 들어, 도 1에서 서브넷-1 내의 컴퓨팅 인스턴스(C1)가 서브넷-2 내의 컴퓨팅 인스턴스(D1)로 패킷을 전송하기를 원하는 경우, 패킷은 먼저 컴퓨팅 인스턴스(C1)와 연관된 VNIC에 의해 프로세싱된다. 컴퓨팅 인스턴스(C1)와 연관된 VNIC는 VCN VR의 디폴트 루트 또는 포트 10.0.0.1를 사용하여 패킷을 VCN VR(105)로 라우팅하도록 구성된다. VCN VR(105)는 포트 10.1.0.1을 사용하여 패킷을 서브넷-2로 라우팅하도록 구성된다. 그런 다음, 패킷은 D1과 연관된 VNIC에 의해 수신되고 프로세싱되며, VNIC은 패킷을 컴퓨팅 인스턴스(D1)로 포워딩한다.

[0058] VCN(104) 내의 컴퓨팅 인스턴스로부터 VCN(104) 외부에 있는 엔드포인트로 통신될 패킷에 대해, 통신은 소스 컴퓨팅 인스턴스와 연관된 VNIC, VCN VR(105), 및 VCN(104)와 연관된 게이트웨이들에 의해 가능하게 된다. 하나 이상의 유형들의 게이트웨이들이 VCN(104)과 연관될 수 있다. 게이트웨이는 VCN과 다른 엔드포인트 사이의 인터페이스이며, 여기서 다른 엔드포인트는 VCN 외부에 있다. 게이트웨이는 계층-3/IP 계층 개념이며, VCN이 VCN 외부의 엔드포인트들과 통신하는 것을 가능하게 한다. 따라서, 게이트웨이는 VCN과 다른 VCN들 또는 네트워크들 사이의 트래픽 흐름을 가능하게 한다. 다양하고 상이한 유형들의 게이트웨이들이 상이한 유형들의 엔드포인트들과의 상이한 유형들의 통신들을 가능하게 하기 위해 VCN에 대해 구성될 수 있다. 게이트웨이에 의존하여, 통신들은 공용 네트워크들(예를 들어, 인터넷)을 통하거나 또는 사설 네트워크들을 통할 수 있다. 다양한 통신 프로토콜들이 이러한 통신들을 위해 사용될 수 있다.

[0059] 예를 들어, 컴퓨팅 인스턴스(C1)는 VCN(104) 외부의 엔드포인트와 통신하기를 원할 수 있다. 패킷은 먼저 소스 컴퓨팅 인스턴스(C1)와 연관된 VNIC에 의해 프로세싱될 수 있다. VNIC 프로세싱은, 패킷에 대한 목적지가 C1의 서브넷-1 외부에 있다는 것을 결정한다. C1과 연관된 VNIC는 패킷을 VCN(104)에 대한 VCN VR(105)로 포워딩할 수 있다. 그런 다음, VCN VR(105)은 패킷을 프로세싱하며, 프로세싱의 부분으로서, 패킷에 대한 목적지에 기초하여, 패킷에 대한 다음 홉으로서 VCN(104)과 연관된 특정 게이트웨이를 결정한다. 그런 다음, VCN VR(105)는 패킷을 특정 식별된 게이트웨이로 포워딩할 수 있다. 예를 들어, 목적지가 고객의 사내 네트워크 내의 엔드포인트인 경우, 패킷은 VCN(104)에 대해 구성된 동적 라우팅 게이트웨이(Dynamic Routing Gateway; DRG) 게이트웨이(122)로 VCN VR(105)에 의해 포워딩될 수 있다. 그런 다음, 패킷은 그 최종 의도된 목적지로의 패킷의 통신을 가능하게 하기 위해 게이트웨이로부터 다음 홉으로 포워딩될 수 있다.

[0060] 다양하고 상이한 유형들의 게이트웨이들은 VCN에 대해 구성될 수 있다. VCN에 대해 구성될 수 있는 게이트웨이들의 예들이 도 1에 도시되며 이하에서 설명된다. VCN과 연관된 게이트웨이들의 예들은 또한 도 12, 도 13, 도 14, 및 도 15(예를 들어, 참조 번호들 1234, 1236, 1238, 1334, 1336, 1338, 1434, 1436, 1438, 1534, 1536, 및 1538에 의해 참조되는 게이트웨이들)에 도시되며 이하에서 설명된다. 도 1에 도시된 실시예에 도시된 바와 같이, 동적 라우팅 게이트웨이(Dynamic Routing Gateway; DRG)(122)는 고객 VCN(104)에 추가되거나 또는 이와 연관될 수 있으며, 고객 VCN(104)과 다른 엔드포인트 사이의 사설 네트워크 트래픽 통신을 위한 경로를 제공하고, 여기서 다른 엔드포인트는 고객의 사내 네트워크(116), CSPI(101)의 상이한 지역 내의 VCN(108), 또는 CSPI(101)에 의해 호스팅되지 않는 다른 원격 클라우드 네트워크들(118)일 수 있다. 고객 사내 네트워크(116)는 고객의 자원들을 사용하여 구축된 고객 네트워크 또는 고객 데이터 센터일 수 있다. 고객 사내 네트워크(116)에 대한 액세스는 일반적으로 매우 제한된다. 고객 사내 네트워크(116) 및 CSPI(101)에 의해 클라우드에 배포되거나 또는 호스팅되는 하나 이상의 VCN들(104) 둘 모두를 갖는 고객에 대해, 고객은 그들의 사내 네트워크(116)와 그들의 클라우드-기반 VCN(104)이 서로 통신할 수 있는 것을 원할 수 있다. 이는, 고객이 CSPI(101)에 의해 호스팅되는 고객의 VCN(104) 및 그들의 사내 네트워크(116)를 포괄하는 확장된 하이브리드 환경을 구축하는 것을 가능하게 한다. DRG(122)는 이러한 통신을 가능하게 한다. 이러한 통신들을 가능하게 하기 위해, 통신 채널(124)이 셋업되며, 여기서 채널의 하나의 엔드포인트는 고객의 사내 네트워크(116) 내에 있고 다른 엔드포인트는 CSPI(101) 내에 있으며 고객 VCN(104)에 연결된다. 통신 채널(124)은 인터넷과 같은 공용 통신 네트워크를

통하거나 또는 사설 통신 네트워크들을 통할 수 있다. 인터넷과 같은 공용 통신 네트워크통한 IPsec VPN 기술, 공용 네트워크 대신에 사설 네트워크를 사용하는 오라클의 FastConnect 기술, 및 다른 것들과 같은 다양하고 상이한 프로토콜들이 사용될 수 있다. 통신 채널(124)에 대한 하나의 엔드 포인트를 형성하는 고객 사내 네트워크(116) 내의 디바이스 또는 장비는 도 1에 도시된 CPE(126)와 같은 고객 사내 장비(customer premise equipment; CPE)로 지칭된다. CSPI(101) 측 상에서, 엔드포인트는 DRG(122)를 실행하는 호스트 머신일 수 있다.

[0061] 특정 실시예들에서, 원격 피어링 연결(Remote Peering Connection; RPC)이 DRG에 추가될 수 있으며, 이는 고객이 하나의 VCN을 상이한 지역 내의 다른 VCN과 피어링하는 것을 가능하게 한다. 이러한 RPC를 사용하면, 고객 VCN(104)은 다른 지역 내의 VCN(108)과 연결하기 위해 DRG(122)를 사용할 수 있다. DRG(122)는 또한, 마이크로소프트 Azure 클라우드, 아마존 AWS 클라우드, 및 다른 것들과 같은 CSPI(101)에 의해 호스팅되지 않는 다른 원격 클라우드 네트워크들(118)과 통신하기 위해 사용될 수 있다.

[0062] 도 1에 도시된 바와 같이, 인터넷 게이트웨이(Internet Gateway; IGW)(120)는, VCN(104) 상의 컴퓨팅 인스턴스가 인터넷과 같은 공용 네트워크를 통해 액세스가능한 공용 엔드포인트들(114)과 통신하는 것을 가능하게 하는 고객 VCN(104)에 대해 구성될 수 있다. IGW(120)는, VCN을 인터넷과 같은 공용 네트워크에 연결하는 게이트웨이이다. IGW(120)는 VCN(104)와 같은 VCN 내의 공용 서브넷(여기서 공용 서브넷 내의 자원들은 공인 오버레이 IP 어드레스들을 가짐)이 인터넷과 같은 공용 네트워크(114) 상의 공용 엔드포인트들(112)에 직접 액세스하는 것을 가능하게 한다. IGW(120)를 사용하면, 연결들은 VCN(104) 내의 서브넷으로부터 또는 인터넷으로부터 개시될 수 있다.

[0063] 네트워크 어드레스 변환(Network Address Translation; NAT) 게이트웨이(128)는 고객의 VCN(104)에 대해 구성될 수 있으며, 전용 공인 오버레이 IP 어드레스를 갖지 않는 고객의 VCN 내의 클라우드 자원들이 인터넷에 액세스하는 것을 가능하게 하고, 이는, 이러한 자원들을 직접 인커밍 인터넷 연결들(예를 들어, L4-L7 연결들)에 노출하지 않고 그렇게 한다. 이는 인터넷 상의 공용 엔드포인트들에 대한 사설 액세스를 갖는 VCN(104) 내의 사설 서브넷-1과 같은 VCN 내의 사설 서브넷을 가능하게 한다. NAT 게이트웨이들에서, 연결들은 인터넷으로부터 사설 서브넷으로가 아니라 사설 서브넷으로부터 공용 인터넷으로만 개시될 수 있다.

[0064] 특정 실시예들에서, 서비스 게이트웨이(Service Gateway; SGW)(126)는 고객 VCN(104)에 대해 구성될 수 있으며, 서비스 네트워크(110) 내의 지원되는 서비스 엔드포인트들과 VCN(104) 사이의 사설 네트워크 트래픽에 대한 경로를 제공한다. 특정 실시예들에서, 서비스 네트워크(110)는 CSP에 의해 제공될 수 있으며 다양한 서비스들을 제공할 수 있다. 이러한 서비스 네트워크의 일 예는 오라클의 Services Network이며, 이는 고객들에 의해 사용될 수 있는 다양한 서비스들을 제공한다. 예를 들어, 고객 VCN(104)의 사설 서브넷 내의 컴퓨팅 인스턴스(예를 들어, 데이터베이스 시스템)는 공인 IP 어드레스들을 필요로 하지 않고 또는 인터넷에 액세스하지 않고 서비스 엔드포인트(예를 들어, 객체 스토리지)에 데이터를 백업할 수 있다. 특정 실시예들에서, VCN는 하나의 SGW만을 가질 수 있으며, 연결들은 서비스 네트워크(110)로부터가 아니라 오직 VCN 내의 서브넷으로부터만 개시될 수 있다. VCN가 다른 것과 피어링되는 경우, 다른 VCN 내의 자원들은 전형적으로 SGW에 액세스할 수 없다. FastConnect 또는 VPN 연결로 VCN에 연결된 사내 네트워크들 내의 자원들은 또한 해당 VCN에 대해 구성된 서비스 게이트웨이를 사용할 수 있다.

[0065] 특정 구현예들에서, SGW(126)는 서비스 클래스리스 인터-도메인 라우팅(Classless Inter-Domain Routing; CIDR) 라벨의 개념을 사용하며, 이는, 관심이 있는 서비스 또는 서비스들의 그룹에 대한 모든 지역적인 공인 IP 어드레스 범위를 나타내는 스트링이다. 고객은, 고객이 서비스에 대한 트래픽을 제어하기 위해 SGW 및 관련된 라우팅 규칙들을 구성할 때 서비스 CIDR 라벨을 사용한다. 고객은 선택적으로, 서비스의 공인 IP 어드레스들이 향후 변경되는 경우 이들을 조정할 필요 없이 보안 규칙들을 구성할 때 이를 사용할 수 있다.

[0066] 로컬 피어링 게이트웨이(Local Peering Gateway; LPG)(132)는, 고객 VCN(104)에 추가될 수 있으며 VCN(104)이 동일한 지역 내의 다른 VCN과 피어링하는 것을 가능하게 하는 게이트웨이이다. 피어링은, VCN들이, 트래픽이 인터넷과 같은 공용 네트워크를 통과(traverse)하지 않고 또는 고객의 사내 네트워크(116)를 통해 트래픽을 라우팅하지 않고 사설 IP 어드레스들을 사용하여 통신한다는 것을 의미한다. 선호되는 실시예들에서, VCN은 VCN이 설정한 각각의 피어링에 대해 별도의 LPG를 갖는다. 로컬 피어링 또는 VCN 피어링은 상이한 애플리케이션들 또는 인프라스트럭처 관리 기능들 사이에 네트워크 연결성을 설정하기 위해 사용되는 일반적인 관행이다.

[0067] 서비스 네트워크(110) 내의 서비스들의 제공자들과 같은 서비스 제공자들은 상이한 액세스 모델들을 사용하여 서비스들에 대한 액세스를 제공할 수 있다. 공용 액세스 모델에 따르면, 서비스들은 인터넷과 같은 공용 네트워

크를 통해 고객 VCN 내의 컴퓨팅 인스턴스에 의해 공개적으로 액세스가능한 공용 엔드포인트들로서 노출될 수 있거나 또는 SGW(126)에 의해 사적으로(privately) 액세스가능할 수 있다. 특정 사설 액세스 모델에 따르면, 서비스들은 고객의 VCN 내의 사설 서브넷 내의 사설 IP 엔드포인트들로서 액세스가능하게 만들어진다. 이는 사설 엔드포인트(Private Endpoint; PE) 액세스로 지칭되며, 서비스 제공자가 그들의 서비스를 고객의 사설 네트워크 내의 인스턴스로서 노출하는 것을 가능하게 한다. 사설 엔드포인트 자원은 고객의 VCN 내의 서비스를 나타낸다. 각각의 PE는 고객의 VCN에서 고객에 의해 선택된 서브넷 내의 VNIC(하나 이상의 사설 IP들을 갖는, PE-VNIC로 지칭됨)로서 매니페스트(manifest)된다. 따라서, PE는 VNIC를 사용하여 사설 고객 VCN 서브넷 내에서 서비스를 제공하기 위한 방법을 제공한다. 엔드포인트가 VNIC로서 노출되기 때문에, 라우팅 규칙들, 보안 리스트들, 등과 같은 VNIC와 연관된 모든 특징들은 이제 PE VNIC에 대해 이용가능하다.

[0068] 서비스 제공자는 PE를 통한 액세스를 가능하게 하기 위해 그들의 서비스를 등록할 수 있다. 제공자는, 고객의 테넌시들에 대한 서비스들의 가시성을 제한하는 정책들을 서비스와 연관시킬 수 있다. 제공자는, 특히 다중-테넌트 서비스들에 대해, 단일 가상 IP 어드레스(virtual IP address; VIP) 하에서 다수의 서비스들을 등록할 수 있다. 동일한 서비스를 나타내는 (다수의 VCN들 내의) 다수의 이러한 사설 엔드포인트들이 있을 수 있다.

[0069] 그러면, 사설 서브넷 내의 컴퓨팅 인스턴스들은 서비스에 액세스하기 위해 PE VNIC의 사설 IP 어드레스 또는 서비스 DNS 명칭을 사용할 수 있다. 고객 VCN 내의 컴퓨팅 인스턴스들은 고객 VCN 내의 PE의 사설 IP 어드레스로 트래픽을 전송함으로써 서비스에 액세스할 수 있다. 사설 액세스 게이트웨이(Private Access Gateway; PAGW)(130)는, 고객 서브넷 사설 엔드포인트들로부터의/이로의 모든 트래픽에 대한 진입/진출 포인트로서 역할하는 서비스 제공자 VCN(예를 들어, 서비스 네트워크(110) 내의 VCN)에 접속될 수 있는 게이트웨이 자원이다. PAGW(130)는 제공자가 그 내부 IP 어드레스 자원들을 사용하지 않고 PE 연결들의 수를 스케일링하는 것을 가능하게 한다. 제공자는 단일 VCN에 등록된 임의의 수의 서비스들에 대해 하나의 PAGW만을 구성하면 된다. 제공자들은 1명 이상의 고객들의 다수의 VCN들 내의 사설 엔드포인트로서 서비스를 표현할 수 있다. 고객의 관점으로부터, PE VNIC는, 고객의 인스턴트에 접속되는 대신에, 고객이 상호작용하기를 희망하는 서비스에 접속되는 것을 나타난다. 사설 엔드포인트로 향하는 트래픽은 PAGW(130)를 통해 서비스로 라우팅된다. 이들은 고객-대-서비스 사설 연결들(C2S(customer-to-service) 연결들)로 지칭된다.

[0070] PE 개념은 또한, 트래픽이 FastConnect/IPsec 링크들 및 고객 VCN 내의 사설 엔드포인트를 통해 흐르는 것을 가능하게 함으로써 서비스에 대한 액세스를 고객의 사내 네트워크들 및 데이터 센터들로 확장하기 위해 사용될 수 있다. 서비스에 대한 사설 액세스는 또한, 트래픽이 고객의 VCN 내의 PE와 LPG(132) 사이에서 흐르는 것을 가능하게 함으로써 고객의 피어링된 VCN들로 확장될 수 있다.

[0071] 고객은 서브넷 레벨로 VCN에서의 라우팅을 제어할 수 있으며, 따라서 고객은 VCN(104)과 같은 고객의 VCN에서 어떤 서브넷들이 각각의 게이트웨이를 사용할지를 지정할 수 있다. VCN의 루트 테이블들은, 특정 게이트웨이를 통해 VCN 밖으로 향하는 트래픽이 허용되는지 여부를 결정하기 위해 사용된다. 예를 들어, 특정 경우에서, 고객 VCN(104) 내의 공용 서브넷에 대한 루트 테이블은 IGW(120)를 통해 비-로컬 트래픽을 전송할 수 있다. 동일한 고객 VCN(104) 내의 사설 서브넷에 대한 루트 테이블은 SGW(126)을 통해 CSP 서비스들로 향하는 트래픽을 전송할 수 있다. 모든 남아 있는 트래픽은 NAT 게이트웨이(128)를 통해 전송될 수 있다. 루트 테이블들은 VCN 밖으로 나가는 트래픽만을 제어한다.

[0072] VCN과 연관된 보안 리스트들은 인바운드(inbound) 연결들을 통해서 게이트웨이를 통해 VCN 내로 들어오는 트래픽을 제어하기 위해 사용된다. 서브넷 내의 모든 자원들은 동일한 루트 테이블 및 보안 리스트들을 사용한다. 보안 리스트들은 VCN의 서브넷 내의 인스턴스들 안팎으로 허용된 특정 유형들의 트래픽을 제어하기 위해 사용될 수 있다. 보안 리스트 규칙들은 진입(인바운드) 및 진출(아웃바운드) 규칙들을 포함할 수 있다. 예를 들어, 진입 규칙은 허용된 소스 어드레스 범위를 지정할 수 있으며, 반면 진출 규칙은 허용된 목적지 어드레스 범위를 지정할 수 있다. 보안 규칙들은 특정 프로토콜(예를 들어, TCP, ICMP), 특정 포트(예를 들어, SSH에 대해 22, 윈도우즈 RDP에 대해 3389), 등을 지정할 수 있다. 특정 구현예들에서, 인스턴스의 운영 체제는, 보안 리스트 규칙들과 정렬된 그 자체의 방화벽 규칙들을 시행할 수 있다. 규칙들은 스테이트풀(예를 들어, 연결이 추적되며, 응답은 응답 트래픽에 대한 명시적인 보안 리스트 규칙 없이 자동으로 허용됨) 또는 스테이트리스일 수 있다.

[0073] 고객 VCN으로부터의 액세스(즉, VCN(104) 상에 배포된 자원 또는 컴퓨팅 인스턴스에 의한 액세스)는 공용 액세스, 사설 액세스, 또는 전용 액세스로 분류될 수 있다. 공용 액세스는, 공인 IP 어드레스 또는 NAT가 공용 엔드포인트에 액세스하기 위해 사용되는 액세스 모델을 의미한다. 사설 액세스는 사설 IP 어드레스들을 갖는

VCN(104) 내의 고객 워크로드들(예를 들어, 사설 서브넷 내의 자원들)이 인터넷과 같은 공용 네트워크를 통과하지 않고 서비스들에 액세스하는 것을 가능하게 한다. 특정 실시예들에서, CSPI(101)는 사설 IP 어드레스들을 갖는 고객 VCN 워크로드들이 서비스 게이트웨이를 사용하여 서비스들(이의 공용 서비스 엔드포인트들)에 액세스하는 것을 가능하게 한다. 따라서, 서비스 게이트웨이는, 고객의 VCN과 고객의 사설 네트워크 외부에 상주하는 서비스의 공용 엔드포인트 사이에 가상 링크를 설정함으로써 사설 액세스 모델을 제공한다.

[0074] 추가적으로, CSPI는 FastConnect 공용 피어링과 같은 기술들을 사용하여 전용 공용 액세스를 제공할 수 있으며, 여기서 고객 사내 인스턴스들은 인터넷과 같은 공용 네트워크를 통하지 않고 FastConnect 연결을 사용하여 고객 VCN 내의 하나 이상의 서비스들에 액세스할 수 있다. CSPI는 또한 FastConnect 사설 피어링을 사용하는 전용 사설 액세스를 제공할 수 있으며, 여기서 사설 IP 어드레스들을 갖는 고객 사내 인스턴스들은 FastConnect 연결을 사용하여 고객의 VCN 워크로드들에 액세스할 수 있다. FastConnect는 고객의 사내 네트워크를 CSPI 및 그 서비스들에 연결하기 위해 공용 인터넷을 사용하는 것에 대안적인 네트워크 연결성이다. FastConnect는, 인터넷-기반 연결들과 비교할 때 더 높은 대역폭 옵션들 및 더 신뢰할 수 있고 일관된 네트워킹 경험을 갖는 전용 사설 연결을 생성하기 위한 쉽고, 탄력적이며, 경제적인 방식을 제공한다.

[0075] 도 1 및 이상의 첨부된 설명은 예시적인 가상 네트워크 내의 다양한 가상화된 구성요소들을 설명한다. 이상에서 설명된 바와 같이, 가상 네트워크는 기초 물리적 또는 서브스트레이트 네트워크 상에 구축된다. 도 2는 특정 실시예들에 따른, CSPI(200) 내의 물리적 네트워크 내의 물리적 구성요소들의 단순화된 아키텍처 도면을 도시한다. 도시된 바와 같이, CSPI(200)는 클라우드 서비스 제공자(cloud service provider; CSP)에 의해 제공되는 구성요소들 및 자원들(예를 들어, 컴퓨팅, 메모리, 및 네트워킹 자원들)을 포함하는 분산형 환경을 제공한다. 이러한 구성요소들 및 자원들은 가입 고객들, 즉, CSP에 의해 제공되는 하나 이상의 서비스들에 가입한 고객들에게 클라우드 서비스들(예를 들어, IaaS 서비스들)을 제공하기 위해 사용된다. 고객이 가입한 서비스들에 기초하여, CSPI(200)의 자원들(예를 들어, 컴퓨팅, 메모리, 및 네트워킹 자원들)의 서브세트가 고객에 대해 프로비저닝된다. 그러면, 고객들은 CSPI(200)에 의해 제공되는 컴퓨팅, 메모리, 및 네트워킹 자원들을 사용하여 그들 자체의 클라우드-기반(즉, CSPI-호스팅형) 맞춤화가능 사설 가상 네트워크들을 구축할 수 있다. 이상에서 표시된 바와 같이, 이러한 고객 네트워크들은 가상 클라우드 네트워크(virtual cloud network; VCN)들로 지칭된다. 고객은 이러한 고객 VCN들 상에 컴퓨팅 인스턴스들과 같은 하나 이상의 고객 자원들을 배포할 수 있다. 컴퓨팅 인스턴스들은 가상 머신들, 베어 메탈 인스턴스들, 및 유사한 것의 형태일 수 있다. CSPI(200)는, 고객들이 고도로 이용가능한 호스팅되는 환경에서 광범위한 애플리케이션 및 서비스들을 구축하고 실행하는 것을 가능하게 하는 상보적인 클라우드 서비스들의 세트 및 인프라스트럭처를 제공한다.

[0076] 도 2에 도시된 예시적인 실시예에서, CSPI(200)의 물리적 구성요소들은 하나 이상의 물리적 호스트 머신들 또는 물리적 서버들(예를 들어, 202, 206, 208), 네트워크 가상화 디바이스(network virtualization device; NVD)들(예를 들어, 210, 212), 탑-오브-랙(top-of-rack; TOR) 스위치들(예를 들어, 214, 216), 및 물리적 네트워크(예를 들어, 218), 및 물리적 네트워크(218) 내의 스위치들을 포함한다. 물리적 호스트 머신들 또는 서버들은 VCN의 하나 이상의 서브넷들에 참여하는 다양한 컴퓨팅 인스턴스들을 호스팅하고 실행할 수 있다. 컴퓨팅 인스턴스들은 가상 머신 인스턴스들 및 베어 메탈 인스턴스들을 포함할 수 있다. 예를 들어, 도 1에 도시된 다양한 컴퓨팅 인스턴스들은 도 2에 도시된 물리적 호스트 머신들에 의해 호스팅될 수 있다. VCN 내의 가상 머신 컴퓨팅 인스턴스들은 하나의 호스트 머신에 의해 또는 다수의 상이한 호스트 머신들에 의해 실행될 수 있다. 물리적 호스트 머신들은 또한 가상 호스트 머신들, 컨테이너-기반 호스트들 또는 기능들, 및 유사한 것을 호스팅할 수 있다. 도 1에 도시된 VNIC들 및 VCN VR은 도 2에 도시된 NVD들에 의해 실행될 수 있다. 도 1에 도시된 게이트웨이들은 도 2에 도시된 호스트 머신에 의해 및/또는 NVD들에 의해 실행될 수 있다.

[0077] 호스트 머신들 또는 서버들은, 호스트 머신들 상에 가상화된 환경을 생성하고 인에이블하는 하이퍼바이저(가상 머신 모니터 또는 VMM으로 지칭됨)를 실행할 수 있다. 가상화 또는 가상화된 환경은 클라우드-기반 컴퓨팅을 가능하게 한다. 하나 이상의 컴퓨팅 인스턴스들은 호스트 머신 상의 하이퍼바이저에 의해 해당 호스트 머신 상에 생성되고, 실행되며, 관리될 수 있다. 호스트 머신 상의 하이퍼바이저는, 호스트 머신의 물리적 컴퓨팅 자원들(예를 들어, 컴퓨팅, 메모리, 및 네트워킹 자원들)이 호스트 머신에 의해 실행되는 다양한 컴퓨팅 인스턴스들 사이에 공유되는 것을 가능하게 한다.

[0078] 예를 들어, 도 2에 도시된 바와 같이, 호스트 머신들(202 및 208)은 각각 하이퍼바이저들(260 및 266)을 실행한다. 이러한 하이퍼바이저들은 소프트웨어, 펌웨어, 또는 하드웨어, 또는 이들의 조합들을 사용하여 구현될 수 있다. 전형적으로, 하이퍼바이저는 호스트 머신의 운영 체제(operating system; OS)의 상단 상에 위치하며 결과적으로 호스트 머신의 하드웨어 프로세서 상에서 실행되는 프로세스 또는 소프트웨어 계층이다.

하이퍼바이저는, 호스트 머신의 물리적 컴퓨팅 자원들(예를 들어, 프로세서들/코어들과 같은 프로세싱 자원들, 메모리 자원들, 네트워킹 자원들)이 호스트 머신에 의해 실행되는 다양한 가상 머신 컴퓨팅 인스턴스들 사이에 공유되는 것을 가능하게 함으로써 가상화된 환경을 제공한다. 예를 들어, 도 2에서, 하이퍼바이저(260)는 호스트 머신(202)의 OS의 상단 상에 위치할 수 있으며, 호스트 머신(202)의 컴퓨팅 자원들(예를 들어, 프로세싱, 메모리, 및 네트워킹 자원들)이 호스트 머신(202)에 의해 실행되는 컴퓨팅 인스턴스들(예를 들어, 가상 머신들) 사이에서 공유되는 것을 가능하게 한다. 가상 머신은 그 자체의 운영 체제(게스트 운영 체제로 지칭됨)를 가질 수 있으며, 이는 호스트 머신의 OS와 동일하거나 또는 상이할 수 있다. 호스트 머신에 의해 실행되는 가상 머신의 운영 체제는 동일한 호스트 머신에 의해 실행되는 다른 가상 머신의 운영 체제와 동일하거나 또는 상이할 수 있다. 따라서, 하이퍼바이저는 다수의 운영 체제가 호스트 머신의 동일한 컴퓨팅 자원들을 공유하면서 서로 나란히 실행되는 것을 가능하게 한다. 도 2에 도시된 호스트 머신들은 동일하거나 또는 상이한 유형들의 하이퍼바이저들을 가질 수 있다.

[0079] 컴퓨팅 인스턴스는 가상 머신 인스턴스 또는 베어 메탈 인스턴스일 수 있다. 도 2에서, 호스트 머신(202) 상의 컴퓨팅 인스턴스(268) 및 호스트 머신(208) 상의 컴퓨팅 인스턴스(274)는 가상 머신 인스턴스들의 예들이다. 호스트 머신(206)은 고객에게 제공되는 베어 메탈 인스턴스의 일 예이다.

[0080] 특정 경우들에서, 전체 호스트 머신이 단일 고객에게 프로비저닝될 수 있으며, 해당 호스트 머신에 의해 호스팅되는 하나 이상의 컴퓨팅 인스턴스들(가상 머신들 또는 베어 메탈 인스턴스) 모두는 해당되는 동일한 고객에 속한다. 다른 경우들에서, 호스트 머신은 다수의 고객들(즉, 다수의 테넌트들) 사이에서 공유될 수 있다. 이러한 다중-테넌시 시나리오에서, 호스트 머신은 상이한 고객들에 속하는 가상 머신 컴퓨팅 인스턴스들을 호스팅할 수 있다. 이러한 컴퓨팅 인스턴스들은 상이한 고객들의 상이한 VCN들이 멤버들일 수 있다. 특정 실시예들에서, 베어 메탈 컴퓨팅 인스턴스는 하이퍼바이저 없이 베어 메탈 서버에 의해 호스팅된다. 베어 메탈 컴퓨팅 인스턴스가 프로비저닝될 때, 단일 고객 또는 테넌트는 베어 메탈 인스턴스를 호스팅하는 호스트 머신의 물리적 CPU, 메모리, 및 네트워크 인터페이스들의 제어를 유지하며, 호스트 머신은 다른 고객들 또는 테넌트들과 공유되지 않는다.

[0081] 이상에서 설명된 바와 같이, VCN의 부분인 각각의 컴퓨팅 인스턴스는, 컴퓨팅 인스턴스가 VCN의 서브넷의 멤버가 되는 것을 가능하게 하는 VNIC와 연관된다. 컴퓨팅 인스턴스와 연관된 VNIC는 컴퓨팅 인스턴스로의 그리고 이로부터의 패킷들 또는 프레임들의 통신을 가능하게 한다. VNIC, 컴퓨팅 인스턴스가 생성될 때 컴퓨팅 인스턴스와 연관된다. 특정 실시예들에서, 호스트 머신에 의해 실행되는 컴퓨팅 인스턴스에 대해, 해당 컴퓨팅 인스턴스와 연관된 VNIC는 호스트 머신에 연결된 NVD에 의해 실행된다. 예를 들어, 도 2에서, 호스트 머신(202)은 VNIC(276)와 연관된 가상 머신 컴퓨팅 인스턴스(268)을 실행하며, VNIC(276)은 호스트 머신(202)에 연결된 NVD(210)에 의해 실행된다. 다른 예로서, 호스트 머신(206)에 의해 호스팅되는 베어 메탈 인스턴스(272)는, 호스트 머신(206)에 연결된 NVD(212)에 의해 실행되는 VNIC(280)와 연관된다. 또 다른 예로서, VNIC(284)는 호스트 머신(208)에 의해 실행되는 컴퓨팅 인스턴스(274)와 연관되며, VNIC(284)는 호스트 머신(208)에 연결된 NVD(212)에 의해 실행된다.

[0082] 호스트 머신에 의해 호스팅되는 컴퓨팅 인스턴스들에 대해, 호스트 머신에 연결된 NVD는 또한 컴퓨팅 인스턴스들이 이의 멤버들인 VCN들에 대응하는 VCN VR들을 실행한다. 예를 들어, 도 2에 도시된 실시예에서, NVD(210)는 컴퓨팅 인스턴스(268)가 이의 멤버인 VCN에 대응하는 VCN VR(277)을 실행한다. NVD(212)는 또한 호스트 머신들(206 및 208)에 의해 호스팅되는 컴퓨팅 인스턴스들에 대응하는 VCN들에 대응하는 하나 이상의 VCN VR들(283)을 실행할 수 있다.

[0083] 호스트 머신은, 호스트 머신이 다른 디바이스들에 연결되는 것을 가능하게 하는 하나 이상의 네트워크 인터페이스 카드(network interface card; NIC)들을 포함할 수 있다. 호스트 머신 상의 NIC는, 호스트 머신이 다른 디바이스에 통신가능하게 연결되는 것을 가능하게 하는 하나 이상의 포트들(또는 인터페이스들)을 제공할 수 있다. 예를 들어, 호스트 머신은 호스트 머신 상에 그리고 NVD 상에 제공된 하나 이상의 포트들(또는 인터페이스들)을 사용하여 NVD에 연결될 수 있다. 호스트 머신은 또한 다른 호스트 머신과 같은 다른 디바이스들에 연결될 수 있다.

[0084] 예를 들어, 도 2에서, 호스트 머신(202)은, 호스트 머신(202)의 NIC(232)에 의해 제공된 포트(234) 사이에서 그리고 NVD(210)의 포트(236) 사이에 연장되는 링크(220)를 사용하여 NVD(210)에 연결된다. 호스트 머신(206)은, 호스트 머신(206)의 NIC(244)에 의해 제공된 포트(246) 사이에서 그리고 NVD(212)의 포트(248) 사이에 연장되는 링크(224)를 사용하여 NVD(212)에 연결된다. 호스트 머신(208)은, 호스트 머신(208)의 NIC(250)에 의해 제공된

포트(252) 사이에서 그리고 NVD(212)의 포트(254) 사이에 연장되는 링크(226)를 사용하여 NVD(212)에 연결된다.

[0085] NVD들은 결과적으로 통신 링크들을 통해 탑-오브-랙(top-of-the-rack; TOR) 스위치들에 연결되며, 이들은 물리적 네트워크(218)(스위치 패브릭(switch fabric)으로도 지칭됨)에 연결된다. 특정 실시예들에서, 호스트 머신과 NVD 사이의 그리고 NVD와 TOR 스위치 사이의 링크들은 이더넷 링크들이다. 예를 들어, 도 2에서, NVD들(210 및 212)은 각각 링크들(228 및 230)을 사용하여 TOR 스위치들(214 및 216)에 연결된다. 특정 실시예들에서, 링크들(220, 224, 226, 228, 및 230)은 이더넷 링크들이다. TOR에 연결된 NVD들 및 호스트 머신들의 모음은 때때로 랙(rack)으로 지칭된다.

[0086] 물리적 네트워크(218)는, TOR 스위치들이 서로 통신하는 것을 가능하게 하는 통신 패브릭을 제공한다. 물리적 네트워크(218)는 다중-계층형 네트워크일 수 있다. 특정 구현예들에서, 물리적 네트워크(218)는 스위치들의 다중-계층형 클로스 네트워크(Clos network)이며, 여기서 TOR 스위치들(214 및 216)은 다중-계층형 및 다중-노드 물리적 스위칭 네트워크(218)의 리프(leaf) 레벨 노드들을 나타낸다. 비제한적으로 2-계층 네트워크, 3-계층 네트워크, 4-계층 네트워크, 5-계층 네트워크, 및 일반적으로 "n"-계층 네트워크를 포함하여 상이한 클로스 네트워크 구성들이 가능하다. 클로스 네트워크의 일 예가 도 5에 도시되고 이하에서 설명된다.

[0087] 1-대-1 구성, 다-대-1 구성, 1-대-다 구성, 및 다른 것들과 같은 다양하고 상이한 연결 구성들이 호스트 머신들과 NVD들 사이에서 가능하다. 1-대-1 구성 구현예에서, 각각의 호스트 머신은 그 자체의 별도의 NVD에 연결된다. 예를 들어, 도 2에서, 호스트 머신(202)은 호스트 머신(202)의 NIC(232)를 통해 NVD(210)에 연결된다. 다-대-1 구성에서, 다수의 호스트 머신들은 하나의 NVD에 연결된다. 예를 들어, 도 2에서, 호스트 머신들(206 및 208)은 각각 NIC들(244 및 250)을 통해 동일한 NVD(212)에 연결된다.

[0088] 1-대-다 구성에서, 하나의 호스트 머신은 다수의 NVD들에 연결된다. 도 3은, 호스트 머신이 다수의 NVD들에 연결되는 CSPI(300) 내의 일 예를 도시한다. 도 3에 도시된 바와 같이, 호스트 머신(302)은 다수의 포트들(306 및 308)을 포함하는 네트워크 인터페이스 카드(network interface card; NIC)(304)를 포함한다. 호스트 머신(300)은 포트(306) 및 링크(320)를 통해 제1 NVD(310)에 연결되고, 포트(308) 및 링크(322)를 통해 제2 NVD(312)에 연결된다. 포트들(306 및 308)은 이더넷 포트들이며, 호스트 머신(302)과 NVD들(310 및 312) 사이의 링크들(320 및 322)은 이더넷 링크들일 수 있다. NVD(310)는 결과적으로 제1 TOR 스위치(314)에 연결되고, NVD(312)는 제2 TOR 스위치(316)에 연결된다. NVD들(310 및 312)과 TOR 스위치들(314 및 316) 사이의 링크들은 이더넷 링크들일 수 있다. TOR 스위치들(314 및 316)은 다중-계층형 물리적 네트워크(318) 내의 계층-0 스위칭 디바이스들을 나타낸다.

[0089] 도 3에 도시된 배열은 물리적 스위치 네트워크(318)로부터 그리고 이로의 호스트 머신(302)으로의 그리고 이로부터의 2개의 별도의 물리적 네트워크 경로들을 제공한다: TOR 스위치(314)를 통과하여 NVD(310)를 지나 호스트 머신(302)으로의 제1 경로 및 TOR 스위치(316)를 통과하여 NVD(312)를 지나 호스트 머신(302)으로의 제2 경로. 별도의 경로들은 호스트 머신(302)의 향상된 가용성(고 가용성으로 지칭됨)을 위해 제공된다. 경로들 중 하나에 문제가 있는 경우(예를 들어, 경로들 중 하나의 경로의 링크가 다운된 경우) 또는 디바이스들 중 하나에 문제가 있는 경우(예를 들어, 특정 NVD가 기능하지 않는 경우), 다른 경로가 호스트 머신(302)으로의/으로부터의 통신들을 위해 사용될 수 있다.

[0090] 도 3에 도시된 구성에서, 호스트 머신은 호스트 머신의 NIC에 의해 제공된 2개의 상한 포트들을 사용하여 2개의 상이한 NVD들에 연결된다. 다른 실시예들에서, 호스트 머신은, 다수의 NVD들에 대한 호스트 머신의 연결성을 가능하게 하기 위해 다수의 NIC들을 포함할 수 있다.

[0091] 다시 도 2를 참조하면, NVD는, 하나 이상의 네트워크 및/또는 스토리지 가상화 기능들을 수행하는 물리적 디바이스 또는 구성요소이다. NVD는 하나 이상의 프로세싱 유닛들(예를 들어, CPU들, 네트워크 프로세싱 유닛(Network Processing Unit; NPU)들, FPGA들, 패킷 프로세싱 파이프라인들, 등), 캐시를 포함하는 메모리, 및 포트들을 갖는 임의의 디바이스일 수 있다. 다양한 가상화 기능들은 NVD의 하나 이상의 프로세싱 유닛들에 의해 실행되는 소프트웨어/펌웨어에 의해 수행될 수 있다.

[0092] NVD는 다양하고 상이한 형태들로 구현될 수 있다. 예를 들어, 특정 실시예들에서, NVD는 내장형 프로세서 온보드를 갖는 지능형 NIC 또는 smartNIC로 지칭되는 인터페이스 카드로서 구현된다. smartNIC는 호스트 머신들 상의 NIC들과는 별도의 디바이스이다. 도 2에서, NVD들(210 및 212)은 각각, 호스트 머신들(202), 및 호스트 머신들(206 및 208)에 연결된 smartNIC들로서 구현될 수 있다.

[0093] 그러나, smartNIC는 단지 NVD 구현의 일 예일 뿐이다. 다른 구현예들이 가능하다. 예를 들어, 일부 다른 구현예

들에서, NVD 또는 NVD에 의해 수행되는 하나 이상의 기능들은 하나 이상의 호스트 머신들, 하나 이상의 TOR 스위치들, 및 CSPI(200)에 통합되거나 또는 이들에 의해 수행될 수 있다. 예를 들어, NVD는 호스트 머신에 구현될 수 있으며, 여기서 NVD에 의해 수행되는 기능들은 호스트 머신에 의해 수행된다. 다른 예로서, NVD는 TOR 스위치의 부분일 수 있거나, 또는 TOR 스위치는, TOR 스위치가 공용 클라우드에 대해 사용되는 다양하고 복잡한 패킷 변환(transformation)들을 수행하는 것을 가능하게 하는 NVD에 의해 수행되는 기능들을 수행하도록 구성될 수 있다. NVD의 기능들을 수행하는 TOR은 때때로 스마트 TOR로 지칭된다. 베어 메탈(bare metal; BM) 인스턴스들이 아니라 가상 머신(virtual machine; VM) 인스턴스들이 고객들에게 제공되는 또 다른 구현예들에서, NVD에 의해 수행되는 기능들은 호스트 머신의 하이퍼바이저 내부에 구현될 수 있다. 일부 다른 구현예들에서, NVD의 기능들 중 일부는 호스트 머신들의 플리트 상에서 실행 중인 중앙 집중식 서비스로 오프로딩될 수 있다.

[0094] 특정 실시예들에서, 예컨대 도 2에 도시된 바와 같은 smartNIC로서 구현될 때, NVD는, NVD가 하나 이상의 호스트 머신들에 그리고 하나 이상의 TOR 스위치들에 연결되는 것을 가능하게 하는 다수의 물리적 포트들을 포함할 수 있다. NVD 상의 포트는 호스트-방향(host-facing) 포트("사우스 포트(south port)"로도 지칭됨) 또는 네트워크-방향 또는 TOR-방향 포트("노스 포트(north port)"로도 지칭됨)로서 분류될 수 있다. NVD의 호스트-방향 포트는 NVD를 호스트 머신에 연결하기 위해 사용되는 포트이다. 도 2의 호스트-방향 포트들의 예들은 NVD(210) 상의 포트(236), 및 NVD(212) 상의 포트들(248 및 254)을 포함한다. NVD의 네트워크-방향 포트는 NVD를 TOR 스위치에 연결하기 위해 사용되는 포트이다. 도 2의 네트워크-방향 포트들의 예들은 NVD(210) 상의 포트(256), 및 NVD(212) 상의 포트(258)를 포함한다. 도 2에 도시된 바와 같이, NVD(210)는, NVD(210)의 포트(256)로부터 TOR 스위치(214)까지 연장되는 링크(228)를 사용하여 TOR 스위치(214)에 연결된다. 마찬가지로, NVD(212)는, NVD(212)의 포트(258)로부터 TOR 스위치(216)까지 연장되는 링크(230)를 사용하여 TOR 스위치(216)에 연결된다.

[0095] NVD는 호스트-방향 포트를 통해 호스트 머신으로부터 프레임들 및 패킷들(예를 들어, 호스트 머신에 의해 호스팅되는 컴퓨팅 인스턴스에 의해 생성된 패킷들 및 프레임들)을 수신하며, 필수 패킷 프로세싱을 수행한 이후에, 패킷들 및 프레임들을 NVD의 네트워크-방향 포트를 통해 TOR 스위치로 포워딩할 수 있다. NVD는 NVD의 네트워크-방향 포트를 통해 TOR 스위치로부터 패킷들 및 프레임들을 수신할 수 있으며, 필수 패킷 프로세싱을 수행한 이후에, 패킷들 및 프레임들을 NVD의 호스트-방향 포트를 통해 호스트 머신으로 포워딩할 수 있다.

[0096] 특정 실시예들에서, NVD와 TOR 스위치 사이에 다수의 포트들 및 연관된 링크들이 있을 수 있다. 이러한 포트들 및 링크들은 다수의 포트들 및 링크들의 링크 집성기 그룹(LAG(link aggregator group)로 지칭됨)을 형성하기 위해 집성(aggregate)될 수 있다. 링크 집성은, 2개의 엔드-포인트들 사이의(예를 들어, NVD와 TOR 스위치 사이의) 다수의 물리적 링크들이 단일 논리적 링크로서 취급되는 것을 가능하게 한다. 주어진 LAG 내의 모든 물리적 링크들은 동일한 속도로 풀-듀플렉스(full-duplex) 모드로 동작할 수 있다. LAG들은 2개의 엔드포인트들 사이의 연결의 대역폭 및 신뢰성을 증가시키는 것을 돕는다. LAG 내의 물리적 링크들 중 하나가 다운되는 경우, 트래픽은 LAG 내의 다른 물리적 링크들 중 하나로 동적으로 그리고 투명하게 재할당된다. 집성된 물리적 링크들을 각각의 개별적인 링크보다 더 높은 대역폭을 공급한다. LAG와 연관된 다수의 포트들은 단일 논리적 포트로서 취급된다. 트래픽은 LAG의 다수의 물리적 링크들에 걸쳐 로드-밸런싱될 수 있다. 하나 이상의 LAG들이 2개의 엔드포인트들 사이에 구성될 수 있다. 2개의 엔드포인트들은 NVD와 TOR 스위치 사이에, 호스트 머신과 NVD 사이에 있는 등일 수 있다.

[0097] NVD는 네트워크 가성화 기능들을 구현하거나 또는 수행한다. 이러한 기능들은 NVD에 의해 실행되는 소프트웨어/펌웨어에 의해 수행된다. 네트워크 가상화 기능들의 예들은 비제한적으로 다음을 포함한다: 패킷 캡슐화 및 캡슐화-해제 기능들; VCN 네트워크를 생성하기 위한 기능들; VCN 보안 리스트 (방화벽) 기능성과 같은 네트워크 정책들을 구현하기 위한 기능들; VCN 내의 컴퓨팅 인스턴스들로 그리고 이로부터 패킷들을 라우팅하고 포워딩하는 것을 가능하게 하는 기능들; 및 유사한 것. 특정 실시예들에서, 패킷의 수신 시에, NVD는 패킷을 프로세싱하고 패킷이 포워딩되거나 또는 라우팅되는 방법을 결정하기 위해 패킷 프로세싱 파이프라인을 실행하도록 구성된다. 이러한 패킷 프로세싱 파이프라인의 부분으로서, NVD는, VCN 내의 컴퓨팅 인스턴스들(cis)과 연관된 VNIC들을 실행하는 것, VCN과 연관된 가상 라우터(Virtual Router; VR)를 실행하는 것, 가상 네트워크 내에서의 포워딩 또는 라우팅을 가능하게 하기 위한 패킷들의 캡슐화 및 캡슐화해제, 특정 게이트웨이들(예를 들어, 로컬 피어링 게이트웨이)의 실행, 보안 리스트들의 구현, 네트워크 보안 그룹들, 네트워크 어드레스 변환(NAT) 기능성(예를 들어, 공용 IP를 호스트 단위로 호스트 상의 사설 IP로 변환하는 것), 스로틀링 기능들, 및 다른 기능들과 같은 오버레이 네트워크와 연관된 하나 이상의 가상 기능들을 실행할 수 있다.

[0098] 특정 실시예들에서, NVD 내의 패킷 프로세싱 데이터 경로는 다수의 패킷 파이프라인들을 포함할 수 있으며, 각각의 패킷 파이프라인은 일련의 패킷 변환 스테이지들로 구성된다. 특정 구현예들에서, 패킷의 수신 시에, 패킷

은 파싱(parse)되고 단일 파이프라인으로 분류된다. 그런 다음 패킷은, 패킷이 드롭되거나 또는 NVD의 인터페이스를 통해 전송될 때까지 선형적인 방식으로 한 스테이지씩 프로세싱된다. 이러한 스테이지들은, 새로운 파이프라인들이 기존 스테이지들을 조직(compose)함으로써 구성될 수 있고 새로운 기능성이 새로운 스테이지들을 생성하고 이들을 기존 파이프라인들에 삽입함으로써 추가될 수 있도록 기본 기능 패킷 프로세싱 구축 블록들(예를 들어, 헤더들의 검증, 스로틀의 시행, 새로운 계층-2 헤더들의 삽입, L4 방화벽의 시행, VCN 캡슐화/캡슐화해제, 등)을 제공한다.

[0099] NVD는 VCN의 제어 평면 및 데이터 평면에 대응하는 제어 평면 기능 및 데이터 평면 기능 둘 모두를 수행할 수 있다. VCN 제어 평면의 예들은 또한 도 12, 도 13, 도 14 및 도 15(참조 부호 1216, 1316, 1416, 및 1516 참조)에 도시되며 이하에서 설명된다. VCN 데이터 평면의 예들은 도 12, 도 13, 도 14 및 도 15(참조 부호 1218, 1318, 1418, 및 1518 참조)에 도시되며 이하에서 설명된다. 제어 평면 기능들은, 데이터가 포워딩될 방법을 제어하는 네트워크를 구성(예를 들어, 루트들 및 루트 테이블들을 셋업하는 것, VNIC들을 구성하는 것, 등)하기 위해 사용되는 기능들을 포함한다. 특정 실시예들에서, 오버레이-대-서브스트레이트 매핑들 모두를 중앙에서 컴퓨팅하고 이들을 NVD들에 그리고 DRG, SGW, IGW, 등과 같은 다양한 게이트웨이들과 같은 가상 네트워크 에지 디바이스들에 퍼블리싱하는 VCN 제어 패널이 제공된다. 방화벽 규칙들도 또한 동일한 메커니즘을 사용하여 퍼블리싱된다. 특정 실시예들에서, NVD는 해당 NVD와 관련된 매핑들만을 획득한다. 데이터 평면 기능들은, 제어 평면을 사용하여 셋업된 구성에 기초하여 패킷의 실제 라우팅/포워딩을 위한 기능들을 포함한다. VCN 데이터 평면은, 고객의 네트워크 패킷들이 서브스트레이트 네트워크를 통과하기 이전에 고객의 네트워크 패킷들을 캡슐화함으로써 구현된다. 캡슐화/캡슐화해제 기능성은 NVD들 상에 구현된다. 특정 실시예들에서, NVD는 호스트 머신들 안팎으로의 모든 네트워크 패킷들을 인터셉트하고 네트워크 가상화 기능들을 수행하도록 구성된다.

[0100] 이상에서 표시된 바와 같이, NVD는 VNIC들 및 VCN VR들을 포함하여 다양한 가상화 기능들을 실행한다. NVD는 VNIC에 연결된 하나 이상의 호스트 머신들에 의해 호스팅되는 컴퓨팅 인스턴스들과 연관된 VNIC들을 실행할 수 있다. 예를 들어, 도 2에 도시된 바와 같이, NVD(210)는, NVD(210)에 연결된 호스트 머신(202)에 의해 호스팅되는 컴퓨팅 인스턴스(268)와 연관된 VNIC(276)에 대한 기능성을 실행한다. 다른 예로서, NVD(212)는 호스팅 머신(206)에 의해 호스팅되는 베어 메탈 컴퓨팅 인스턴스(272)와 연관된 VNIC(280)를 실행하고, 호스트 머신(208)에 의해 호스팅되는 컴퓨팅 인스턴스(274)와 연관된 VNIC(284)를 실행한다. 호스트 머신은 상이한 고객들에 속한 상이한 VCN들에 속하는 컴퓨팅 인스턴스들을 호스팅할 수 있으며, 호스트 머신에 연결된 NVD는 컴퓨팅 인스턴스들에 대응하는 VNIC들 실행할 수 있다(즉, VNIC-관련 기능성을 실행할 수 있다).

[0101] NVD는 또한 컴퓨팅 인스턴스들의 VCN들에 대응하는 VCN 가상 라우터들을 실행한다. 예를 들어, 도 2에 도시된 실시예에서, NVD(210)는 컴퓨팅 인스턴스(268)가 속하는 VCN에 대응하는 VCN VR(277)을 실행한다. NVD(212)는 호스트 머신들(206 및 208)에 의해 호스팅되는 컴퓨팅 인스턴스들이 속하는 하나 이상의 VCN들에 대응하는 하나 이상의 VCN VR들(283)을 실행한다. 특정 실시예들에서, 해당 VCN에 대응하는 VCN VR은, 해당 VCN에 속하는 적어도 하나의 컴퓨팅 인스턴스를 호스팅하는 호스트 머신들에 연결된 NVD들 모두에 의해 실행된다. 호스트 머신이 상이한 VCN들에 속하는 컴퓨팅 인스턴스들을 호스팅하는 경우, 해당 호스트 머신에 연결된 NVD는 이러한 상이한 VCN들에 대응하는 VCN VR들을 실행할 수 있다.

[0102] VNIC들 및 VCN VR들에 추가하여, NVD는 소프트웨어(예를 들어, 데몬(daemon)들)을 실행할 수 있으며, NVD에 의해 수행되는 다양한 네트워크 가상화 기능들을 가능하게 하는 하나 이상의 하드웨어 구성요소들을 포함할 수 있다. 단순성의 목적들을 위해, 이러한 다양한 구성요소들은 도 2에 도시된 "패킷 프로세싱 구성요소들"로서 함께 그룹화된다. 예를 들어, NVD(210)는 패킷 프로세싱 구성요소들(286)을 포함하며, NVD(212)는 패킷 프로세싱 구성요소들(288)을 포함한다. 예를 들어, NVD에 대한 패킷 프로세싱 구성요소들은, NVD에 의해 수신되고 이를 사용하여 통신되는 모든 패킷들을 모니터링하고 네트워크 정보를 저장하기 위해 하드웨어 인터페이스들 및 NVD의 포트들과 상호작용하도록 구성된 패킷 프로세서를 포함할 수 있다. 네트워크 정보는, 예를 들어, NVD에 의해 핸들링되는 상이한 네트워크 흐름들을 식별하는 네트워크 흐름 정보 및 흐름별 정보(예를 들어, 흐름별 통계자료)를 포함할 수 있다. 특정 실시예들에서, 네트워크 흐름 정보는 VNIC별로 저장될 수 있다. 패킷 프로세서는 패킷별 조작들을 수행할 뿐만 아니라 스테이트풀 NAT 및 L4 방화벽(FW)을 구현할 수 있다. 다른 예로서, 패킷 프로세싱 구성요소들은, NVD에 의해 저장된 정보를 하나 이상의 상이한 복제 목표 스토어들에 복제하도록 구성된 복제 에이전트를 포함할 수 있다. 또 다른 예로서, 패킷 프로세싱 구성요소는, NVD에 대한 로깅 기능들을 수행하도록 구성된 로깅 에이전트를 포함할 수 있다. 패킷 프로세싱 구성요소들은 또한, NVD의 성능 및 헬스(health)를 모니터링하며 또한 어쩌면 NVD에 연결된 다른 구성요소들의 상태 및 헬스를 모니터링하기 위한 소프트웨어를 포함할 수 있다.

[0103] 도 1은, VCN, VCN 내의 서브넷들, 서브넷들 상에 배포된 컴퓨팅 인스턴스들, 컴퓨팅 인스턴스들과 연관된 VNIC들, VCN에 대한 VR, 및 VCN에 대해 구성된 게이트웨이들의 세트를 포함하는 예시적인 가상 또는 오버레이 네트워크의 구성요소들을 도시한다. 도 1에 도시된 오버레이 구성요소들은 도 2에 도시된 물리적 구성요소들 중 하나 이상에 의해 실행되거나 또는 호스팅될 수 있다. 예를 들어, VCN 내의 컴퓨팅 인스턴스들은 도 2에 도시된 하나 이상의 호스트 머신들에 의해 실행되거나 또는 호스팅될 수 있다. 호스트 머신에 의해 호스팅되는 컴퓨팅 인스턴스에 대해, 해당 컴퓨팅 인스턴스와 연관된 VNIC는 전형적으로 호스트 머신에 연결된 NVD에 의해 실행된다(즉, VNIC 기능성은 해당 호스트 머신에 연결된 NVD에 의해 제공된다). VCN에 대한 VCN VR은, 해당 VCN의 부분인 컴퓨팅 인스턴스를 호스팅하거나 또는 실행하는 호스트 머신들에 연결된 NVD들 모두에 의해 실행된다. VCN과 연관된 게이트웨이들은 하나 이상의 상이한 유형들의 NVD들에 의해 실행될 수 있다. 예를 들어, 특정 게이트웨이들은 smartNIC들에 의해 실행될 수 있으며, 반면 다른 것들은 NVD들의 다른 구현예들 또는 하나 이상의 호스트 머신들에 의해 실행될 수 있다.

[0104] 이상에서 설명된 바와 같이, 고객 VCN 내의 컴퓨팅 인스턴스는 다양하고 상이한 엔드포인트들(여기서, 엔드포인트들은 소스 컴퓨팅 인스턴스와 동일한 서브넷 내에 있을 수 있거나, 또는 소스 컴퓨팅 인스턴스와 동일한 VCN 내에 있지만 상이한 서브넷 내에 있을 수 있음)과 통신할 수 있거나, 또는 소스 컴퓨팅 인스턴스의 VCN 외부에 있는 엔드포인트와 통신할 수 있다. 이러한 통신들은 컴퓨팅 인스턴스들과 연관된 VNIC들, VCN VR들, 및 VCN들과 연관된 게이트웨이들을 사용하여 가능하게 된다.

[0105] VCN 내의 동일한 서브넷 상의 2개의 컴퓨팅 인스턴스들 사이의 통신들에 대해, 통신은 소스 컴퓨팅 인스턴스 및 목적지 컴퓨팅 인스턴스와 연관된 VNIC들을 사용하여 가능하게 된다. 소스 컴퓨팅 인스턴스 및 목적지 컴퓨팅 인스턴스는 동일한 호스트 머신에 의해 또는 상이한 호스트 머신들에 의해 호스팅될 수 있다. 소스 컴퓨팅 인스턴스로부터 발원하는 패킷은 소스 컴퓨팅 인스턴스를 호스팅하는 호스트 머신으로부터 해당 호스트 머신에 연결된 NVD로 포워딩될 수 있다. NVD 상에서, 패킷은 패킷 프로세싱 파이프라인을 사용하여 프로세싱되며, 이는 소스 컴퓨팅 인스턴스와 연관된 VNIC의 실행을 포함할 수 있다. 패킷에 대한 목적지 엔드포인트가 동일한 서브넷 내에 있기 때문에, 소스 컴퓨팅 인스턴스와 연관된 VNIC의 실행은 패킷들이 목적지 컴퓨팅 인스턴스와 연관된 VNIC를 실행하는 NVD로 포워딩되는 결과를 가져오며, 이는 그 후에 패킷을 프로세싱하고 이를 목적지 컴퓨팅 인스턴스로 포워딩한다. 소스 컴퓨팅 인스턴스 및 목적지 컴퓨팅 인스턴스와 연관된 VNIC들은 동일한 NVD 상에서 실행될 수 있거나(예를 들어, 소스 컴퓨팅 인스턴스 및 목적지 컴퓨팅 인스턴스 둘 모두가 동일한 호스트 머신에 의해 호스팅될 때) 또는 상이한 NVD들 상에서 실행될 수 있다(예를 들어, 소스 컴퓨팅 인스턴스 및 목적지 컴퓨팅 인스턴스가 상이한 NVD들에 연결된 상이한 호스트 머신들에 의해 호스팅될 때). VNIC들은 패킷에 대한 다음 홉을 결정하기 위해 NVD에 의해 저장된 라우팅/포워딩 테이블들을 사용할 수 있다.

[0106] 서브넷 내의 컴퓨팅 인스턴스로부터 동일한 VCN 내의 상이한 서브넷 내의 엔드포인트로 통신될 패킷에 대해, 소스 컴퓨팅 인스턴스로부터 발원하는 패킷은 소스 컴퓨팅 인스턴스를 호스팅하는 호스팅 머신으로부터 해당 호스트 머신에 연결된 NVD로 통신된다. NVD 상에서, 패킷은 패킷 프로세싱 파이프라인을 사용하여 프로세싱되며, 이는 하나 이상의 VNIC들, 및 VCN과 연관된 VR의 실행을 포함할 수 있다. 예를 들어, 패킷 프로세싱 파이프라인의 부분으로서, NVD는 소스 컴퓨팅 인스턴스와 연관된 VNIC에 대응하는 기능성을 실행하거나 또는 호출(invoke)한다(VNIC를 실행하는 것으로도 지칭됨). VNIC에 의해 수행되는 기능성은 패킷 상의 VLAN 태그를 확인하는 것을 포함할 수 있다. 패킷의 목적지가 서브넷 외부에 있기 때문에, 다음으로 NVD에 의해 VCN VR 기능성이 호출되고 실행된다. 그런 다음, VCN VR은 패킷을 목적지 컴퓨팅 인스턴스와 연관된 VNIC를 실행하는 NVD로 라우팅한다. 그런 다음, 목적지 컴퓨팅 인스턴스와 연관된 VNIC가 패킷을 프로세싱하고 패킷을 목적지 컴퓨팅 인스턴스로 포워딩한다. 소스 컴퓨팅 인스턴스 및 목적지 컴퓨팅 인스턴스와 연관된 VNIC들은 동일한 NVD 상에서 실행될 수 있거나(예를 들어, 소스 컴퓨팅 인스턴스 및 목적지 컴퓨팅 인스턴스 둘 모두가 동일한 호스트 머신에 의해 호스팅될 때) 또는 상이한 NVD들 상에서 실행될 수 있다(예를 들어, 소스 컴퓨팅 인스턴스 및 목적지 컴퓨팅 인스턴스가 상이한 NVD들에 연결된 상이한 호스트 머신들에 의해 호스팅될 때).

[0107] 패킷에 대한 목적지가 소스 컴퓨팅 인스턴스의 VCN 외부에 있는 경우, 소스 컴퓨팅 인스턴스로부터 발원하는 패킷은 소스 컴퓨팅 인스턴스를 호스팅하는 호스팅 머신으로부터 해당 호스트 머신에 연결된 NVD로 통신된다. NVD는 소스 컴퓨팅 인스턴스와 연관된 VNIC를 실행한다. 패킷의 목적지 엔드 포인트가 VCN 외부에 있기 때문에, 그런 다음 패킷은 해당 VCN에 대한 VCN VR에 의해 프로세싱된다. NVD는 VCN VR 기능성을 호출하며, 이는 패킷이 VCN과 연관된 적절한 게이트웨이를 실행하는 NVD로 포워딩되는 결과를 가져올 수 있다. 예를 들어, 목적지가 고객의 사내 네트워크 내의 엔드포인트인 경우, 패킷은 VCN에 대해 구성된 DRG 게이트웨이를 실행하는 NVD로 VCN VR에 의해 포워딩될 수 있다. VCN VR은 소스 컴퓨팅 인스턴스와 연관된 VNIC를 실행하는 NVD와 동일한 NVD 상에

서 실행될 수 있거나 또는 상이한 NVD에 의해 실행될 수 있다. 게이트웨이는, smartNIC, 호스트 머신, 또는 다른 NVD 구현예일 수 있는 NVD에 의해 실행될 수 있다. 그런 다음, 패킷은 게이트웨이에 의해 프로세싱되고 다음 홉으로 포워딩되며, 이는 그 최종 의도된 목적지 엔드포인트로의 패킷의 통신을 가능하게 한다. 예를 들어, 도 2에 도시된 실시예에서, 컴퓨팅 인스턴스(268)로부터 발원하는 패킷은 (NIC(232)를 사용하여) 링크(220)를 통해 호스트 머신(202)으로부터 NVD(210)로 통신될 수 있다. NVD(210) 상에서, VNIC가 소스 컴퓨팅 인스턴스(268)와 연관된 VNIC이기 때문에 VNIC(276)가 호출된다. VNIC(276)는, 패킷 내의 캡슐화된 정보를 검사하고, 그 의도된 목적지 엔드포인트로의 패킷의 통신을 가능하게 하려는 목적으로 패킷을 포워딩하기 위한 다음 홉을 결정하며, 그런 다음 패킷을 결정된 다음 홉으로 포워딩하도록 구성된다.

[0108] VCN 상에 배포된 컴퓨팅 인스턴스는 다양하고 상이한 엔드포인트들과 통신할 수 있다. 이러한 엔드포인트들은 CSPI(200)에 의해 호스팅되는 엔드포인트들 및 CSPI(200) 외부의 엔드포인트들을 포함할 수 있다. CSPI(200)에 의해 호스팅되는 엔드포인트들은, 고객의 VCN들 또는 고객에게 속하지 않은 VCN들일 수 있는 동일한 VCN 또는 다른 VCN들 내의 인스턴스들을 포함할 수 있다. CSPI(200)에 의해 호스팅되는 엔드포인트들 사이의 통신들은 물리적 네트워크(218)를 통해 수행될 수 있다. 컴퓨팅 인스턴스는 또한, CSPI(200)에 의해 호스팅되지 않거나 또는 CSPI(200) 외부에 있는 엔드포인트들과 통신할 수 있다. 이러한 엔드포인트들의 예들은 고객의 사내 네트워크 또는 데이터 센터 내의 엔드포인트들, 또는 인터넷과 같은 공용 네트워크를 통해 액세스가능한 공용 엔드포인트들을 포함한다. CSPI(200) 외부의 엔드포인트들과의 통신들은 다양한 통신 프로토콜들을 사용하여 공용 네트워크들(예를 들어, 인터넷(도 2에 미도시) 또는 사설 네트워크들(도 2에 미도시)을 통해 수행될 수 있다.

[0109] 도 2에 도시된 CSPI(200)의 아키텍처는 단지 일 예이며, 제한적으로 의도되지 않는다. 대안적인 실시예들에서 변형들, 대안들, 및 수정들이 가능하다. 예를 들어, 일부 구현예들에서, CSPI(200)은 도 2에 도시된 것보다 더 많거나 또는 더 적은 시스템들 또는 구성요소들을 가질 수 있거나, 2개 이상의 시스템들을 결합할 수 있거나, 또는 시스템들의 상이한 구성 또는 배열을 가질 수 있다. 도 2에 도시된 시스템들, 서브시스템들, 및 다른 구성요소들은 개별적인 시스템들, 하드웨어, 또는 이들의 조합들의 하나 이상의 프로세싱 유닛들(예를 들어, 프로세서들, 코어들)에 의해 실행되는 소프트웨어(예를 들어, 코드, 명령어들, 프로그램)로 구현될 수 있다. 소프트웨어는 비-일시적 저장 매체 상에(예를 들어, 메모리 디바이스 상에) 저장될 수 있다.

[0110] 도 4는 특정 실시예들에 따른, 다중테넌시(multitenancy)를 지원하기 위한 I/O 가상화를 제공하기 위한 NVD과 호스트 머신 사이의 연결성을 도시한다. 도 4에 도시된 바와 같이, 호스트 머신(402)은 가상화된 환경을 제공하는 하이퍼바이저(404)를 실행한다. 호스트 머신(402)은 2개의 가상 머신 인스턴스들, 즉, 고객/테넌트 #1에 속하는 VM1(406) 및 고객/테넌트 #2에 속하는 VM2(408)를 실행한다. 호스트 머신(402)은 링크(414)를 통해 NVD(412)에 연결된 물리적 NIC(410)를 포함한다. 컴퓨팅 인스턴스들 각각은 NVD(412)에 의해 실행되는 VNIC에 접속된다. 도 4의 실시예에서, VM1(406)은 VNIC-VM1(420)에 접속되고, VM2(408)는 VNIC-VM2(422)에 접속된다.

[0111] 도 4에 도시된 바와 같이, NIC(410)는 2개의 논리적 NIC들, 즉 논리적 NIC A(416)및 논리적 NIC B(418)를 포함한다. 각각의 가상 머신은 그 자체의 논리적 NIC에 접속되고 이와 함께 작동하도록 구성된다. 예를 들어, VM1(406)은 논리적 NIC A(416)에 접속되며, VM2(408)는 논리적 NIC B(418)에 접속된다. 호스트 머신(402)이 다수의 테넌트들에 의해 공유되는 하나의 물리적 NIC(410)만을 포함하지만, 논리적 NIC들로 인해, 각각의 테넌트의 가상 머신은 이들이 그들 자체의 호스트 모신 및 NIC를 갖는다고 믿는다.

[0112] 특정 실시예들에서, 각각의 논리적 NIC에는 그 자체의 VLAN ID가 할당된다. 따라서, 특정 VLAN ID가 테넌트 #1에 대한 논리적 NIC A(416)에 할당되며 별도의 VLAN ID가 테넌트 #2에 대한 논리적 NIC B(418)에 할당된다. 패킷이 VM1(406)로부터 통신될 때, 테넌트 #1에 할당된 태그가 하이퍼바이저에 의해 패킷에 첨부되며, 그런 다음 패킷은 링크(414)를 통해 호스트 머신(402)으로부터 NVD(412)로 통신된다. 유사한 방식으로, 패킷이 VM2(408)로부터 통신될 때, 테넌트 #2에 할당된 태그가 하이퍼바이저에 의해 패킷에 첨부되며, 그런 다음 패킷은 링크(414)를 통해 호스트 머신(402)으로부터 NVD(412)로 통신된다. 따라서, 호스트 머신(402)으로부터 NVD(412)로 통신된 패킷(424)은, 특정 테넌트 및 연관된 VM을 식별하는 연관된 태그(426)를 갖는다. NVD 상에서, 호스트 머신(402)으로부터 수신된 패킷(424)에 대해, 패킷과 연관된 태그(426)는 패킷이 VNIC-VM1(420)에 의해 프로세싱될지 또는 VNIC-VM2(422)에 의해 프로세싱될지 여부를 결정하기 위해 사용된다. 그런 다음, 패킷은 대응하는 VNIC에 의해 프로세싱된다. 도 4에 도시된 구성은, 각각의 테넌트의 컴퓨팅 인스턴스가 이들이 이들 자체의 호스트 머신 및 NIC를 갖는다고 믿는 것을 가능하게 한다. 도 4에 도시된 셋업은 다중-테넌시를 지원하기 위한 I/O 가상화를 제공한다.

[0113] 도 5는 특정 실시예들에 따른 물리적 네트워크(500)의 단순화된 블록도를 도시한다. 도 5에 도시된 실시예는 클

로스 네트워크로서 구조화된다. 클로스 네트워크는 높은 이등분(bisection) 대역폭 및 최대 자원 사용을 유지하면서 연결 중복성을 제공하도록 설계된 특정 유형의 네트워크 토폴로지이다. 클로스 네트워크는 비-차단, 멀티 스테이지 또는 다중-계층형 스위칭 네트워크의 일 유형이며, 여기서 스테이지들 또는 계층들의 수는 2, 3, 4, 5, 등일 수 있다. 도 5에 도시된 실시예는 계층들 1, 2, 및 3을 포함하는 3-계층형 네트워크이다. TOR 스위치들(504)은 클로스 네트워크 내의 계층-0 스위치들을 나타낸다. 하나 이상의 NVD들이 TOR 스위치들에 연결된다. 계층-0 스위치들은 물리적 네트워크의 에지 디바이스들로도 지칭된다. 계층-0 스위치들은, 리프 스위치들로도 지칭되는 계층-1 스위치들에 연결된다. 도 5에 도시된 실시예에서, "n"개의 계층-0 TOR 스위치들의 세트는 "n"개의 계층-1 스위치들의 세트에 연결되어 함께 포드(pod)를 형성한다. 포드 내의 각각의 계층-0 스위치는 포드 내의 모든 계층-1 스위치들에 상호연결되지만, 포드들 사이의 스위치들의 연결성은 존재하지 않는다. 특정 구현예들에서, 2개의 포드들이 블록으로 지칭된다. 각각의 블록은 "n"개의 계층-2 스위치들(때때로 스파인(spine) 스위치들로 지칭됨)의 세트에 의해 서비스되거나 또는 이에 연결된다. 물리적 네트워크 토폴로지 내에 몇몇 블록들이 있을 수 있다. 계층-2 스위치들은 결과적으로 "n"개의 계층-3 스위치들(때때로 슈퍼-스파인(super-spine) 스위치들로 지칭됨)에 연결된다. 물리적 네트워크(500)를 통한 패킷들의 통신은 전형적으로 하나 이상의 계층-3 통신 프로토콜들을 사용하여 수행된다. 전형적으로, TOR 계층을 제외한 물리적 네트워크 내의 모든 계층들은 n-방향(n-way) 중복이며 따라서 고 가용성을 허용한다. 물리적 네트워크의 스케일링을 가능하게 하기 위해 물리적 네트워크 내에서 서로에 대한 스위치들의 가시성을 제어하기 위해 포드들 및 블록들에 대해 정책들이 지정될 수 있다.

[0114] 계층-0 스위치로부터 다른 계층-0 스위치에(또는 계층-0 스위치에 연결된 NVD로부터 계층-0 스위치에 연결된 다른 NVD에) 도달하기 위한 최대 홉 카운트가 고정된다는 것이 클로스 네트워크의 특징이다. 예를 들어, 3-계층형 클로스 네트워크에서, 패킷이 하나의 NVD로부터 다른 NVD에 도달하기 위해 최대 7개의 홉들이 필요하며, 여기서 소스 및 목표 NVD들은 클로스 네트워크의 리프 계층에 연결된다. 마찬가지로, 4-계층형 클로스 네트워크에서, 패킷이 하나의 NVD로부터 다른 NVD에 도달하기 위해 최대 9개의 홉들이 필요하며, 여기서 소스 및 목표 NVD들은 클로스 네트워크의 리프 계층에 연결된다. 따라서, 클로스 네트워크 아키텍처는 네트워크 전체에 걸쳐 일관된 레이턴시를 유지하며, 이는 데이터 센터들 내의 그리고 데이터 센터들 사이의 통신에 대해 중요하다. 클로스 토폴로지는 수평으로 스케일링하며 비용 효율적이다. 네트워크의 대역폭/스루풋 용량은, 다양한 계층들에 더 많은 스위치들(예를 들어, 더 많은 리프 및 스파인 스위치들)을 추가함으로써 그리고 인접한 계층들의 스위치들 사이의 링크들의 수를 증가시킴으로써 쉽게 증가될 수 있다.

[0115] 특정 실시예들에서, CSPI 내의 각각의 자원에는 클라우드 식별자(Cloud Identifier; CID)라고 하는 고유 식별자가 할당된다. 이러한 식별자는 자원의 정보의 부분으로서 포함되며, 예를 들어, 콘솔을 통해 또는 API들을 통해 자원을 관리하기 위해 사용될 수 있다. CID에 대한 예시적인 신택스(syntax)는 다음과 같다:

[0116] ocid1.<RESOURCE TYPE>.<REALM>.[REGION][.FUTURE USE].<UNIQUE ID>

[0117] 여기서,

[0118] ocid1: CID의 버전을 표시하는 리터럴(literal) 스트링;

[0119] resource type: 자원의 유형(예를 들어, 인스턴스, 볼륨, VCN, 서브넷, 사용자, 그룹, 등);

[0120] realm: 자원이 있는 렐름. 예시적인 값들은 상업적 렐름에 대해 "c1", 정부 클라우드 렐름에 대해 "c2", 또는 연방 정부 클라우드 렐름에 대해 "c3" 등이다. 각각의 렐름은 그 자체의 도메인 명칭을 가질 수 있다;

[0121] region: 자원이 있는 지역. 지역이 자원에 적용가능하지 않은 경우, 이러한 부분은 블랭크일 수 있다;

[0122] future use: 향후 사용을 위해 예약됨.

[0123] unique ID: ID의 고유 부분. 포맷은 자원 또는 서비스의 유형에 의존하여 변화할 수 있다.

[0124] 도 6은 특정 실시예들에 따른 클로스 네트워크 배열을 통합하는 클라우드 인프라스트럭처(600)의 블록도를 도시한다. 클라우드 인프라스트럭처(600)는 복수의 랙들(예를 들어, 랙 1(610), 및 랙 2(620))을 포함한다. 각각의 랙은 복수의 호스트 머신들(본 명세서에서 호스트들로도 지칭됨)을 포함한다. 랙 1(610)은 2개의 호스트 머신들, 즉, 호스트 1-A(612) 및 호스트 1-B(614)를 포함한다. 랙 2(620)는 2개의 호스트 머신들, 즉, 호스트 2-A(622) 및 호스트 2-B(624)를 포함한다. 도 6의 예시(즉, 2개의 호스트 머신들을 포함하는 각각의 랙)가 예시적이고 비-제한적으로 의도된다는 것이 이해될 것이다. 예를 들어, 클라우드 인프라스트럭처는 3개 이상의 랙들을 포함할 수 있으며, 여기서 각각의 랙은 3개 이상의 호스트 머신들을 포함할 수 있다. 또한, 각각의 랙이 동

일한 수의 호스트들을 갖도록 제한되지 않는다는 것을 유의해야 한다. 오히려, 랙은 다른 랙에 포함된 호스트 머신들의 수와 비교하여 더 높거나 또는 더 낮은 수의 호스트 머신들을 가질 수 있다.

[0125] 각각의 호스트 머신은 복수의 그래픽 프로세싱 유닛(graphical processing unit; GPU)들을 포함한다. 예를 들어, 호스트 머신 1-A(612)는 N개의 GPU들, 예를 들어, GPU 1(613)을 포함한다. 또한, 각각의 호스트 머신이 동일한 수의 GPU들, 즉, N개의 GPU들을 갖는 도 6의 예시가 예시적이고 비-제한적이며, 즉, 각각의 호스트 머신이 상이한 수의 GPU들을 포함할 수 있다는 것이 이해될 것이다. 각각의 랙은, 랙 내의 호스트 머신들에서 호스팅되는 GPU들과 통신가능하게 결합된 탑 오브 랙(top of rack; TOR) 스위치를 포함한다. 예를 들어, 랙 1(610)은 호스트 머신들인 호스트 1-A(612) 및 호스트 1-B(614)에 통신가능하게 결합되는 TOR 스위치(즉, TOR 1)(616)를 포함하며, 반면 랙 2(620)는 호스트 머신들인 호스트 2-A(622) 및 호스트 2-K(624)에 통신가능하게 결합되는 TOR 스위치(즉, TOR 2)(626)를 포함한다. 도 6에 도시된 TOR 스위치들(즉, TOR 1(616) 및 TOR M(626)) 각각은, TOR 스위치를 랙에 포함된 각각의 호스트 머신에서 호스팅되는 N개의 GPU들에 통신가능하게 결합하기 위해 사용되는 N개의 포트들을 포함한다는 것이 이해된다. 도 6에 도시된 바와 같이 TOR 스위치들을 GPU들에 결합하는 것은 예시적이고 비-제한적으로 의도된다. 예를 들어, 일부 실시예들에서, TOR 스위치는 복수의 포트들을 가질 수 있으며, 포트들 각각은 각각의 호스트 머신 상의 GPU에 대응하고, 즉, 호스트 머신 상의 GPU는 통신 링크를 통해 TOR의 고유 포트에 연결될 수 있다.

[0126] 각각의 랙으로부터의 TOR 스위치들은 복수의 스파인(spine) 스위치들, 예를 들어, 스파인 스위치 1(630) 및 스파인 스위치 p(640)에 통신가능하게 결합된다. 예를 들어, 도 6에 도시된 바와 같이, TOR 1(616)은 2개의 링크들을 통해 스파인 스위치 1(630)에 그리고 다른 2개의 링크들을 통해 스파인 스위치 P(640)에 각각 연결된다. 특징 TOR 스위치로부터 스파인 스위치로부터 전송되는 정보는 본 명세서에서 업링크들을 통해 수행되는 통신으로 지칭되며, 반면 스파인 스위치로부터 TOR 스위치로 송신되는 정보는 본 명세서에서 다운링크들을 통해 수행되는 통신으로 지칭된다. 일부 실시예들에 따르면, TOR 스위치들 및 스파인 스위치들은 클로스 네트워크 배열(예를 들어, 다중-스테이지 스위칭 네트워크)로 연결되며, 여기서 각각의 TOR 스위치는 클로스 네트워크 내의 '리프' 노드를 형성한다.

[0127] 일부 실시예들에 따르면, 호스트 머신들에 포함된 GPU들은 기계 학습과 관련된 태스크들을 실행한다. 이러한 세팅에서, 단일 태스크는, 다수의 호스트 머신들에 걸쳐 그리고 다수의 랙들에 걸쳐 분산될 수 있는 많은 수의 GPU들(예를 들어, 64개의 GPU들)에 걸쳐 수행/분산될 수 있다. 이러한 모든 GPU들이 동일한 태스크(즉, 워크로드)에 대해 작동하기 때문에, 이들 모두는 시간 동기화 방식으로 서로 통신해야 한다. 또한, 임의의 주어진 시점에, GPU들은 컴퓨팅 모드 또는 통신 모드 중 하나에 있으며, 즉, GPU들은 거의 동일한 시간 순간에 서로 대화한다. 워크로드의 속도는 가장 느린 GPU의 속도에 의해 결정된다.

[0128] 전형적으로, 소스 GPU로부터 목적지 GPU로 패킷들을 라우팅하기 위해, 등가 비용 다중경로(equal cost multipath; ECMP) 라우팅이 사용된다. ECMP 라우팅에서, 발신기로부터 수신기로 트래픽을 라우팅하기 위해 다수의 등가 비용 경로들이 이용가능할 때, 특정 경로를 선택하기 위해 선택 기술이 사용된다. 따라서, 트래픽을 수신하는 네트워크 디바이스(예를 들어, TOR 스위치 또는 스파인 스위치)에서, 선택 알고리즘은 네트워크 디바이스로부터 후속 디바이스로 트래픽을 포워딩하기 위해 사용될 아웃고잉 링크를 선택하기 위해 사용된다. 이러한 아웃고잉 링크 선택은 발신기(sender)로부터 수신기까지의 경로 내의 각각의 네트워크 디바이스에서 발생한다. 해시-기반 선택은 광범위하게 사용되는 ECMP 선택 기술이며, 여기서 해시는, 예를 들어, 패킷의 4-튜플(tuple)(예를 들어, 소스 포트, 목적지 포트, 소스 IP, 목적지 IP)에 기초할 수 있다.

[0129] ECMP 라우팅은 흐름 인지 라우팅 기술이며, 여기서 각각의 흐름(즉, 데이터 패킷들의 스트림)은 흐름의 지속기간 동안 동일한 경로로 해시된다. 따라서, 흐름 내의 패킷들은 특정 아웃고잉 포트/링크를 사용하여 네트워크 디바이스로부터 포워딩된다. 이는 전형적으로, 흐름 내의 패킷들이 순서대로 도달하는 것, 즉, 패킷들의 재-순서화가 필요하지 않다는 것을 보장하기 위해 이루어진다. 그러나, ECMP는 대역폭(또는 스루풋)을 인식하지 못한다. 다시 말해서, TOR 및 스파인 스위치들은 병렬 링크들 상의 흐름들의 통계적 흐름-인식(스루풋 미인식) ECMP 로드 밸런싱을 수행한다.

[0130] 표준 ECMP 라우팅(즉, 흐름 인식 라우팅 단독)에서, 문제는, 2개의 별도의 인커밍 링크들을 통해 네트워크 디바이스에 의해 수신되는 흐름들이 동일한 아웃고잉 링크에 해시되어 흐름 충돌을 야기할 수 있다는 점이다. 예를 들어, 2개의 흐름들이 2개의 별도의 인커밍 100G 링크들을 통해 들어오고, 흐름들 각각이 동일한 100G 아웃고링 링크에 해시되는 상황을 고려해 본다. 이러한 상황은, 인커밍 대역폭이 200G이지만 아웃고잉 대역폭은 100G이기 때문에 혼잡(즉, 흐름 충돌)을 야기하고 패킷들이 드롭되는 것을 야기한다. 도 6에 도시된 바와 같이, 2개의 흐

름들이 있다: 호스트 머신인 호스트 1-A(612)로부터 TOR 스위치(616)로 보내지는 흐름 1(641), 및 호스트 머신(614) 상의 다른 GPU로부터 TOR 스위치(616)로 보내지는 흐름 2(643). 2개의 흐름들이 별도의 링크들 상의 TOR 스위치로 보내 진다는 것을 유의해야 한다. 도 6에 도시된 모든 링크들은 100G의 용량(즉, 대역폭)을 갖는 것으로 가정된다. TOR 스위치(616)가 ECMP 라우팅 알고리즘을 수행하는 경우에, 2개의 흐름들이 TOR의 동일한 링크, 예를 들어, TOR 스위치(616)를 스파인 스위치(630)에 연결하는 링크(650)를 사용하도록 해시되는 것이 가능하다. 이러한 경우에, 2개의 흐름들 사이의 충돌('X' 마크로 표현됨)이 있으며, 이는 패킷들이 드롭되는 것을 야기한다.

[0131] 이러한 충돌 시나리오는 일반적으로 프로토콜과 무관하게 모든 유형들의 트래픽에 대해 문제가 된다. 예를 들어, TCP는, 패킷이 드롭되고 발신자가 해당 드롭된 패킷에 대해 수신확인을 획득하지 못할 때 패킷이 재-송신된다는 점에서 지능형이다. 그러나, 이러한 상황은 원격 직접 메모리 액세스(remote direct memory access; RDMA) 유형 트래픽에 대해 더 악화된다. RDMA 네트워크들은 다양한 이유로 TCP를 사용하지 않는다(예를 들어, TCP는 높은 성능을 갖지 않는다). RDMA 네트워크들은 인피니밴드(Infiniband)를 통한 RDMA 또는 수렴형 이더넷을 통한 RDMA(RDMA over converged Ethernet; RoCE)와 같은 프로토콜들을 사용한다. RoCE에서, 혼잡 제어 알고리즘이 존재하며, 여기서 발신자는 혼잡 또는 드롭된 패킷들의 발생을 식별할 때, 발신자는 패킷들의 송신의 속도를 늦춘다. 드랍된 패킷에 대해, 드랍된 패킷들뿐만 아니라, 드롭된 패킷들 주위의 몇몇 패킷들이 재송신되며, 이는 이용가능한 대역폭을 추가로 잡아먹고 열악한 성능을 야기한다.

[0132] 흐름 충돌 이슈는, 엄격한 시간 동기화 요건들로 인해 GPU들에 대해 중요한 문제이다. 예를 들어, 이상에서 언급된 바와 같이, GPU들은 기계 학습 태스크(즉, 워크로드)를 실행할 수 있으며, 여기서 모든 GPU들이 시간 동기화 방식으로 서로 통신한다. 기계 학습 태스크들 및 다른 유형들의 태스크들에 대해, 논리적 토폴로지(예를 들어, 링 토폴로지, 트리 토폴로지, 등)는 GPU들 사이의 통신을 가능하게 하기 위해 호스트 머신들에 대해 구성된다. GPU들은 다중레벨 또는 다중-차원일 수 있는 논리적 토폴로지를 사용하여 서로 연결된다. 일부 구현예들에서, 워크로드를 실행하기 위해, 애플리케이션은 GPU들을 상호연결하기 위한 가상(또는 논리적) 토폴로지를 구성한다. 전형적으로, 이러한 애플리케이션은 호스트 머신들의 기초 물리적 토폴로지를 인식하지 못하며, 따라서 랜덤한(즉, 임의적인) 방식으로 논리적 토폴로지를 구성하려고 시도한다. 이러한 랜덤하게 구성된 논리적 토폴로지들은 GPU 호스트 머신들이 그들의 로컬 네트워크 이웃에 어떤 다른 GPU 호스트들이 있는지에 관계없이 트래픽을 교환하는 결과를 가져온다. 이와 같이, 트래픽 혼잡에 대한 가능성이 증가되며, 이는 열악한 GPU 스루풋을 야기한다. 예를 들어, 호스트 머신들의 쌍이 특정 기계 학습 태스크를 실행하기 위한 필요한 경우를 고려해 본다. 이러한 경우에, 호스트 머신들의 쌍의 랜덤 선택이 수행되며, 여기서 호스트 머신들 중 하나는 제1 로컬 이웃(예를 들어, 제1 랙)에 존재하고 다른 호스트 머신은 (제1 로컬 이웃과는 상이한) 다른 로컬 이웃, 예를 들어, 제2 랙에 존재하는 경우, 기계 학습 태스크의 실행은 특정한 양의 레이턴시(예를 들어, 제1 호스트 머신과 제2 호스트 머신 사이의 통신에서 초래되는 지연)를 초래할 뿐만 아니라 트래픽 혼잡의 가능성을 증가시킬 수 있다. 이와 대조적으로, 기계 학습 태스크를 실행하기 위해 선택된 호스트 머신들의 쌍이 동일한 로컬 이웃에 존재하는 경우(예를 들어, 동일한 랙에 존재하는 경우), 호스트 머신들 사이의 통신은 최소 레이턴시를 초래하며 뿐만 아니라 트래픽 혼잡을 회피할 가능성을 개선한다는 것이 이해될 것이다.

[0133] 이상에서 설명된 문제들을 극복하기 위한 기술들이 이하에서 설명된다. 구체적으로, 본 명세서에서 설명되는 기술들은 논리적 토폴로지를 구성하는 프로세스에서 GPU들의 지역성 정보를 사용하여 불필요한 트래픽 혼잡을 회피한다. 또한, 본 개시내용의 실시예들은 고객들이 근접한 호스트 머신들에 그들의 워크로드(들)를 "위치"시킴으로써 그들의 애플리케이션-대-서비스 레이턴시를 감소시키는 것을 가능하게 한다. 또한, 고객들은 더 높은 안티-어피니티를 획득하고 그에 따라 자원들의 공유된 운영을 감소시킴으로써 더 높은 복원력을 얻을 수 있는 방식으로 지역성 정보를 사용하고 그들의 워크로드들을 위치시킬 수 있다.

[0134] 이제 도 7을 참조하면, 특정 실시예들에 따른 랙(700)의 예시적인 구성이 예시된다. 도 7에 도시된 바와 같이, 랙(700)은 2개의 호스트 머신들, 즉, 호스트 1-A(710) 및 호스트 1-B(720)를 포함한다. 랙(700)이 2개의 호스트 머신들을 포함하는 것으로 도시되지만, 랙(700)이 더 많은 호스트 머신들을 포함할 수 있다는 것이 이해될 것이다. 각각의 호스트 머신은 복수의 GPU들 및 복수의 CPU들을 더 포함할 수 있다.

[0135] 호스트 머신들(즉, 호스트 1-A(710) 및 호스트 1-B(720))은 탑 오브 랙 스위치, 즉, TOR 1(750)을 통해 네트워크 패브릭에 통신가능하게 결합된다. 이러한 네트워크 패브릭은 본 명세서에서 랙(700)의 프런트-엔드 네트워크로도 지칭되며, 외부 네트워크에 대응한다. 호스트 머신, 즉, 호스트 1-A(710)는 네트워크 인터페이스 카드(network interface card; NIC)(730) 및 TOR 1(750)에 결합된 NVC(735)(즉, 네트워크 가상화 디바이스)를 통해 프런트-엔드 네트워크에 연결된다. 호스트 머신, 즉, 호스트 1-B(720)는 네트워크 인터페이스 카드(network

interface card; NIC)(740) 및 TOR 1(750)에 결합된 네트워크 가상화 디바이스(745)를 통해 프런트-엔드 네트워크에 연결된다. 호스트 머신들인 호스트 1-A(710) 및 호스트 1-B(720)는 다른 측에서 QoS-가능(QoS-enabled) 백-엔드 네트워크에 연결된다. QoS 가능 백엔드 네트워크는 도 6에 도시된 바와 같은 GPU 클러스터 네트워크에 대응할 수 있다. 호스트 머신 1-A(710)는 다른 NIC(765)를 통해 TOR 2 스위치(760)에 연결되며, 이는 호스트 머신을 백-엔드 네트워크에 통신가능하게 결합한다. 유사하게, 호스트 머신 1-B(720)는 NIC(780)를 통해 TOR 2 스위치(760)에 연결되며, 이는 호스트 머신을 백-엔드 네트워크에 통신가능하게 결합한다.

[0136] 일부 실시예들에 따르면, 호스트 머신들은 네트워크의 물리적 토폴로지를 인식하지 못하며, 즉, 특정 호스트 머신은 네트워크 내의 다른 호스트 머신들의 물리적 위치/포지션(location/position)을 인식하지 못한다. 예를 들어, 도 6을 참조하면, 호스트 머신 1-A(612)는, 호스트 머신 1-B(614)가 실제로 동일한 랙(즉, 랙 1(610))에 포함되고 동일한 TOR 스위치, 즉 TOR 1(616) 뒤에 위치된다는 것을 인식하지 못한다. 그러나, 네트워크 제어 평면은 호스트 머신들의 전체 물리적 토폴로지를 인식한다. 일 구현예에서, 네트워크 제어 평면은 트래픽 지역성을 달성하고 불필요한 트래픽 혼잡을 피하기 위해 이러한 지역성 정보를 호스트 머신들로 퍼블리싱(publish)한다. 이렇게 하는 것은 도 8a 및 도 8b를 참조하여 이하에서 예시하는 바와 같이 GPU 워크로드들의 성능에 상당한 영향을 미친다.

[0137] 일부 실시예들에 의해, 네트워크 제어 평면은 호스트 머신으로 메타데이터 정보(예를 들어, 지역성 정보)를 퍼블리싱(그리고 저장)하기 위해 인스턴스 메타데이터 서비스(instance metadata service; IMDS)를 사용한다. 이러한 메타데이터 정보는 호스트 머신(들)과 연관된 개별적인 스마트 NIC들을 통해 호스트 머신(들)으로 퍼블리싱될 수 있다. 예를 들어, 도 7을 참조하면, 지역성 정보는 스마트 NIC(730)를 통해 호스트 1-A(710)로 (IMDS를 통해서) 퍼블리싱되며, 반면 지역성 정보는 스마트 NIC(740)를 통해 호스트 1-B(720)로 (IMDS를 통해서) 퍼블리싱된다. 지역성 정보는 호스트 머신의 랙 정보를 나타내는 메타데이터를 포함할 수 있으며, 예를 들어, 랙 정보는 특정 호스트 머신이 속하는 랙 ID(예를 들어, 랙 1(610))을 나타낼 뿐만 아니라 랙과 연관된 TOR 스위치(예를 들어, TOR 1 스위치(616))를 나타낼 수 있다는 것이 이해될 것이다. 호스트 머신들 각각이 호스트 머신과 연관된 메타데이터 정보를 획득하기 위해 IMDS를 쿼리(query)할 수 있다는 것이 이해될 것이다. 이하에서 설명되는 바와 같이, 퍼블리싱된 지역성 정보는 더 높은 GPU 워크로드 스루풋을 달성하기 위해 최적 논리적 토폴로지들을 구성하기 위해 사용된다.

[0138] 이제 도 8a를 참조하면, 특정 실시예들에 따른 지역성 정보 없이 구성된 논리적 토폴로지가 도시된다. 도 8a에 도시된 논리적 토폴로지는 4개의 호스트 머신들, 즉, 도 6의 호스트 머신 1-A(612), 호스트 머신 1-B(614), 호스트 머신 2-A(622), 및 호스트 머신 2-B(624)에 대응한다. 도 6에 도시된 바와 같이, 호스트 머신들 1-A 및 1-B는 랙 1(610)에 속하며(즉, 동일한 TOR 스위치인 TOR 1(616) 뒤에 위치되며), 반면 호스트 머신들 2-A 및 2-B는 랙 2(620)에 속한다는 것(즉, 동일한 TOR 스위치인 TOR 2(626) 뒤에 위치된다는 것)을 유의해야 한다.

[0139] 도 8a에 도시된 논리적 토폴로지는, 지역성 정보 없이 구성된 링 토폴로지이며, 즉, 링 토폴로지는 랜덤한(임의적인) 방식으로 구성된다. 도 8a에 도시된 바와 같이, 링은, 호스트 1-A가 805로 라벨링된 링크들을 통해 호스트 2-B에 직접적으로 연결되도록(즉, 호스트 1-A의 8개의 GPU들이 호스트 2-B의 대응하는 8개의 GPU들에 연결되도록) 하는 방식으로 구성된다. 또한, 호스트 2-B는 810으로 라벨링된 링크들을 통해 호스트 1-B에 연결되며, 호스트 1-B는 815로 라벨링된 링크들을 통해 호스트 2-A에 연결되고, 호스트 2-A는 820으로 라벨링된 링크들을 통해 호스트 1-A에 연결된다.

[0140] 도 8a에 도시된 바와 같은 방식으로 구성된 논리적 토폴로지는 ECMP 트래픽 분산으로 인해 네트워크 흐름 충돌들이 발생하기 쉽다. 호스트 1-A(612) 및 호스트 1-B(614)가 동일한 TOR 스위치, 즉, TOR 1(616) 뒤에 위치되지만, 호스트 1-B로부터 발원하여 호스트 1-A로 목표되는 트래픽은 다음의 루트를 통과한다: 트래픽은 먼저 (가상 링크들(815)을 통해) 호스트 1-B로부터 호스트 2-A로 라우팅되고, 그런 다음 (가상 링크들(820)을 통해) 호스트 2-A로부터 호스트 1-A로 라우팅된다는 것을 유의해야 한다. 따라서, 출발(origination) 호스트(즉, 호스트 1-B)와 동일한 랙(즉, 랙 1)에 위치된 목적지 호스트(예를 들어, 호스트 1-A)에 대해 의도된 트래픽은 불필요하게 TOR 스위치를 통해 랙 외부의 호스트 머신(즉, 호스트 2-A)으로 라우팅되고, 추가로 랙 내의 목적지 호스트 머신으로 리다이렉션(re-direct)된다. 트래픽이 호스트 머신 2-B로부터 호스트 머신 2-A로 라우팅될 때 유사한 상황이 발생한다. 개별적인 호스트 머신들이 그들의 지역성 정보를 인식하지 못하며 이는 링 토폴로지의 임의적인 구성을 야기한다는 것을 유의해야 한다. 도 8a의 임의적으로 구성된 논리적 토폴로지는 흐름 충돌들의 증가된 가능성을 야기하며, 이는 GPU 워크로드의 스루풋을 감소시킨다(예를 들어, 더 높은 레이턴시, 지터 손실, 등을 야기한다).

[0141] 도 8b는 특정 실시예들에 따른 지역성 정보를 사용하여 구성된 다른 논리적 토폴로지를 도시한다. 도 8b에 도시된 논리적 토폴로지는 4개의 호스트 머신들, 즉, 도 6의 호스트 머신 1-A(612), 호스트 머신 1-B(614), 호스트 머신 2-A(622), 및 호스트 머신 2-B(624)에 대응하며, 지역성 정보에 기초하여 구성된다. 호스트 머신들 1-A 및 1-B는 랙 1(610)에 속하며(즉, 동일한 TOR 스위치인 TOR 1(616) 뒤에 위치되며), 반면 호스트 머신들 2-A 및 2-B는 랙 2(620)에 속한다는 것(즉, 동일한 TOR 스위치인 TOR 2(626) 뒤에 위치된다는 것)을 유의해야 한다.

[0142] 도 8b에 도시된 논리적 토폴로지는 지역성 정보로 구성된 링 토폴로지이며, 즉, 링 토폴로지는, 예를 들어, IMDS로부터 획득된 지역성 정보에 기초하여 구성된다. 일부 실시예들에 의해, 논리적 토폴로지는, 랙에 포함된 호스트 머신들 중 하나일 수 있는 구성 호스트 머신에 의해 구성될 수 있다. 도 8b에 도시된 바와 같이, 링은, 호스트 1-B가 865로 라벨링된 링크들을 통해 호스트 1-A에 직접적으로 연결되도록(즉, 호스트 1-B의 8개의 GPU들이 호스트 1-A의 대응하는 8개의 GPU들에 연결되도록) 하는 방식으로 구성된다. 또한, 호스트 1-A는 850으로 라벨링된 링크들을 통해 호스트 2-B에 연결되며, 호스트 2-B는 855로 라벨링된 링크들을 통해 호스트 2-A에 연결되고, 호스트 2-A는 860으로 라벨링된 링크들을 통해 호스트 1-B에 연결된다.

[0143] 도 8b에 도시된 논리적 토폴로지는 동일한 랙에 있는(즉, 동일한 TOR 스위치 뒤에 위치된) 2개의 호스트 머신들 사이의 트래픽이 불필요하게 랙 외부로 라우팅되는 것을 피한다. 예를 들어, 트래픽이 호스트 1-B로부터 호스트 1-A로 라우팅되는 경우를 고려해 보면, 트래픽은 (가상 링크들(865)을 통해) 호스트 머신 1-B로부터 호스트 머신 1-A로 직접적으로 전송될 수 있다. 이는, 호스트 1-B에서 발원하여 호스트 1-A로 목표된 트래픽이 먼저 호스트 2-A로 보내지고(즉, TOR 스위치를 통해 외부로 라우팅되고) 추가로 목적지 호스트 머신, 즉, 호스트 1-A로 전송되는 도 8a에 대해 수행되는 바와 같은 라우팅과는 대조적이다. 이러한 방식으로 지역성 정보에 기초하여 논리적 토폴로지를 구성하는 것은 흐름 충돌들의 감소된 가능성을 야기하며, 이는 GPU 워크로드들의 스루풋을 증가시킨다.

[0144] 도 9는 특정 실시예들에 따른 요청을 프로비저닝하는 데 수행되는 단계들을 도시하는 예시적인 흐름도를 예시한다. 도 9에 도시된 프로세싱은 개별적인 시스템들, 하드웨어, 또는 이들의 조합의 하나 이상의 프로세싱 유닛들(예를 들어, 프로세서들, 코어들)에 의해 실행되는 소프트웨어(예를 들어, 코드, 명령어들, 프로그램)로 구현될 수 있다. 소프트웨어는 비-일시적 저장 매체 상에(예를 들어, 메모리 디바이스 상에) 저장될 수 있다. 도 9에 제시되고 이하에서 설명되는 방법은 예시적으로 그리고 비-제한적으로 의도된다. 도 9가 특정 시퀀스 또는 순서로 발생하는 다양한 프로세싱 단계들을 도시하지만, 이는 제한적으로 의도되지 않는다. 특정한 대안적인 실시예들에서, 단계들은 어떤 상이한 순서로 수행될 수 있으며, 일부 단계들이 또한 병렬로 수행될 수 있다.

[0145] 프로세스는 단계(905)에서 개시되며, 여기서 복수의 호스트 머신들(즉, 클러스터에 포함된 호스트 머신들)의 각각의 호스트 머신에 대해, 호스트 머신에 대한 지역성 정보가 저장된다. 호스트 머신에 대한 지역성 정보는 호스트 머신을 포함하는 랙을 식별하는 정보를 포함한다. 일부 실시예들에 의해, 인스턴스 메타데이터 서비스는 호스트 머신과 연관된 네트워크 가상화 디바이스(network virtualization device; NVD)를 통해 지역성 정보를 호스트 머신에 저장하기 위해 사용될 수 있다. 지역성 정보는, 호스트 머신을 포함하는 랙의 식별자(즉, 랙 ID), 랙과 연관된 TOR 스위치의 식별자, 등을 나타내는 정보에 대응할 수 있다는 것을 유의해야 한다.

[0146] 단계(910)에서, 제어 평면은 (예를 들어, 고객으로부터) 워크로드의 실행을 요청하는 요청을 수신한다. 워크로드는 호스트 머신들과 연관된 GPU들을 사용하여 실행될 하나 이상의 프로세스들에 대응한다는 것을 유의해야 한다. 그런 다음 프로세스는 단계(915)로 이동하며, 여기서 복수의 호스트 머신들로부터 하나 이상의 호스트 머신들이 워크로드를 실행하기 위해 이용가능한 것으로서 식별된다. 이용가능한 호스트 머신들의 식별은 몇몇 방식들로 수행될 수 있다. 예를 들어, 일 실시예에 의해, 제어 평면은 호스트 머신에 의해 핸들링되는 현재 로드(즉, 워크로드)를 유지할 수 있다. 각각의 호스트 머신의 용량 및 호스트 머신에 의해 핸들링되는 현재 워크로드의 양에 기초하여, 제어 평면은 고객의 워크로드를 실행하기 위해 이용가능한 하나 이상의 호스트 머신들을 선택할 수 있다. 추가적으로, 다른 실시예에 의해, 제어 평면은 특정한 수의 호스트 머신들을 고객별로 사전-할당할 수 있다. 또한, 이용가능한 호스트 머신들은 이러한 특정한 수의 호스트 머신들로부터 결정될 수 있다.

[0147] 그런 다음 프로세스는 단계(920)로 이동하며, 여기서 단계(915)에서 식별된 하나 이상의 호스트 머신들 각각에 대한 지역성 정보가 획득된다. (각각의 호스트 머신의) 계층적 지역성 정보는, 예를 들어, 인스턴스 메타데이터 서비스에 의해 대응하는 호스트 머신에 저장될 수 있다는 것을 유의해야 한다. 단계(925)에서, 프로세스는 하나 이상의 호스트 머신들의 링키지(linkage) 정보를 식별한다. 하나 이상의 호스트 머신들의 링키지 정보가 하나 이상의 호스트 머신들에 의해 형성된 논리적 토폴로지(예를 들어, 도 8b에 도시된 바와 같은 논리적 토폴로지)에 대응한다는 것이 이해될 것이다. 그 후에, 프로세스는 단계(930)로 이동하며, 여기서 하나 이상의 호스트 머

신들의 지역성 정보 및 링키지 정보가 요청에 응답하여 제공된다. 일부 실시예들에 따르면, 고객은 지역성 정보 및 링키지 정보의 획득 시에 워크로드를 실행하기 위한 하나 이상의 호스트 머신들의 서브세트를 선택할 수 있다. 호스트 머신들의 선택은 워크로드와 연관된 하나 이상의 제약들에 기초하여 수행될 수 있다는 것을 유의해야 한다. 제약들과 관련된 세부사항들은 이하에서 도 11 및 도 11을 참조하여 설명된다. 호스트 머신들의 서브세트를 선택하면, 워크로드는 선택된 호스트 머신들에서 실행될 수 있다.

[0148] 도 10은 특정 실시예들에 따른 요청을 프로세싱하는 데 수행되는 단계들을 도시하는 예시적인 흐름도를 예시한다. 도 10에 도시된 프로세싱은 개별적인 시스템들, 하드웨어, 또는 이들의 조합의 하나 이상의 프로세싱 유닛들(예를 들어, 프로세서들, 코어들)에 의해 실행되는 소프트웨어(예를 들어, 코드, 명령어들, 프로그램)로 구현될 수 있다. 소프트웨어는 비-일시적 저장 매체 상에(예를 들어, 메모리 디바이스 상에) 저장될 수 있다. 도 10에 제시되고 이하에서 설명되는 방법은 예시적으로 그리고 비-제한적으로 의도된다. 도 10이 특정 시퀀스 또는 순서로 발생하는 다양한 프로세싱 단계들을 도시하지만, 이는 제한적으로 의도되지 않는다. 특정한 대안적인 실시예들에서, 단계들은 어떤 상이한 순서로 수행될 수 있으며, 일부 단계들이 또한 병렬로 수행될 수 있다.

[0149] 프로세스는 단계(1005)에서 개시되며, 여기서 복수의 호스트 머신들(즉, 클러스터에 포함된 호스트 머신들)의 각각의 호스트 머신에 대해, 호스트 머신에 대한 지역성 정보가 저장된다. 호스트 머신에 대한 지역성 정보는 호스트 머신을 포함하는 랙을 식별하는 정보를 포함한다. 일부 실시예들에 의해, 인스턴스 메타데이터 서비스는 호스트 머신과 연관된 네트워크 가상화 디바이스(network virtualization device; NVD)를 통해 지역성 정보를 호스트 머신에 저장하기 위해 사용될 수 있다. 지역성 정보는, 호스트 머신을 포함하는 랙의 식별자(즉, 랙 ID), 랙과 연관된 TOR 스위치의 식별자, 등을 나타내는 정보에 대응할 수 있다는 것을 유의해야 한다.

[0150] 단계(1010)에서, 제어 평면은 고객으로부터 요청을 수신하며, 여기서 요청은 고객에게 할당된 하나 이상의 호스트 머신들의 정보를 요청한다. 단계(1015)에서, 제어 평면은 고객에게 할당된 하나 이상의 호스트 머신들을 식별한다. 일부 실시예들에 의해, 제어 평면은 상이한 고객들에게 할당된 호스트 머신들의 매핑을 유지할 수 있다. 이러한 방식으로, 제어 평면은 고객에게 할당된 하나 이상의 호스트 머신들을 식별하기 위해 매핑을 사용할 수 있다.

[0151] 그런 다음 프로세스는 단계(1020)로 이동하며, 여기서 단계(1015)에서 식별된 하나 이상의 호스트 머신들에 대한 지역성 정보가 획득된다. (각각의 호스트 머신의) 계층적 지역성 정보는, 예를 들어, 인스턴스 메타데이터 서비스에 의해 대응하는 호스트 머신에 저장될 수 있다는 것을 유의해야 한다. 단계(1025)에서, 프로세스는 하나 이상의 호스트 머신들의 링키지 정보를 식별한다. 하나 이상의 호스트 머신들의 링키지 정보가 하나 이상의 호스트 머신들에 의해 형성된 논리적 토폴로지(예를 들어, 도 8b에 도시된 바와 같은 논리적 토폴로지)에 대응한다는 것이 이해될 것이다. 그 후에, 프로세스는 단계(1030)로 이동하며, 여기서 하나 이상의 호스트 머신들의 지역성 정보 및 링키지 정보가 요청에 응답하여 (예를 들어, 고객에게) 제공된다. 이상에서 언급된 바와 같이, 고객은, 워크로드와 연관된 하나 이상의 제약들에 따라 워크로드를 실행하기 위한 호스트 머신들을 선택하기 위해 단계(1030)에서 제공된 정보를 사용할 수 있다.

[0152] 도 11은 특정 실시예들에 따른 고객의 워크로드 요청을 프로비저닝하는 데 수행되는 단계들을 도시하는 예시적인 흐름도를 예시한다. 도 11에 도시된 프로세싱은 개별적인 시스템들, 하드웨어, 또는 이들의 조합의 하나 이상의 프로세싱 유닛들(예를 들어, 프로세서들, 코어들)에 의해 실행되는 소프트웨어(예를 들어, 코드, 명령어들, 프로그램)로 구현될 수 있다. 소프트웨어는 비-일시적 저장 매체 상에(예를 들어, 메모리 디바이스 상에) 저장될 수 있다. 도 11에 제시되고 이하에서 설명되는 방법은 예시적으로 그리고 비-제한적으로 의도된다. 도 11이 특정 시퀀스 또는 순서로 발생하는 다양한 프로세싱 단계들을 도시하지만, 이는 제한적으로 의도되지 않는다. 특정한 대안적인 실시예들에서, 단계들은 어떤 상이한 순서로 수행될 수 있으며, 일부 단계들이 또한 병렬로 수행될 수 있다.

[0153] 프로세스는 단계(1105)에서 개시되며, 여기서 복수의 호스트 머신들(즉, 클러스터에 포함된 호스트 머신들)의 각각의 호스트 머신에 대해, 호스트 머신에 대한 지역성 정보가 저장된다. 호스트 머신에 대한 지역성 정보는 호스트 머신을 포함하는 랙을 식별하는 정보를 포함한다. 일부 실시예들에 의해, 인스턴스 메타데이터 서비스는 호스트 머신과 연관된 네트워크 가상화 디바이스(network virtualization device; NVD)를 통해 지역성 정보를 호스트 머신에 저장하기 위해 사용될 수 있다. 지역성 정보는, 호스트 머신을 포함하는 랙의 식별자(즉, 랙 ID), 랙과 연관된 TOR 스위치의 식별자, 등을 나타내는 정보에 대응할 수 있다는 것을 유의해야 한다.

[0154] 단계(1110)에서, 고객으로부터 요청이 수신된다. 요청은 워크로드를 실행하기 위해 고객에 의해 희망되는 다수의 호스트 머신들을 요청한다. 요청이 워크로드와 연관된 하나 이상의 제약들에 대응하는 메타데이터를 포함할

수 있다는 것이 이해될 것이다. 일부 실시예들에 의해, 하나 이상의 제약들은 레이턴시 임계치와 연관된 제1 제약을 포함할 수 있으며, 즉, 고객은 레이턴시를 미리 결정된 임계 값보다 더 낮게 함으로써 워크로드가 실행되는 것을 희망할 수 있다. 제2 제약은 안티-어피니티 제약에 대응할 수 있다. 이러한 제약은 고객이 호스트 머신들의 특정한 정도의 가용성을 갖기를 희망하는 것에 대응하며, 즉, 워크로드를 실행하기 위해 선택된 호스트 머신들 중 적어도 일부는 상이한 랙들에 위치되어야 한다. 이러한 제약은 전형적으로 랙 고장 이슈들을 해결하기 위해 고객에 의해 통합된다.

[0155] 단계(1115)에서, 하나 이상의 제약들은 단계(1110)에서 수신된 요청으로부터 추출된다. 그 후에, 프로세스는 단계(1120)으로 이동하며, 여기서 제어 평면은 (복수의 호스트 머신들로부터) 고객의 워크로드를 실행하기 위한 하나 이상의 호스트 머신들을 할당한다. 워크로드를 실행하기 위해 할당된 호스트 머신들은 단계(1115)에서 식별된 하나 이상의 제약들에 따라 결정될 수 있다는 것이 이해될 것이다. 일부 실시예들에 따르면, 단계(1120)의 프로세스는 네트워크 클러스터의 토폴로지, 예를 들어, 랙에 포함된 호스트 머신들의 수, 시스템 내에서 이용가능한 랙들의 수, 등에 따라 워크로드를 실행하기 위한 호스트 머신들을 할당한다. 예를 들어, 각각의 랙이 총 N개의 호스트 머신들을 포함하고, 사용자에 의해 요청된 호스트 머신들의 수가 K개의 호스트 머신들인 경우를 고려해 본다. K가 N보다 더 작은 경우에(즉, K<N), 일 구현예에서 K개의 호스트 머신들 모두는 단일 랙으로부터 할당될 수 있으며, 즉, 모든 K개의 호스트 머신들이 단일 TOR 스위치 뒤에 위치될 수 있다. 이러한 할당이 최소 레이턴시를 가져올 것임을 유의해야 한다. 그러나, 이러한 할당은, 사용자가 또한 특정 레벨의 안티-어피니티, 즉, 가용성을 달성하기를 희망할 수 있기 때문에 사용자의 승인이 계류 중인 동안(pending) 제공될 수 있으며, 이러한 경우에 사용자에게 할당된 호스트 머신들 중 일부는 네트워크 클러스터로부터의 다른 랙(들)으로부터 선택될 수 있다. 다른 예는 K가 N보다 더 큰 경우(K>N)에 대응할 수 있으며, 즉, 사용자에 의해 요청된 호스트 머신들의 수가 랙에 포함된 호스트 머신들의 수보다 더 큰 경우이다. 이러한 경우에, 일 구현예에 의해, 할당은, 최소 수의 TOR 스위치들이 사용되도록, 즉, 인트라-랙 라우팅이 최소화되도록 하는 방식으로 수행될 수 있다. 예를 들어, 도 6 및 도 8b를 참조하고, 제약이 최소 레이턴시를 달성하는 것인 경우를 고려하면, 이러한 경우에, 기계 학습 태스크를 실행하기 위해 2개의 호스트 머신들이 필요한 경우, 일 구현예에서, 호스트 1-A 및 호스트 1-B(이들 둘 모두는 도 6에 도시된 바와 같이 랙 1(610)에 존재함)가 기계 학습 태스크를 실행하도록 할당될 수 있다. 이러한 태스크의 실행 시에, 트래픽은 랙 1 내에서 로컬적으로 포함되어 낮은 레이턴시를 달성한다는 것을 유의해야 한다. 이와 대조적으로, 다른 예에서, 제약이 높은 가용성을 달성하는 것인 경우, 이러한 경우에, 호스트 1-A(랙 1 내에 있음) 및 호스트 2-B(랙 2 내에 있음)가 기계 학습 태스크를 실행하기 위해 할당될 수 있다.

[0156] 그런 다음 프로세스는 단계(1125)로 이동하며, 여기서 단계(1120)에서 할당된 하나 이상의 호스트 머신들 각각에 대한 지역성 정보가 획득된다. (각각의 호스트 머신의) 지역성 정보는 인스턴스 메타데이터 서비스에 의해 대응하는 호스트 머신에 저장될 수 있다는 것을 유의해야 한다. 프로세스는 추가로 하나 이상의 호스트 머신들의 링키지 정보를 식별한다. 하나 이상의 호스트 머신들의 링키지 정보가 하나 이상의 호스트 머신들에 의해 형성된 논리적 토폴로지(예를 들어, 도 8b에 도시된 바와 같은 논리적 토폴로지)에 대응한다는 것이 이해될 것이다. 일 구현예에서, 애플리케이션은 호스트 머신들의 지역성 정보에 기초하여 호스트 머신들의 논리적 토폴로지를 구성할 수 있으며, 여기서 각각의 호스트 머신은 인스턴스 메타데이터 서비스로부터 그 지역성 정보를 획득한다는 것을 유의해야 한다. 고객에게 할당된 하나 이상의 호스트 머신들의 획득된 지역성 정보 및 링키지 정보는 단계(1125)에서 고객에게 제공된다. 프로세스는 단계(1130)로 이동하며, 여기서 고객의 워크로드는 고객에 대해 할당된 하나 이상의 호스트 머신들 상에서 실행된다.

[0157] 따라서, 본 개시내용의 기술들은 고객들이 그들이 워크로드들을 근처에, 예를 들어, 랙 내의 이웃 호스트 머신들에 "위치"시킴으로써 그들의 애플리케이션 대 서비스 레이턴시를 감소시키는 것을 가능하게 한다. 또한, 고객들은 더 높은 안티-어피니티를 달성하고 그에 따라 더 높은 복원력을 획득하는 방식으로 지역성 정보를 사용하고 그들의 워크로드들을 위치시킬 수 있다. 일 실시예에 따르면, GPU 워크로드들을 실행하기 위해 호스트 머신들을 할당하는 프로세스에서 지역성 정보(예를 들어, 랙 정보)를 사용하는 특징은, 랜덤한 방식으로, 즉, 호스트 머신들의 지역성 정보를 고려하지 않거나 또는 이를 고객들에 게 제공하지 않고 호스트 머신들을 사용자 요청들에 할당하는 나이브한 기술들의 성능과 비교하여 GPU 워크로드들의 성능(예를 들어, 스루풋)을 6배만큼 개선한다.

[0158] 도 9 내지 도 11에 도시된 흐름도들에 대한 수정들이 본 개시내용의 범위 내에 잘 속한다는 것이 이해될 것이다. 예를 들어, 일 구현예에서, 워크로드를 실행하도록 구성된 하나 이상의 호스트 머신들에 대한 지역성 정보는 고객에게 제공될 수 있다. 고객은, 그들의 개별적인 사용 케이스들에 대한 링키지 토폴로지를 생성하기

위해, 예를 들어, 하나 이상의 호스트 머신들과 연관된 어피티니, 안티-어피티니, 또는 다른 제약들을 획득하기 위해 하나 이상의 호스트 머신들의 지역성 정보를 사용할 수 있다.

[0159] 예시적인 클라우드 인프라스트럭처 실시예

[0160] 이상에서 언급된 바와 같이, 서비스형 인프라스트럭처(infrastructure as a service; IaaS)는 클라우드 컴퓨팅의 하나의 특정 유형이다. IaaS는 공용 네트워크(예를 들어, 인터넷)를 통해 가상화된 컴퓨팅 자원들을 제공하도록 구성될 수 있다. IaaS 모델에서, 클라우드 컴퓨팅 제공자는 인프라스트럭처 구성요소들(예를 들어, 서버들, 저장 디바이스들, 네트워크 노드들(예를 들어, 하드웨어), 배포 소프트웨어, 플랫폼 가상화(예를 들어, 하이퍼바이저 계층), 또는 유사한 것)을 호스팅할 수 있다. 일부 경우들에서, IaaS 제공자는 또한 이러한 인프라스트럭처 구성요소들에 수반되는 다양한 서비스들(예를 들어, 청구, 모니터링, 로깅, 보안, 로드 밸런싱 및 클러스팅, 등)을 공급할 수 있다. 따라서, 이러한 서비스들이 정책-구동될 수 있기 때문에, IaaS 사용자들은 애플리케이션 가용성 및 성능을 유지하기 위한 로드 밸런싱을 구동하기 위한 정책들을 구현할 수 있다.

[0161] 일부 인스턴스들에서, IaaS 고객은 인터넷과 같은 광역 네트워크(wide area network; WAN)를 통해 자원들 및 서비스들에 액세스할 수 있으며, 애플리케이션 스택의 나머지 요소들을 설치하기 위해 클라우드 제공자의 서비스들을 사용할 수 있다. 예를 들어, 사용자는 가상 머신(VM)들을 생성하고, 각각의 VM에 운영 체계(operating system; OS)들을 설치하며, 데이터베이스와 같은 미들웨어를 배치하고, 작업부하들 및 백업들에 대한 저장 버킷들을 생성하며, 심지어 해당 VM 내에 기업 소프트웨어를 설치하기 위해 IaaS 플랫폼에 로그인할 수 있다. 그런 다음, 고객들은, 네트워크 트래픽의 밸런싱, 애플리케이션 이슈들의 트러블슈팅, 성능의 모니터링, 재해 복구의 관리, 등을 포함하는 다양한 기능들을 수행하기 위해 제공자의 서비스들을 사용할 수 있다.

[0162] 대부분의 경우들에서, 클라우드 컴퓨팅 모델은 클라우드 제공자의 참여를 필요로 할 것이다. 클라우드 제공자는, 반드시 그래야 하지는 않지만, IaaS를 제공(예를 들어, 제공, 임대, 판매)하는 것을 전문으로 하는 제3자 서비스일 수 있다. 엔티티(entity)는 또한 사설 클라우드를 배포하여 인프라스트럭처 서비스들의 자체 제공자가 될 것을 선택할 수도 있다.

[0163] 일부 예들에서, IaaS 배포는 새로운 애플리케이션 또는 애플리케이션의 새로운 버전을 준비된 애플리케이션 서버 또는 유사한 것에 배치하는 프로세스이다. 이는 또한, 서버를 준비하는(예를 들어, 라이브러리들, 데몬들, 등을 설치하는) 프로세스를 포함할 수 있다. 이는 보통 하이퍼바이저 계층(예를 들어, 서버들, 스토리지, 네트워크 하드웨어, 및 가상화) 아래에서 클라우드 제공자에 의해 관리된다. 따라서, 고객은 (예를 들어, 셀프-서비스 가상 머신들(예를 들어, 요청 시 가동될 수 있음) 또는 유사한 것 상에서) 핸들링(OS), 미들웨어, 및/또는 애플리케이션 배포를 담당할 수 있다.

[0164] 일부 예들에서, IaaS 프로비저닝은 사용하기 위한 컴퓨터들 또는 가상 호스트들을 획득하고, 심지어 이들에 필요한 라이브러리들 또는 서비스들을 설치하는 것을 나타낼 수 있다. 대부분의 경우들에서, 배포는 프로비저닝을 포함하지 않으며, 프로비저닝이 먼저 수행되어야 할 수 있다.

[0165] 일부 경우들에서, IaaS 프로비저닝에 대해 2개의 상이한 과제들이 있다. 첫째, 어떤 것도 실행되기 이전에 인프라스트럭처의 초기 세트를 프로비저닝해야 하는 초기 과제가 있다. 둘째, 일단 모든 것이 프로비저닝되면 기존 인프라스트럭처를 진화시키는 과제(예를 들어, 새로운 서비스들을 추가하는 것, 서비스들을 변경하는 것, 서비스들을 제거하는 것, 등)가 있다 일부 경우들에서, 이러한 2개의 과제들은, 인프라스트럭처의 구성이 선언적으로 정의되는 것을 가능하게 함으로써 해결될 수 있다. 다시 말해서, 인프라스트럭처(예를 들어, 필요한 구성요소들 및 이들이 상호작용하는 방법)는 하나 이상의 구성 파일들에 의해 정의될 수 있다. 따라서, 인프라스트럭처의 전체 토폴로지(예를 들어, 어떤 리소스들이 어떤 것에 의존하는지, 그리고 이들이 각각 함께 작용하는 방법)는 선언적으로 설명될 수 있다. 일부 경우들에서, 일단 토폴로지가 정의되면, 구성 파일들에서 설명된 상이한 구성요소들을 생성하거나 및/또는 관리하는 워크플로우(workflow)가 생성될 수 있다.

[0166] 일부 예들에서, 인프라스트럭처는 다수의 상호연결된 요소들을 가질 수 있다. 예를 들어, 코어 네트워크로도 알려진 하나 이상의 가상 사설 클라우드(virtual private cloud; VPC)(예를 들어, 구성가능한 및/또는 공유된 컴퓨팅 자원들의 잠재적으로 온-디맨드 풀)가 존재할 수 있다. 일부 예들에서, 네트워크의 보안이 셋업될 방법 및 하나 이상의 가상 머신(VM)들을 정의하기 위해 프로비저닝된 하나 이상의 보안 그룹 규칙들도 존재할 수 있다. 로드 밸런서, 데이터베이스, 또는 유사한 것과 같은 다른 인프라스트럭처 요소들이 또한 프로비저닝될 수 있다. 점점 더 많은 인프라스트럭처 요소들이 희망되거나 및/또는 추가됨에 따라, 인프라스트럭처는 점진적으로 발전할 수 있다.

[0167] 일부 경우들에서, 연속적인 배포 기술들은 다양한 가상 컴퓨팅 환경들에 걸친 인프라스트럭처 코드의 배포를 가능하게 하기 위해 이용될 수 있다. 추가적으로, 설명되는 기술들은 이러한 환경들 내에서 인프라스트럭처 관리를 가능하게 할 수 있다. 일부 예들에서, 서비스 팀들은, (예를 들어, 다양하고 상이한 지리적 위치들에 걸쳐, 때때로 전 세계에 걸쳐) 하나 이상의 하지만 흔히 다수의 상이한 생산 환경들로 배포되도록 희망되는 코드를 작성할 수 있다. 그러나, 일부 예들에서, 코드가 배포될 인프라스트럭처가 먼저 셋업되어야 한다. 일부 경우들에서, 프로비저닝은 수동으로 이루어질 수 있으며, 프로비저닝 툴은 자원들을 프로비저닝하기 위해 사용될 수 있거나 및/또는 배포 툴은 일단 인프라스트럭처가 프로비저닝되면 코드를 배포하기 위해 사용될 수 있다.

[0168] 도 12는 적어도 일 실시예에 따른 IaaS 아키텍처의 예시적인 패턴을 예시하는 블록도(1200)이다. 서비스 운영자들(1202)은 가상 클라우드 네트워크(virtual cloud network; VCN)(1206)를 포함할 수 있는 보안 호스트 테넌시(tenancy)(1204) 및 보안 호스트 서브넷(1208)에 통신가능하게 결합될 수 있다. 일부 예들에서, 서비스 운영자들(1202)은, Microsoft Windows Mobile® 및/또는 iOS, Windows Phone, Android, BlackBerry 8, Palm OS, 및 유사한 것과 같은 다양한 모바일 운영 체제들과 같은 소프트웨어를 실행하며, 인터넷, e-메일, 단문 메시지 서비스(short message service; SMS), Blackberry®, 또는 다른 통신 프로토콜 가능형(enabled)일 수 있는, 휴대용 핸드헬드 디바이스들(예를 들어, iPhone®, 셀룰러 전화기, iPad®, 컴퓨팅 태블릿, 개인용 정보 단말기(personal digital assistant; PDA)) 또는 웨어러블(wearable) 디바이스들(예를 들어, Google Glass® 머리 착용형 디스플레이(head mounted display))일 수 있는 하나 이상의 클라이언트 컴퓨팅 디바이스들을 사용하고 있을 수 있다. 대안적으로, 클라이언트 컴퓨팅 디바이스들은, 예로서, 다양한 버전들의 Microsoft Windows®, Apple Macintosh®, 및/또는 리눅스 운영 체제들을 실행하는 개인용 컴퓨터들 및/또는 랩탑 컴퓨터들을 포함하는 범용 개인용 컴퓨터들일 수 있다. 클라이언트 컴퓨팅 디바이스들은, 비제한적으로, 예를 들어, 구글 크롬 OS와 같은 다양한 GNU/리눅스 운영 체제들을 포함하는, 다양한 상용-이용가능 UNIX® 또는 UNIX-유사 운영 체제들 중 임의의 것을 실행하는 워크스테이션 컴퓨터들일 수 있다. 대안적으로 또는 추가적으로, 클라이언트 컴퓨팅 디바이스들은, VCN(1206)에 액세스할 수 있는 네트워크 및/또는 인터넷을 통해 통신할 수 있는 임의의 다른 전자 디바이스, 예컨대 씬-클라이언트(thin-client) 컴퓨터, 인터넷-가능형 게이밍 시스템(예를 들어, Kinect® 제스처 입력 디바이스를 갖는 또는 이를 갖지 않는 마이크로소프트 Xbox 게이밍 콘솔), 및/또는 개인용 메시징 디바이스일 수 있다.

[0169] VCN(1206)은, SSH VCN(1212)에 포함된 LPG(1210)를 통해 보안 쉘(secure shell; SSH) VCN(1212)에 통신가능하게 결합될 수 있는 로컬 피어링 게이트웨이(local peering gateway; LPG)(1210)를 포함할 수 있다. SSH VCN(1212)은 SSH 서브넷(1214)을 포함할 수 있으며, SSH VCN(1212)은 제어 평면 VCN(1216)에 포함된 LPG(1210)를 통해 제어 평면 VCN(1216)에 통신가능하게 결합될 수 있다. 또한, SSH VCN(1212)은 LPG(1210)를 통해 데이터 평면 VCN(1218)에 통신가능하게 결합될 수 있다. 제어 평면 VCN(1216) 및 데이터 평면 VCN(1218)은, IaaS 제공자에 의해 소유되거나 및/또는 운영될 수 있는 서비스 테넌시(1219)에 포함될 수 있다.

[0170] 제어 평면 VCN(1216)은 주변 네트워크(예를 들어, 기업 인트라넷과 외부 네트워크들 사이의 기업 네트워크의 부분들)로서 역할하는 제어 평면 비무장 구역(demilitarized zone; DMZ) 계층(1220)을 포함할 수 있다. DMZ-기반 서버들은 제한된 책임들을 가질 수 있으며 보안 침해들을 억제하는 것을 도울 수 있다. 추가적으로, DMZ 계층(1220)은 하나 이상의 부하 밸런서(load balancer; LB) 서브넷(들)(1222), 앱 서브넷(들)(1226)을 포함할 수 있는 제어 평면 앱 계층(1224), 데이터베이스(database; DB) 서브넷(들)(1230)(예를 들어, 프런트엔드 DB 서브넷(들) 및/또는 백엔드 DB 서브넷(들))을 포함할 수 있는 제어 평면 데이터 계층(1228)을 포함할 수 있다. 제어 평면 DMZ 계층(1220)에 포함된 LB 서브넷(들)(1222)은 제어 평면 앱 계층(1224)에 포함된 앱 서브넷(들)(1226) 및 제어 평면 VCN(1216)에 포함될 수 있는 인터넷 게이트웨이(1234)에 통신가능하게 결합될 수 있으며, 앱 서브넷(들)(1226)은 제어 평면 데이터 계층(1228)에 포함된 DB 서브넷(들)(1230), 서비스 게이트 웨이(1236) 및 네트워크 어드레스 변환(network address translation; NAT) 게이트웨이(1238)에 통신가능하게 결합될 수 있다. 제어 평면 VCN(1216)은 서비스 게이트웨이(1236) 및 NAT 게이트웨이(1238)를 포함할 수 있다.

[0171] 제어 평면 VCN(1216)은 앱 서브넷(들)(1226)을 포함할 수 있는 데이터 평면 미러 앱 계층(1240)을 포함할 수 있다. 데이터 평면 미러 앱 계층(1240)에 포함된 앱 서브넷(들)(1226)은 컴퓨팅 인스턴스(1244)를 실행할 수 있는 가상 네트워크 인터페이스 제어기(virtual network interface controller; VNIC)(1242)를 포함할 수 있다. 컴퓨팅 인스턴스(1244)는, 데이터 평면 미러 앱 계층(1240)의 앱 서브넷(들)(1226)을 데이터 평면 앱 계층(1246)에 포함될 수 있는 앱 서브넷(들)(1226)에 통신가능하게 결합할 수 있다.

[0172] 데이터 평면 VCN(1218)은 데이터 평면 앱 계층(1246), 데이터 평면 DMZ 계층(1248), 및 데이터 평면 데이터 계층(1250)을 포함할 수 있다. 데이터 평면 DMZ 계층(1248)은, 데이터 평면 앱 계층(1246)의 앱 서브넷(들)(1226)

및 데이터 평면 VCN(1218)의 인터넷 게이트웨이(1234)에 통신가능하게 결합될 수 있는 LB 서브넷(들)(1222)을 포함할 수 있다. 앱 서브넷(들)(1226)은 데이터 평면 VCN(1218)의 서비스 게이트웨이(1236) 및 데이터 평면 VCN(1218)의 NAT 게이트 웨이(1238)에 통신가능하게 결합될 수 있다. 데이터 평면 데이터 계층(1250)은 또한, 데이터 평면 앱 계층(1246)의 앱 서브넷(들)(1226)에 통신가능하게 결합될 수 있는 DB 서브넷(들)(1230)을 포함할 수 있다.

[0173] 제어 평면 VCN(1216) 및 데이터 평면 VCN(1218)의 인터넷 게이트웨이(1234)는, 공용 인터넷(1254)에 통신가능하게 결합될 수 있는 메타데이터 관리 서비스(1252)에 통신가능하게 결합될 수 있다. 공용 인터넷(1254)은 제어 평면 VCN(1216) 및 데이터 평면 VCN(1218)의 NAT 게이트웨이(1238)에 통신가능하게 결합될 수 있다. 제어 평면 VCN(1216) 및 데이터 평면 VCN(1218)의 서비스 게이트웨이(1236)는 클라우드 서비스들(1256)에 통신가능하게 결합될 수 있다.

[0174] 일부 예들에서, 제어 평면 VCN(1216) 또는 데이터 평면 VCN(1218)의 서비스 게이트웨이(1236)는 공용 인터넷(1254)을 거치지 않는 클라우드 서비스들(1256)에 대한 애플리케이션 프로그래밍 인터페이스(application programming interface; API) 호출(call)들을 만들 수 있다. 서비스 게이트웨이(1236)로부터 클라우드 서비스들(1256)로의 API 호출들은 1-방향일 수 있다: 서비스 게이트웨이(1236)는 클라우드 서비스들(1256)로의 호출들을 만들 수 있으며, 클라우드 서비스들(1256)은 서비스 게이트웨이(1236)로 요청된 데이터를 전송할 수 있다. 그러나, 클라우드 서비스들(1256)은 서비스 게이트웨이(1236)에 대한 API 호출들을 개시하지 않을 수 있다.

[0175] 일부 예들에서, 보안 호스트 테넌시(1204)는, 그렇지 않았다면 격리되었을 서비스 테넌시(1219)에 직접적으로 연결될 수 있다. 보안 호스트 서브넷(1208)은, 달리 격리된 시스템을 통한 2-방향 통신을 가능하게 할 수 있는 LPG(1210)를 통해 SSH 서브넷(1214)과 통신할 수 있다. 보안 호스트 서브넷(1208)을 SSH 서브넷(1214)에 연결하는 것은 서비스 테넌시(1219) 내의 다른 엔티티들에게 보안 호스트 서브넷(1208) 액세스를 제공할 수 있다.

[0176] 제어 평면 VCN(1216)은 서비스 테넌시(1219)의 사용자들이 희망되는 자원을 셋업하거나 또는 달리 프로비저닝하는 것을 가능하게 할 수 있다. 제어 평면 VCN(1216)에 프로비저닝된 희망되는 자원들은 데이터 평면 VCN(1218)에서 배포되거나 또는 달리 사용될 수 있다. 일부 예들에서, 제어 평면 VCN(1216)는 데이터 평면 VCN(1218)으로부터 격리될 수 있으며, 제어 평면 VCN(1216)의 데이터 평면 미러 앱 계층(1240)은, 데이터 평면 미러 앱 계층(1240) 및 데이터 평면 앱 계층(1246)에 포함될 수 있는 VNIC들(1242)을 통해 데이터 평면 VCN(1218)의 데이터 평면 앱 계층(1246)과 통신할 수 있다.

[0177] 일부 예들에서, 시스템의 사용자들 또는 고객들은, 메타데이터 관리 서비스(1252)로 요청들을 통신할 수 있는 공용 인터넷(1254)을 통해 요청들, 예를 들어, 생성, 판독, 업데이트 또는 삭제(create, read, update, or delete; CRUD) 동작들을 만들 수 있다. 메타데이터 관리 서비스(1252)는 요청을 인터넷 게이트웨이(1234)를 통해 제어 평면 VCN(1216)으로 통신할 수 있다. 요청은 제어 평면 DMZ 계층(1220)에 포함된 LB 서브넷(들)(1222)에 의해 수신될 수 있다. LB 서브넷(들)(1222)은 요청이 유효하다는 것을 결정할 수 있으며, 이러한 결정에 응답하여, LB 서브넷(들)(1222)은 요청을 제어 평면 앱 계층(1224)에 포함된 앱 서브넷(들)(1226)로 송신할 수 있다. 요청이 검증되고 공용 인터넷(1254)에 대한 호출을 필요로 하는 경우, 공용 인터넷(1254)에 대한 호출은 공용 인터넷(1254)에 대한 호출을 만들 수 있는 NAT 게이트웨이(1238)로 송신될 수 있다. 요청에 의해 저장되도록 희망될 수 있는 메모리는 DB 서브넷(들)(1230)에 저장될 수 있다.

[0178] 일부 예들에서, 데이터 평면 미러 앱 계층(1240)은 제어 평면 VCN(1216)과 데이터 평면 VCN(1218) 사이의 직접 통신을 가능하게 할 수 있다. 예를 들어, 구성에 대한 변경들, 업데이트들, 또는 다른 적절한 수정들은 데이터 평면 VCN(1218)에 포함된 자원들에 적용되도록 희망될 수 있다. VNIC(1242)를 통해, 제어 평면 VCN(1216)은 데이터 평면 VCN(1218)에 포함된 자원들과 직접적으로 통신할 수 있으며, 그럼으로써 데이터 평면 VCN에 포함된 자원들에 대해 구성에 대한 변경들, 업데이트들 또는 다른 적절한 수정들을 실행할 수 있다.

[0179] 일부 실시예들에서, 제어 평면 VCN(1216) 및 데이터 평면 VCN(1218)은 서비스 테넌시(1219)에 포함될 수 있다. 이러한 경우에, 시스템의 사용자 또는 고객은 제어 평면 VCN(1216) 또는 데이터 평면 VCN(1218)를 소유하거나 또는 운영하지 않을 수 있다. 대신에, IaaS 제공자가 제어 평면 VCN(1216) 및 데이터 평면 VCN(1218)를 소유하거나 또는 운영할 수 있으며, 이들 둘 모두는 서비스 테넌시(1219)에 포함될 수 있다. 이러한 실시예는, 사용자들 또는 고객들이 다른 사용자들의 또는 다른 고객들의 자원들과 상호작용하는 것을 방지할 수 있는 네트워크들의 격리를 가능하게 할 수 있다. 또한, 이러한 실시예는 시스템의 사용자들 또는 고객들이, 저장을 위해 희망되는 보안 레벨을 갖지 않을 수 있는 공용 인터넷(1254)에 의존할 필요 없이 데이터베이스들을 사적으로 저장하는 것을 가능하게 할 수 있다.

[0180] 다른 실시예들에서, 제어 평면 VCN(1216)에 포함된 LB 서브넷(들)(1222)은 서비스 게이트웨이(1236)로부터 신호를 수신하도록 구성될 수 있다. 이러한 실시예에서, 제어 평면 VCN(1216) 및 데이터 평면 VCN(1218)은, 공용 인터넷(1254)을 호출하지 않고 IaaS 제공자의 고객에 의해 호출되도록 구성될 수 있다. 고객들이 사용하는 데이터베이스(들)가 IaaS 제공자에 의해 제어될 수 있으며 공용 인터넷(1254)으로부터 격리될 수 있는 서비스 테넌시(1219) 상에 저장될 수 있기 때문에, IaaS 제공자의 고객들은 이러한 실시예를 희망할 수 있다.

[0181] 도 13은 적어도 일 실시예에 따른 IaaS 아키텍처의 다른 예시적인 패턴을 예시하는 블록도(1300)이다. 서비스 운영자들(1302)(예를 들어, 도 12의 서비스 운영자들(1202))은, 가상 클라우드 네트워크(VCN)(1306)(예를 들어, 도 12의 VCN(1206)) 및 보안 호스트 서브넷(1308)(예를 들어, 도 12의 보안 호스트 서브넷(1208))을 포함할 수 있는 보안 호스트 테넌시(1304)(예를 들어, 도 12의 보안 호스트 테넌시(1204))에 통신가능하게 결합될 수 있다. VCN(1306)은, SSH VCN(1312)에 포함된 LPG(1310)를 통해 보안 쉘(secure shell; SSH) VCN(1312)(예를 들어, 도 12의 SSH VCN(1212))에 통신가능하게 결합될 수 있는 로컬 피어링 게이트웨이(local peering gateway; LPG)(1310)(예를 들어, 도 12의 LPG(1210))를 포함할 수 있다. SSH VCN(1312)은 SSH 서브넷(1314)(예를 들어, 도 12의 SSH 서브넷(1214))을 포함할 수 있으며, SSH VCN(1312)는 제어 평면 VCN(1316)에 포함된 LPG(1310)를 통해 제어 평면 VCN(1316)(예를 들어, 도 12의 제어 평면 VCN(1216))에 통신가능하게 결합될 수 있다. 제어 평면 VCN(1316)은 서비스 테넌시(1319)(예를 들어, 도 12의 서비스 테넌시(1219))에 포함될 수 있으며, 데이터 평면 VCN(1318)(예를 들어, 도 12의 데이터 평면 VCN(1218))은 시스템의 사용자들 또는 고객들에 의해 소유되거나 또는 운영될 수 있는 고객 테넌시(1321)에 포함될 수 있다.

[0182] 제어 평면 VCN(1316)은, LB 서브넷(들)(1322)(예를 들어, 도 12의 LB 서브넷(들)(1222))을 포함할 수 있는 제어 평면 DMZ 계층(1320)(예를 들어, 도 12의 제어 평면 DMZ 계층(1220)), 앱 서브넷(들)(1326)(예를 들어, 도 12의 앱 서브넷(들)(1226))을 포함할 수 있는 제어 평면 앱 계층(1324)(예를 들어, 도 12의 제어 평면 앱 계층(1224)), 데이터베이스(DB) 서브넷(들)(1330)(예를 들어, 도 12의 DB 서브넷(들)(1230)과 유사함)을 포함할 수 있는 제어 평면 데이터 계층(1328)(예를 들어, 도 12의 제어 평면 데이터 계층(1228))을 포함할 수 있다. 제어 평면 DMZ 계층(1320)에 포함된 LB 서브넷(들)(1322)은 제어 평면 앱 계층(1324)에 포함된 앱 서브넷(들)(1326) 및 제어 평면 VCN(1316)에 포함될 수 있는 인터넷 게이트웨이(1334)(예를 들어, 도 12의 인터넷 게이트웨이(1234))에 통신가능하게 결합될 수 있으며, 앱 서브넷(들)(1326)은 제어 평면 데이터 계층(1328)에 포함된 DB 서브넷(들)(1330), 서비스 게이트 웨이(1336)(예를 들어, 도 12의 서비스 게이트웨이) 및 네트워크 어드레스 변환(NAT) 게이트웨이(1338)(예를 들어, 도 12의 NAT 게이트웨이(1238))에 통신가능하게 결합될 수 있다. 제어 평면 VCN(1316)은 서비스 게이트웨이(1336) 및 NAT 게이트웨이(1338)를 포함할 수 있다.

[0183] 제어 평면 VCN(1316)은 앱 서브넷(들)(1326)을 포함할 수 있는 데이터 평면 미러 앱 계층(1340)(예를 들어, 도 12의 데이터 평면 미러 앱 계층(1240))을 포함할 수 있다. 데이터 평면 미러 앱 계층(1340)에 포함된 앱 서브넷(들)(1326)은 컴퓨팅 인스턴스(1344)(예를 들어, 도 12의 컴퓨팅 인스턴스(1244)와 유사함)를 실행할 수 있는 가상 네트워크 인터페이스 제어기(VNIC)(1342)(예를 들어, VNIC(1242))를 포함할 수 있다. 컴퓨팅 인스턴스(1344)는, 데이터 평면 미러 앱 계층(1340)에 포함된 VNIC(1342) 및 데이터 평면 앱 계층(1346)에 포함된 VNIC(1342)를 통한 데이터 평면 앱 계층(1346)(예를 들어, 도 12의 데이터 평면 앱 계층(1246))에 포함될 수 있는 앱 서브넷(들)(1326)과 데이터 평면 미러 앱 계층(1340)의 앱 서브넷(들)(1326) 사이의 통신을 가능하게 할 수 있다.

[0184] 제어 평면 VCN(1316)에 포함된 인터넷 게이트웨이(1334)는, 공용 인터넷(1354)(예를 들어, 도 12의 공용 인터넷(1254))에 통신가능하게 결합될 수 있는 메타데이터 관리 서비스(1352)(예를 들어, 도 12의 메타데이터 관리 서비스(1252))에 통신가능하게 결합될 수 있다. 공용 인터넷(1354)은 제어 평면 VCN(1316)의 NAT 게이트웨이(1338)에 통신가능하게 결합될 수 있다. 제어 평면 VCN(1316)에 포함된 서비스 게이트웨이(1336)는 클라우드 서비스들(1356)(예를 들어, 도 12의 클라우드 서비스들(1256))에 통신가능하게 결합될 수 있다.

[0185] 일부 예들에서, 데이터 평면 VCN(1318)은 고객 테넌시(1321)에 포함될 수 있다. 이러한 경우에, IaaS 제공자는 각각의 고객에 대해 제어 평면 VCN(1316)를 제공할 수 있으며, IaaS 제공자는, 각각의 고객에 대해, 서비스 테넌시(1319)에 포함된 고유 컴퓨팅 인스턴스(1344)를 셋업할 수 있다. 각각의 컴퓨팅 인스턴스(1344)는 서비스 테넌시(1319)에 포함된 제어 평면 VCN(1316)과 고객 테넌시(1321)에 포함된 데이터 평면 VCN(1318) 사이의 통신을 가능하게 할 수 있다. 컴퓨팅 인스턴스(1344)는 서비스 테넌시(1319)에 포함된 제어 평면 VCN(1316)에 프로비저닝된 자원들이 고객 테넌시(1321)에 포함된 데이터 평면 VCN(1318)에 배포되거나 또는 달리 사용되는 것을 가능하게 할 수 있다.

[0186] 다른 예들에서, IaaS 제공자의 고객은 고객 테넌시(1321)에 거주(live)하는 데이터베이스들을 가질 수 있다. 이러한 예에서, 제어 평면 VCN(1316)은 앱 서브넷(들)(1326)을 포함할 수 있는 데이터 평면 미러 앱 계층(1340)을 포함할 수 있다. 데이터 평면 미러 앱 계층(1340)은 데이터 평면 VCN(1318)에 상주할 수 있지만, 데이터 평면 미러 앱 계층(1340)은 데이터 평면 VCN(1318)에 거주하지 않을 수 있다. 즉, 데이터 평면 미러 앱 계층(1340)은 고객 테넌시(1321)에 대한 액세스를 가질 수 있지만, 데이터 평면 미러 앱 계층(1340)은 데이터 평면 VCN(1318)에 존재하지 않을 수 있거나 또는 IaaS 제공자의 고객에 의해 소유되거나 또는 운영되지 않을 수 있다. 데이터 평면 미러 앱 계층(1340)은 데이터 평면 VCN(1318)에 대한 호출들을 만들도록 구성될 수 있지만, 제어 평면 VCN(1316)에 포함된 엔티티에 대한 호출들을 만들도록 구성되지 않을 수 있다. 고객은 제어 평면 VCN(1316)에 프로비저닝된 데이터 평면 VCN(1318)에서 자원들을 배포하거나 또는 달리 사용하는 것을 희망할 수 있으며, 데이터 평면 미러 앱 계층(1340)은 고객의 자원들의 희망되는 배포 또는 다른 사용을 가능하게 할 수 있다.

[0187] 일부 실시예들에서, IaaS 제공자의 고객은 데이터 평면 VCN(1318)에 필터들을 적용할 수 있다. 이러한 실시예에서, 고객은 데이터 평면 VCN(1318)이 액세스할 수 있는 것을 결정할 수 있고, 고객은 데이터 평면 VCN(1318)으로부터 공용 인터넷(1354)에 대한 액세스를 제한할 수 있다. IaaS 제공자는 임의의 외부 네트워크들 또는 데이터베이스들에 대한 데이터 평면 VCN(1318)의 액세스에 필터를 적용하거나 또는 달리 이를 제어하지 못할 수 있다. 고객 테넌시(1321)에 포함된 데이터 평면 VCN(1318) 상에 고객에 의한 필터들 및 제어들을 적용하는 것은 데이터 평면 VCN(1318)을 다른 고객들 및 공용 인터넷(1354)으로부터 격리시키는 것을 도울 수 있다.

[0188] 일부 실시예들에서, 클라우드 서비스들(1356)은, 공용 인터넷(1354) 상에, 제어 평면 VCN(1316) 상에, 또는 데이터 평면 VCN(1318) 상에 존재하지 않을 수 있는 서비스들을 액세스하기 위해 서비스 게이트웨이(1336)에 의해 호출될 수 있다. 클라우드 서비스들(1356)과 제어 평면 VCN(1316) 또는 데이터 평면 VCN(1318) 사이의 연결은 라이브가 아닐 수 있거나 또는 연속적이지 않을 수 있다. 클라우드 서비스들(1356)은 IaaS 제공자에 의해 소유되거나 또는 운영되는 상이한 네트워크 상에 존재할 수 있다. 클라우드 서비스들(1356)은 서비스 게이트웨이(1336)로부터 호출들을 수신하도록 구성될 수 있으며, 공용 인터넷(1354)으로부터의 호출들을 수신하지 않도록 구성될 수 있다. 일부 클라우드 서비스들(1356)은 다른 클라우드 서비스들(1356)으로부터 격리될 수 있으며, 제어 평면 VCN(1316)은, 제어 평면 VCN(1316)과 동일한 영역에 있지 않을 수 있는 클라우드 서비스들(1356)로부터 격리될 수 있다. 예를 들어, 제어 평면 VCN(1316)은 "영역 1"에 위치될 수 있으며, 클라우드 서비스 "배포 12"는 영역 1 및 "영역 2"에 위치될 수 있다. 배포 12에 대한 호출이 영역 1에 위치된 제어 평면 VCN(1316)에 포함된 서비스 게이트웨이(1336)에 의해 이루어지는 경우, 호출은 영역 1 내의 배포 12로 송신될 수 있다. 이러한 예에서, 제어 평면 VCN(1316) 또는 영역 1 내의 배포 12는 영역 2 내의 배포 12와 통신가능하게 결합되거나 또는 이와 달리 통신하지 않을 수 있다.

[0189] 도 14은 적어도 일 실시예에 따른 IaaS 아키텍처의 다른 예시적인 패턴을 예시하는 블록도(1400)이다. 서비스 운영자들(1402)(예를 들어, 도 12의 서비스 운영자들(1202))은, 가상 클라우드 네트워크(VCN)(1406)(예를 들어, 도 12의 VCN(1206)) 및 보안 호스트 서브넷(1408)(예를 들어, 도 12의 보안 호스트 서브넷(1208))을 포함할 수 있는 보안 호스트 테넌시(1404)(예를 들어, 도 12의 보안 호스트 테넌시(1204))에 통신가능하게 결합될 수 있다. VCN(1406)은, SSH VCN(1412)에 포함된 LPG(1410)를 통해 SSH VCN(1412)(예를 들어, 도 12의 SSH VCN(1212))에 통신가능하게 결합될 수 있는 LPG(1410)(예를 들어, 도 12의 LPG(1210))를 포함할 수 있다. SSH VCN(1412)은 SSH 서브넷(1414)(예를 들어, 도 12의 SSH 서브넷(1214))을 포함할 수 있으며, SSH VCN(1412)는 제어 평면 VCN(1416)에 포함된 LPG(1410)를 통해 제어 평면 VCN(1416)(예를 들어, 도 12의 제어 평면 VCN(1216))에 그리고 데이터 평면 VCN(1418)에 포함된 LPG(1410)를 통해 데이터 평면 VCN(1418)(예를 들어, 도 12의 데이터 평면 VCN(1218))에 통신가능하게 결합될 수 있다. 제어 평면 VCN(1416) 및 데이터 평면 VCN(1418)은 서비스 테넌시(1419)(예를 들어, 도 12의 서비스 테넌시(1219))에 포함될 수 있다.

[0190] 제어 평면 VCN(1416)은, 부하 밸런서(LB) 서브넷(들)(1422)(예를 들어, 도 12의 LB 서브넷(들)(1222))을 포함할 수 있는 제어 평면 DMZ 계층(1420)(예를 들어, 도 12의 제어 평면 DMZ 계층(1220)), 앱 서브넷(들)(1426)(예를 들어, 도 12의 앱 서브넷(들)(1226)과 유사함)을 포함할 수 있는 제어 평면 앱 계층(1424)(예를 들어, 도 12의 제어 평면 앱 계층(1224)), DB 서브넷(들)(1430)을 포함할 수 있는 제어 평면 데이터 계층(1428)(예를 들어, 도 12의 제어 평면 데이터 계층(1228))을 포함할 수 있다. 제어 평면 DMZ 계층(1420)에 포함된 LB 서브넷(들)(1422)은 제어 평면 앱 계층(1424)에 포함된 앱 서브넷(들)(1426) 및 제어 평면 VCN(1416)에 포함될 수 있는 인터넷 게이트웨이(1434)(예를 들어, 도 12의 인터넷 게이트웨이(1234))에 통신가능하게 결합될 수 있으며, 앱 서브넷(들)(1426)은 제어 평면 데이터 계층(1428)에 포함된 DB 서브넷(들)(1430), 서비스 게이트 웨이(1436)(예를 들어, 도 12의 서비스 게이트웨이) 및 네트워크 어드레스 변환(NAT) 게이트웨이(1438)(예를 들어,

도 12의 NAT 게이트웨이(1238))에 통신가능하게 결합될 수 있다. 제어 평면 VCN(1416)은 서비스 게이트웨이(1436) 및 NAT 게이트웨이(1438)를 포함할 수 있다.

[0191] 데이터 평면 VCN(1418)은 데이터 평면 앱 계층(1446)(예를 들어, 도 12의 데이터 평면 앱 계층(1246)), 데이터 평면 DMZ 계층(1448)(예를 들어, 도 12의 데이터 평면 DMZ 계층(1248)), 및 데이터 평면 데이터 계층(1450)(예를 들어, 도 12의 데이터 평면 데이터 계층(1250))을 포함할 수 있다. 데이터 평면 DMZ 계층(1448)은, 데이터 평면 앱 계층(1446)의 신뢰할 수 있는 앱 서브넷(들)(1460)과 신뢰할 수 없는 앱 서브넷(들)(1462), 및 데이터 평면 VCN(1418)에 포함된 인터넷 게이트웨이(1434)에 통신가능하게 결합될 수 있는 LB 서브넷(들)(1422)을 포함할 수 있다. 신뢰할 수 있는 앱 서브넷(들)(1460)은 데이터 평면 VCN(1418)에 포함된 서비스 게이트웨이(1436), 데이터 평면 VCN(1418)에 포함된 NAT 게이트웨이(1438), 및 데이터 평면 데이터 계층(1450)에 포함된 DB 서브넷(들)(1430)에 통신가능하게 결합될 수 있다. 신뢰할 수 없는 앱 서브넷(들)(1462)은 데이터 평면 VCN(1418)에 포함된 서비스 게이트웨이(1436) 및 데이터 평면 데이터 계층(1450)에 포함된 DB 서브넷(들)(1430)에 통신가능하게 결합될 수 있다. 데이터 평면 데이터 계층(1450)은, 데이터 평면 VCN(1418)에 포함된 서비스 게이트웨이(1436)에 통신가능하게 결합될 수 있는 DB 서브넷(들)(1430)을 포함할 수 있다.

[0192] 신뢰할 수 없는 앱 서브넷(들)(1462)은 테넌트(tenant) 가상 머신(VM)들(1466(1)-(N))에 통신가능하게 결합될 수 있는 하나 이상의 1차 VNIC들(1464(1)-(N))을 포함할 수 있다. 각각의 테넌트 VM(1466(1)-(N))은, 개별적인 고객 테넌시들(1470(1)-(N))에 포함될 수 있는 개별적인 컨테이너 출구 VCN들(1468(1)-(N))에 포함될 수 있는 개별적인 앱 서브넷(1467(1)-(N))에 통신가능하게 결합될 수 있다. 개별적인 2차 VNIC들(1472(1)-(N))은 데이터 평면 VCN(1418)에 포함된 신뢰할 수 없는 앱 서브넷(들)(1462)과 컨테이너 출구 VCN들(1468(1)-(N))에 포함된 앱 서브넷 사이의 통신을 가능하게 할 수 있다. 각각의 컨테이너 출구 VCN들(1468(1)-(N))은, 공용 인터넷(1454)(예를 들어, 도 12의 공용 인터넷(1254))에 통신가능하게 결합될 수 있는 NAT 게이트웨이(1438)를 포함할 수 있다.

[0193] 제어 평면 VCN(1416)에 포함되고 데이터 평면 VCN(1418)에 포함된 인터넷 게이트웨이(1434)는, 공용 인터넷(1454)에 통신가능하게 결합될 수 있는 메타데이터 관리 서비스(1452)(예를 들어, 도 12의 메타데이터 관리 시스템(1252))에 통신가능하게 결합될 수 있다. 공용 인터넷(1454)은, 제어 평면 VCN(1416)에 포함되고 데이터 평면 VCN(1418)에 포함된 NAT 게이트웨이(1438)에 통신가능하게 결합될 수 있다. 제어 평면 VCN(1416)에 포함되고 데이터 평면 VCN(1418)에 포함된 서비스 게이트웨이(1436)는 클라우드 서비스들(1456)에 통신가능하게 결합될 수 있다.

[0194] 일부 실시예들에서, 데이터 평면 VCN(1418)은 고객 테넌시들(1470)과 통합될 수 있다. 이러한 통합은, 코드를 실행할 때 지원을 희망할 수 있는 경우와 같은 일부 경우들에서 IaaS 제공자의 고객들에 대해 유용하거나 또는 바람직할 수 있다. 고객은, 파괴적일 수 있거나 또는 다른 고객 자원들과 통신할 수 있거나 또는 달리 바람직하지 않은 효과들을 야기할 수 있는, 실행할 코드를 제공할 수 있다. 이에 응답하여, IaaS 제공자는 고객에 의해 IaaS 제공자에게 주어진 코드를 실행할지 여부를 결정할 수 있다.

[0195] 일부 예들에서, IaaS 제공자의 고객은 IaaS 제공자에게 일시적인 네트워크 액세스를 승인할 수 있으며, 데이터 평면 계층 앱(1446)에 첨부될 기능을 요청할 수 있다. 기능을 실행하기 위한 코드는 VM들(1466(1)-(N))에서 실행될 수 있으며, 코드는 데이터 평면 VCN(1418) 상의 어느 곳에서 실행되도록 구성되지 않을 수 있다. 각각의 VM(1466(1)-(N))은 하나의 고객 테넌시(1470)에 연결될 수 있다. VM들(1466(1)-(N))에 포함된 개별적인 컨테이너들(1471(1)-(N))은 코드를 실행하도록 구성될 수 있다. 이러한 경우에, 이중 격리가 존재할 수 있으며(예를 들어, 코드를 실행하는 컨테이너들(1471(1)-(N)), 여기서 컨테이너들(1471(1)-(N))은 신뢰할 수 없는 앱 서브넷(들)(1462)에 포함된 적어도 하나의 VM(1466(1)-(N))에 포함될 수 있음), 이는 부정확하거나 또는 달리 바람직하지 않은 코드가 IaaS 제공자의 네트워크를 손상시키거나 또는 상이한 고객의 네트워크를 손상시키는 것을 방지하는 것을 도울 수 있다. 컨테이너들(1471(1)-(N))은 고객 테넌시(1470)에 통신가능하게 결합될 수 있으며, 고객 테넌시(1470)로 데이터를 송신하거나 또는 이로부터 데이터를 수신하도록 구성될 수 있다. 컨테이너들(1471(1)-(N))은 데이터 평면 VCN(1418) 내의 임의의 다른 엔티티로 데이터를 송신하거나 또는 이로부터 데이터를 수신하도록 구성되지 않을 수 있다. 코드 실행의 완료 시에, IaaS 제공자는 컨테이너들(1471(1)-(N))을 없애거나(kill) 또는 달리 폐기할 수 있다.

[0196] 일부 실시예들에서, 신뢰할 수 있는 앱 서브넷(들)(1460)은, IaaS 제공자에 의해 소유되거나 또는 운영될 수 있는 코드를 실행할 수 있다. 이러한 실시예에서, 신뢰할 수 있는 앱 서브넷(들)(1460)은 DB 서브넷(들)(1430)에 통신가능하게 결합될 수 있으며, DB 서브넷(들)(1430)에서 CRUD 동작들을 실행하도록 구성될 수 있다. 신뢰할

수 없는 앱 서브넷(들)(1462)은 DB 서브넷(들)(1430)에 통신가능하게 결합될 수 있지만, 이러한 실시예에서, 신뢰할 수 없는 앱 서브넷(들)은 DB 서브넷(들)(1430)에서 판독 동작들을 실행하도록 구성될 수 있다. 각각의 고객의 VM(1466(1)-(N))에 포함될 수 있으며 고객으로부터의 코드를 실행할 수 있는 컨테이너들(1471(1)-(N))은 DB 서브넷(들)(1430)과 통신가능하게 결합되지 않을 수 있다.

[0197] 다른 실시예들에서, 제어 평면 VCN(1416) 및 데이터 평면 VCN(1418)은 직접적으로 통신가능하게 결합되지 않을 수 있다. 이러한 실시예에서, 제어 평면 VCN(1416)과 데이터 평면 VCN(1418) 사이의 직접 통신이 존재하지 않을 수 있다. 그러나, 통신은 적어도 하나의 방법을 통해 간접적으로 발생할 수 있다. 제어 평면 VCN(1416)과 데이터 평면 VCN(1418) 사이의 통신을 가능하게 할 수 있는 LPG(1410)가 IaaS 제공자에 의해 수립될 수 있다. 다른 예에서, 제어 평면 VCN(1416) 또는 데이터 평면 VCN(1418)은 서비스 게이트웨이(1436)을 통해 클라우드 서비스들(1456)에 대한 호출을 만들 수 있다. 예를 들어, 제어 평면 VCN(1416)으로부터 클라우드 서비스들(1456)으로의 호출은, 데이터 평면 VCN(1418)과 통신할 수 있는 서비스에 대한 요청을 포함할 수 있다.

[0198] 도 15는 적어도 일 실시예에 따른 IaaS 아키텍처의 다른 예시적인 패턴을 예시하는 블록도(1500)이다. 서비스 운영자들(1502)(예를 들어, 도 12의 서비스 운영자들(1202))은, 가상 클라우드 네트워크(VCN)(1506)(예를 들어, 도 12의 VCN(1206)) 및 보안 호스트 서브넷(1508)(예를 들어, 도 12의 보안 호스트 서브넷(1208))을 포함할 수 있는 보안 호스트 테넌시(1504)(예를 들어, 도 12의 보안 호스트 테넌시(1204))에 통신가능하게 결합될 수 있다. VCN(1506)은, SSH VCN(1512)에 포함된 LPG(1510)를 통해 SSH VCN(1512)(예를 들어, 도 12의 SSH VCN(1212))에 통신가능하게 결합될 수 있는 LPG(1510)(예를 들어, 도 12의 LPG(1210))를 포함할 수 있다. SSH VCN(1512)은 SSH 서브넷(1514)(예를 들어, 도 12의 SSH 서브넷(1214))을 포함할 수 있으며, SSH VCN(1512)는 제어 평면 VCN(1516)에 포함된 LPG(1510)를 통해 제어 평면 VCN(1516)(예를 들어, 도 12의 제어 평면 VCN(1216))에 그리고 데이터 평면 VCN(1518)에 포함된 LPG(1510)를 통해 데이터 평면 VCN(1518)(예를 들어, 도 12의 데이터 평면 VCN(1218))에 통신가능하게 결합될 수 있다. 제어 평면 VCN(1516) 및 데이터 평면 VCN(1518)은 서비스 테넌시(1519)(예를 들어, 도 12의 서비스 테넌시(1219))에 포함될 수 있다.

[0199] 제어 평면 VCN(1516)은, LB 서브넷(들)(1522)(예를 들어, 도 12의 LB 서브넷(들)(1222))을 포함할 수 있는 제어 평면 DMZ 계층(1520)(예를 들어, 도 12의 제어 평면 DMZ 계층(1220)), 앱 서브넷(들)(1526)(예를 들어, 도 12의 앱 서브넷(들)(1226))을 포함할 수 있는 제어 평면 앱 계층(1524)(예를 들어, 도 12의 제어 평면 앱 계층(1224)), DB 서브넷(들)(1530)(예를 들어, 도 14의 DB 서브넷(들)(1430))을 포함할 수 있는 제어 평면 데이터 계층(1528)(예를 들어, 도 12의 제어 평면 데이터 계층(1228))을 포함할 수 있다. 제어 평면 DMZ 계층(1520)에 포함된 LB 서브넷(들)(1522)은 제어 평면 앱 계층(1524)에 포함된 앱 서브넷(들)(1526) 및 제어 평면 VCN(1516)에 포함될 수 있는 인터넷 게이트웨이(1534)(예를 들어, 도 12의 인터넷 게이트웨이(1234))에 통신가능하게 결합될 수 있으며, 앱 서브넷(들)(1526)은 제어 평면 데이터 계층(1528)에 포함된 DB 서브넷(들)(1530), 서비스 게이트 웨이(1536)(예를 들어, 도 12의 서비스 게이트웨이) 및 네트워크 어드레스 변환(NAT) 게이트웨이(1538)(예를 들어, 도 12의 NAT 게이트웨이(1238))에 통신가능하게 결합될 수 있다. 제어 평면 VCN(1516)은 서비스 게이트웨이(1536) 및 NAT 게이트웨이(1538)를 포함할 수 있다.

[0200] 데이터 평면 VCN(1518)은 데이터 평면 앱 계층(1546)(예를 들어, 도 12의 데이터 평면 앱 계층(1246)), 데이터 평면 DMZ 계층(1548)(예를 들어, 도 12의 데이터 평면 DMZ 계층(1248)), 및 데이터 평면 데이터 계층(1550)(예를 들어, 도 12의 데이터 평면 데이터 계층(1250))을 포함할 수 있다. 데이터 평면 DMZ 계층(1548)은, 데이터 평면 앱 계층(1546)의 신뢰할 수 있는 앱 서브넷(들)(1560)(예를 들어, 도 14의 신뢰할 수 있는 앱 서브넷(들)(1460))과 신뢰할 수 없는 앱 서브넷(들)(1562)(예를 들어, 도 14의 신뢰할 수 없는 앱 서브넷(들)(1462)), 및 데이터 평면 VCN(1518)에 포함된 인터넷 게이트웨이(1534)에 통신가능하게 결합될 수 있는 LB 서브넷(들)(1522)을 포함할 수 있다. 신뢰할 수 있는 앱 서브넷(들)(1560)은 데이터 평면 VCN(1518)에 포함된 서비스 게이트웨이(1536), 데이터 평면 VCN(1518)에 포함된 NAT 게이트웨이(1538), 및 데이터 평면 데이터 계층(1550)에 포함된 DB 서브넷(들)(1530)에 통신가능하게 결합될 수 있다. 신뢰할 수 없는 앱 서브넷(들)(1562)은 데이터 평면 VCN(1518)에 포함된 서비스 게이트웨이(1536) 및 데이터 평면 데이터 계층(1550)에 포함된 DB 서브넷(들)(1530)에 통신가능하게 결합될 수 있다. 데이터 평면 데이터 계층(1550)은, 데이터 평면 VCN(1518)에 포함된 서비스 게이트웨이(1536)에 통신가능하게 결합될 수 있는 DB 서브넷(들)(1530)을 포함할 수 있다.

[0201] 신뢰할 수 없는 앱 서브넷(들)(1562)은, 신뢰할 수 없는 앱 서브넷(들)(1562) 내에 상주하는 테넌트 가상 머신(VM)들(1566(1)-(N))에 통신가능하게 결합될 수 있는 1차 VNIC들(1564(1)-(N))을 포함할 수 있다. 각각의 테넌트 VM(1566(1)-(N))은 개별적인 컨테이너(1567(1)-(N))에서 코드를 실행할 수 있으며, 컨테이너 출구 VCN(1568)에 포함될 수 있는 데이터 평면 앱 계층(1546)에 포함될 수 있는 앱 서브넷(1526)에 통신가능하게 결

합될 수 있다. 개별적인 2차 VNIC들(1572(1)-(N))은 데이터 평면 VCN(1518)에 포함된 신뢰할 수 없는 앱 서브넷(들)(1562)과 컨테이너 출구 VCN들(1568)에 포함된 앱 서브넷 사이의 통신을 가능하게 할 수 있다. 컨테이너 출구 VCN은, 공용 인터넷(1554)(예를 들어, 도 12의 공용 인터넷(1254))에 통신가능하게 결합될 수 있는 NAT 게이트웨이(1538)를 포함할 수 있다.

[0202] 제어 평면 VCN(1516)에 포함되고 데이터 평면 VCN(1518)에 포함된 인터넷 게이트웨이(1534)는, 공용 인터넷(1554)에 통신가능하게 결합될 수 있는 메타데이터 관리 서비스(1552)(예를 들어, 도 12의 메타데이터 관리 시스템(1252))에 통신가능하게 결합될 수 있다. 공용 인터넷(1554)은, 제어 평면 VCN(1516)에 포함되고 데이터 평면 VCN(1518)에 포함된 NAT 게이트웨이(1538)에 통신가능하게 결합될 수 있다. 제어 평면 VCN(1516)에 포함되고 데이터 평면 VCN(1518)에 포함된 서비스 게이트웨이(1536)는 클라우드 서비스들(1556)에 통신가능하게 결합될 수 있다.

[0203] 일부 예들에서, 도 15의 블록도(1500)의 아키텍처에 의해 예시된 패턴은 도 14의 블록도(1400)의 아키텍처에 의해 예시된 패턴에 대한 예외로서 간주될 수 있으며, IaaS 제공자가 고객과 직접적으로 통신할 수 없는 경우(예를 들어, 분리된 영역) IaaS 제공자의 고객에게 바람직할 수 있다. 각각의 고객에 대한 VM들(1566(1)-(N))에 포함된 개별적인 컨테이너들(1567(1)-(N))은 고객에 의해 실시간으로 액세스될 수 있다. 컨테이너들(1567(1)-(N))은, 컨테이너 출구 VCN(1568)에 포함될 수 있는 데이터 패널 앱 계층(1546)의 앱 서브넷(들)(1526)에 포함된 개별적인 2차 VNIC들(1572(1)-(N))에 대한 호출들을 만들도록 구성될 수 있다. 2차 VNIC들(1572(1)-(N))은, 공용 인터넷(1554)으로 호출들을 송신할 수 있는 NAT 게이트웨이(1538)로 호출들을 송신할 수 있다. 이러한 예에서, 고객에 의해 실시간으로 액세스될 수 있는 컨테이너들(1567(1)-(N))은 제어 평면 VCN(1516)으로부터 격리될 수 있고, 데이터 평면 VCN(1518)에 포함된 다른 엔티티들로부터 격리될 수 있다. 컨테이너들(1567(1)-(N))은 또한 다른 고객들의 자원들로부터 격리될 수 있다.

[0204] 다른 예들에서, 고객은 클라우드 서비스들(1556)을 호출하기 위해 컨테이너들(1567(1)-(N))을 사용할 수 있다. 이러한 예에서, 고객은, 클라우드 서비스들(1556)으로부터 서비스를 요청하는 컨테이너들(1567(1)-(N))에서 코드를 실행할 수 있다. 컨테이너들(1567(1)-(N))은, 요청을 공용 인터넷(1554)으로 송신할 수 있는 NAT 게이트웨이로 요청을 송신할 수 있는 2차 VNIC들(1572(1)-(N))로 이러한 요청을 송신할 수 있다. 공용 인터넷(1554)은 요청을 인터넷 게이트웨이(1534)를 통해 제어 평면 VCN(1516)에 포함된 LB 서브셋(들)(1522)으로 송신할 수 있다. 요청이 유효하다고 결정하는 것에 응답하여, LB 서브넷(들)은, 서비스 게이트웨이(1536)를 통해 클라우드 서비스들(1556)로 요청을 송신할 수 있는 앱 서브넷(들)(1526)로 요청을 송신할 수 있다.

[0205] 도면들에 도시된 IaaS 아키텍처들(1200, 1300, 1400, 1500)이 도시된 것들과는 다른 구성요소들을 가질 수 있다는 것이 이해되어야 한다. 추가로, 도면에 도시된 실시예는 본 발명의 일 실시예를 통합할 수 있는 클라우드 인프라스트럭처 시스템의 단지 일부 예들이다. 일부 다른 실시예들에서, IaaS 시스템들은 도면들에 도시된 것보다 더 많거나 또는 더 적은 구성요소들을 가질 수 있거나, 2개 이상의 구성요소들을 결합할 수 있거나, 또는 구성요소들의 상이한 구성 또는 배열을 가질 수 있다.

[0206] 특정 실시예들에서, 본원에서 설명된 IaaS 시스템들은, 셀프-서비스(self-service)의, 가입-기반(subscription-based)의, 탄력적 스케일러블(scalable)의, 신뢰할 수 있는, 고도로 이용가능하며, 안전한 방식으로 고객에게 전달되는 애플리케이션들, 미들웨어, 및 데이터베이스 서비스 제공(offering)들의 스위트(suite)를 포함할 수 있다. 이러한 IaaS 시스템의 일 예는 본 양수인에 의해 제공되는 오라클 클라우드 인프라스트럭처(Oracle Cloud Infrastructure; OCI)이다.

[0207] 도 16은, 다양한 실시예들이 구현될 수 있는 예시적인 컴퓨터 시스템(1600)을 예시한다. 시스템(1600)은 이상에서 설명된 컴퓨터 시스템들 중 임의의 것을 구현하기 위하여 사용될 수 있다. 도면에 도시된 바와 같이, 컴퓨터 시스템(1600)은 버스 서브시스템(1602)을 통해 복수의 주변 서브시스템들과 통신하는 프로세싱 유닛(1604)을 포함한다. 이러한 주변 서브시스템들은 프로세싱 가속 유닛(1606), I/O 서브시스템(1608), 스토리지 서브시스템(1618) 및 통신 서브시스템(1624)을 포함할 수 있다. 스토리지 서브시스템(1618)은 유형적인 컴퓨터-판독가능 저장 매체(1622) 및 시스템 메모리(1610)를 포함한다.

[0208] 버스 서브시스템(1602)은 컴퓨터 시스템(1600)의 다양한 구성요소들 및 서브시스템들이 의도된 바와 같이 서로 통신하는 것을 가능하게 하기 위한 메커니즘을 제공한다. 버스 서브시스템(1602)이 단일 버스로서 개략적으로 도시되었지만, 버스 서브시스템의 대안적인 실시예들은 복수의 버스들을 사용할 수 있다. 버스 서브시스템(1602)은, 다양한 버스 아키텍처들 중 임의의 것을 사용하는 메모리 버스 또는 메모리 제어기, 주변기기 버스, 및 로컬 버스를 포함하는 몇몇 유형들의 버스 구조들 중 임의의 버스 구조일 수 있다. 예를 들어, 이러한 아키

텍처들은, 산업 표준 아키텍처(Industry Standard Architecture; ISA) 버스, 마이크로 채널 아키텍처(Micro Channel Architecture; MCA) 버스, 개량 ISA(Enhanced ISA; EISA) 버스, 비디오 전자공학 표준 위원회(Video Electronics Standards Association; VESA) 로컬 버스, 및 주변 구성요소 상호연결(Peripheral Component Interconnect; PCI) 버스를 포함할 수 있으며, 이들은 IEEE P1386.1 표준에 대하여 제조되는 메자닌 버스(Mezzanine bus)로서 구현될 수 있다.

[0209] 하나 이상의 집적 회로들(예를 들어, 통상적인 마이크로프로세서 또는 마이크로제어기)로서 구현될 수 있는 프로세싱 유닛(1604)은 컴퓨터 시스템(1600)의 동작을 제어한다. 하나 이상의 프로세서들이 프로세싱 유닛(1604) 내에 포함될 수 있다. 이러한 프로세서들은 단일 코어 또는 다중코어 프로세서들을 포함할 수 있다. 특정 실시예들에 있어서, 프로세싱 유닛(1604)은 각각의 프로세싱 유닛 내에 포함된 단일 또는 다중코어 프로세서들을 갖는 하나 이상의 독립적인 프로세싱 유닛들(1632 및/또는 1634)로서 구현될 수 있다. 다른 실시예들에 있어서, 프로세싱 유닛(1604)은 또한 2개의 듀얼-코어 프로세서들을 단일 칩 내에 통합함으로써 형성된 쿼드-코어 프로세싱 유닛으로서 구현될 수 있다.

[0210] 다양한 실시예들에 있어서, 프로세싱 유닛(1604)은 프로그램 코드에 응답하여 다양한 프로그램들을 실행할 수 있으며, 프로그램들 또는 프로세스들의 동시 다중 실행을 유지할 수 있다. 임의의 주어진 시점에, 실행될 프로그램 코드의 전부 또는 일부가 프로세서(들)(1604) 내에 및/또는 스토리지 서브시스템(1618) 내에 상주할 수 있다. 적절한 프로그래밍을 통하여, 프로세서(들)(1604)는 이상에서 설명된 다양한 기능들을 제공할 수 있다. 컴퓨터 시스템(1600)은, 디지털 신호 프로세서(digital signal processor; DSP), 특수-목적 프로세서, 및/또는 유사한 것을 포함할 수 있는 프로세싱 가속 유닛(1606)을 추가적으로 포함할 수 있다.

[0211] I/O 서브시스템(1608)은 사용자 인터페이스 입력 디바이스들 및 사용자 인터페이스 출력 디바이스들을 포함할 수 있다. 사용자 인터페이스 입력 디바이스들은, 키보드, 포인팅 디바이스들 예컨대 마우스 또는 트랙볼, 터치패드 또는 디스플레이 내에 통합된 터치 스크린, 스크롤 휠, 클릭 휠, 다이얼, 버튼, 스위치, 키패드, 음성 명령 인식 시스템들을 가진 오디오 입력 디바이스들, 마이크들, 및 다른 유형들의 입력 디바이스들을 포함할 수 있다. 사용자 인터페이스 입력 디바이스들은, 예를 들어, 사용자들이 제스처들 및 구두(spoken) 명령들을 사용하는 자연스러운 사용자 인터페이스를 통해, Microsoft Xbox® 360 게임 제어기와 같은, 입력 디바이스를 제어하고 이와 상호작용하는 것을 가능하게 하는 Microsoft Kinect® 모션 센서와 같은 모션 센싱 및/또는 제스처 인식 디바이스들을 포함할 수 있다. 사용자 인터페이스 입력 디바이스들은 또한, 사용자들로부터 눈 움직임(eye activity)(예를 들어, 사진을 찍는 동안의 및/또는 메뉴를 선택하는 동안의 '깜박임')을 검출하고 눈 제스처들을 입력 디바이스(예를 들어, Google Glass®) 내로의 입력으로서 변환하는 Google Glass® 눈 깜박임 검출기와 같은 눈 제스처 인식 디바이스들을 포함할 수 있다. 추가적으로, 사용자 인터페이스 입력 디바이스들은, 사용자들이 음성 명령들을 통하여 음성 인식 시스템(예를 들어, Siri® 네비게이터(navigator))과 상호작용하는 것을 가능하게 하는 음성 인식 센싱 디바이스들을 포함할 수 있다.

[0212] 사용자 인터페이스 입력 디바이스들은 또한, 비제한적으로, 3차원(3D) 마우스들, 조이스틱들 또는 포인팅 스틱들, 게임패드들 및 그래픽 태블릿들, 및 음향/시각 디바이스들 예컨대 스피커들, 디지털 카메라들, 디지털 캠코더들, 휴대용 매체 플레이어들, 웹캠들, 이미지 스캐너들, 핑거프린트 스캐너들, 바코드 리더 3D 스캐너들, 3D 프린터들, 레이저 거리계들, 및 시선 추적 디바이스들을 포함할 수 있다. 추가적으로, 사용자 인터페이스 입력 디바이스들은, 예를 들어, 의료 이미징 입력 디바이스들 예컨대 컴퓨터 단층촬영, 자기 공명 이미징, 양전자 방출 단층촬영, 의료 초음파 검사 디바이스들을 포함할 수 있다. 사용자 인터페이스 입력 디바이스들은 또한, 예를 들어, 오디오 입력 디바이스들 예컨대 MIDI 키보드들, 디지털 악기들, 및 유사한 것을 포함할 수 있다.

[0213] 사용자 인터페이스 출력 디바이스들은 디스플레이 서브시스템, 표시등들, 또는 비-시각적 디스플레이들 예컨대 오디오 출력 디바이스들, 등을 포함할 수 있다. 디스플레이 서브시스템은 음극선관(cathode ray tube; CRT), 액정 디스플레이(liquid crystal display; LCD) 또는 플라즈마 디스플레이를 사용하는 것과 같은 평면-패널 디바이스, 프로젝션 디바이스, 터치 스크린, 및 유사한 것일 수 있다. 일반적으로, 용어 "출력 디바이스"의 사용은 컴퓨터 시스템(1600)으로부터 사용자 또는 다른 컴퓨터로 정보를 출력하기 위한 모든 가능한 유형들의 디바이스들 및 메커니즘들을 포함하도록 의도된다. 예를 들어, 사용자 인터페이스 출력 디바이스들은, 비제한적으로, 시각적으로 텍스트, 그래픽들 및 오디오/비디오 정보를 전달하는 다양한 디스플레이 디바이스들, 예컨대 모니터들, 프린터들, 스피커들, 헤드폰들, 자동차 네비게이션 시스템들, 플로터(plotter)들, 음성 출력 디바이스들, 및 모뎀들을 포함할 수 있다.

[0214] 컴퓨터 시스템(1600)은, 시스템 메모리(1610) 내에 현재 위치되어 있는 것으로 도시되는 소프트웨어 요소들을

포함하는 스토리지 서브시스템(1618)을 포함할 수 있다. 시스템 메모리(1610)는, 프로세싱 유닛(1604) 상에 로딩가능하며 실행가능한 프로그램 명령어들뿐만 아니라 이러한 프로그램들의 실행 동안 생성되는 데이터를 저장할 수 있다.

[0215] 컴퓨터 시스템(1600)의 구성 및 유형에 따라서, 시스템 메모리(1610)는 (랜덤 액세스 메모리(random access memory; RAM)와 같은) 휘발성일 수 있거나 및/또는 (판독-전용 메모리(read-only memory; ROM), 플래시 메모리, 등과 같은) 비-휘발성일 수 있다. RAM은 전형적으로 프로세싱 유닛(1604)에 의해 현재 실행되고 동작되고 있거나 및/또는 이에 즉시 액세스가능한 데이터 및/또는 프로그램 모듈들을 포함한다. 일부 구현예들에 있어서, 시스템 메모리(1610)는 복수의 상이한 유형들의 메모리들, 예컨대 정적 랜덤 액세스 메모리(static random access memory; SRAM) 또는 동적 랜덤 액세스 메모리(dynamic random access memory; DRAM)를 포함할 수 있다. 일부 구현예들에 있어서, 예컨대 기동 동안에 컴퓨터 시스템(1600) 내의 요소들 사이에서 정보를 전송하는 것을 돕는 기본 루틴들을 포함하는 기본 입력/출력 시스템(basic input/output system; BIOS)은 전형적으로 ROM 내에 저장될 수 있다. 예로서 그리고 비제한적으로, 시스템 메모리(1610)는 또한, 클라이언트 애플리케이션들, 웹 브라우저들, 중간-계층 애플리케이션들, 관계형 데이터 베이스 관리 시스템(relational database management system; RDBMS)들, 등을 포함할 수 있는 애플리케이션 프로그램들(1612), 프로그램 데이터(1614), 및 운영 체제(1616)을 예시한다. 예로서, 운영 체제(1616)는, 다양한 버전들의 Microsoft Windows®, Apple Macintosh®, 및/또는 리눅스 운영 체제들, (비제한적으로 다양한 GNU/리눅스 운영 체제들, Google Chrome® OS, 및 유사한 것을 포함하는) 다양한 상용-이용가능 UNIX® 또는 UNIX-유사 운영 체제들 및/또는 모바일 운영 체제들 예컨대 iOS, Windows® Phone, Android® OS, BlackBerry® 16 OS, 및 Palm® OS 운영 체제들을 포함할 수 있다.

[0216] 스토리지 서브시스템(1618)은 또한 일부 실시예들의 기능을 제공하는 기본 프로그래밍 및 데이터 구성물들을 저장하기 위한 유형의 컴퓨터-판독가능 저장 매체를 제공할 수 있다. 프로세서에 의해 실행될 때 이상에서 설명된 기능을 제공하는 소프트웨어(프로그램들, 코드 모듈들, 명령어들)가 스토리지 서브시스템(1618) 내에 저장될 수 있다. 이러한 소프트웨어 모듈들 또는 명령어들이 프로세싱 유닛(1604)에 의해 실행될 수 있다. 스토리지 서브시스템(1618)은 또한 본 개시에 따라 사용되는 데이터를 저장하기 위한 저장소를 제공할 수 있다.

[0217] 스토리지 서브시스템(1600)은 또한, 추가적으로 컴퓨터-판독가능 저장 매체(1622)에 연결될 수 있는 컴퓨터-판독가능 저장 매체 리더(1620)를 포함할 수 있다. 시스템 메모리(1610)와 함께 그리고 선택적으로 이와 조합되어, 컴퓨터-판독가능 저장 매체(1622)는, 컴퓨터-판독가능 정보를 일시적으로 및/또는 더 영구적으로 포함하고, 저장하며, 송신하고, 및 검색하기 위한 저장 매체들에 더하여 원격, 로컬, 고정, 및/또는 착탈가능 저장 디바이스들을 포괄적으로 나타낼 수 있다. 비-일시적 컴퓨터-판독가능 저장 매체는 휘발성 메모리 저장 디바이스들 및/또는 비-휘발성 저장 디바이스들을 포함하는 물리적으로 유형적인 메모리 또는 저장 디바이스들을 포함할 수 있다. 비-일시적 컴퓨터-판독가능 저장 매체의 예들은 자기 저장 매체(예를 들어, 디스크 또는 테이프들), 광학 저장 매체(예를 들어, DVD들, CD들), 다양한 유형들의 RAM, ROM, 또는 플래시 메모리, 하드 드라이브들, 플로피 드라이브들, 분리가능 메모리 드라이브들(예를 들어, USB 드라이브들), 또는 다른 유형의 저장 디바이스를 포함한다.

[0218] 코드, 또는 코드의 부분들을 포함하는 컴퓨터-판독가능 저장 매체(1622)는 또한, 비제한적으로, 정보의 저장 및/또는 송신을 위하여 임의의 방법 또는 기술로 구현된 휘발성 및 비-휘발성, 착탈가능 및 비-착탈가능 매체와 같은 저장 매체 및 통신 매체를 포함하는 당업계에서 알려지거나 또는 사용되는 임의의 적절한 매체를 포함할 수 있다. 이는 유형의 컴퓨터-판독가능 저장 매체, 예컨대 RAM, ROM, 전기 소거가능 프로그램가능 ROM(electronically erasable programmable ROM; EEPROM), 플래시 메모리 또는 다른 메모리 기술품, CD-ROM, 디지털 다기능 디스크(digital versatile disk; DVD), 또는 광 저장장치, 자기 카세트들, 자기 테이프, 자기 디스크 저장장치 또는 다른 자기 저장 디바이스들, 또는 다른 유형의 컴퓨터 판독가능 매체를 포함할 수 있다.

[0219] 예로서, 컴퓨터-판독가능 저장 매체(1622)는, 비-착탈가능 비휘발성 자기 매체로부터 판독하거나 또는 이에 기입하는 하드 디스크 드라이브, 착탈가능 비휘발성 자기 디스크로부터 판독하거나 또는 이에 기입하는 자기 디스크 드라이브, 및 착탈가능 비휘발성 광 디스크 예컨대 CD ROM, DVD, 및 Blu-Ray® 디스크, 또는 다른 광 매체로부터 판독하거나 또는 이에 기입하는 광 디스크 드라이브를 포함할 수 있다. 컴퓨터-판독가능 저장 매체(1622)는, 비제한적으로, Zip® 드라이브들, 플래시 메모리 카드들, 범용 직렬 버스(universal serial bus; USB) 플래시 드라이브들, 보안 디지털(secure digital; SD) 카드들, DVD 디스크들, 디지털 비디오 테이프, 및 유사한 것을 포함할 수 있다. 컴퓨터-판독가능 저장 매체(1622)는 또한, 비-휘발성 메모리 기반 고체-상태 드라이브(solid-state drive; SSD)들 예컨대 플래시-메모리 기반 SSD들, 기업 플래시 드라이브들, 고체 상태 ROM, 및 유사한 것, 휘발성 메모리 기반 SSD들 예컨대 고체 상태 RAM, 동적 RAM, 정적 RAM, DRAM-기반 SSD들, 자기저항

성 RAM(magnetoresistive RAM; MRAM) SSD들, 및 DRAM 및 플래시 메모리 기반 SSD들의 조합을 사용하는 하이브리드 SSD들을 포함할 수 있다. 디스크 드라이브들 및 그들의 연관된 컴퓨터-판독가능 매체는 컴퓨터-판독가능 명령어들, 데이터 구조들, 프로그램 모듈들, 및 컴퓨터 시스템(1600)에 대한 다른 데이터의 비-휘발성 저장을 제공할 수 있다.

[0220] 통신 서브시스템(1624)은 다른 컴퓨터 시스템들 및 네트워크들에 대한 인터페이스를 제공한다. 통신 서브시스템(1624)은 컴퓨터 시스템(1600)으로부터 다른 시스템들로 데이터를 송신하고 이로부터 데이터를 수신하기 위한 인터페이스로서 역할한다. 예를 들어, 통신 서브시스템(1624)은 컴퓨터 시스템(1600)이 인터넷을 통해 하나 이상의 디바이스들에 연결하는 것을 가능하게 할 수 있다. 일부 실시예들에 있어서, 통신 서브시스템(1624)은 (예를 들어, 셀룰러 전화기 기술, 진보된 데이터 네트워크 기술, 예컨대 3G, 4G 또는 EDGE(enhanced data rates for global evolution), WiFi(IEEE 802.11 패밀리 표준들, 또는 다른 모바일 통신 기술들, 또는 이들의 임의의 조합)을 사용하여) 무선 음성 및/또는 데이터 네트워크들에 액세스하기 위한 라디오 주파수(radio frequency; RF) 트랜시버 구성요소들, 위성 위치확인 시스템(global positioning system; GPS) 수신기 구성요소들, 및/또는 다른 구성요소들을 포함할 수 있다. 일부 실시예들에 있어서, 통신 서브시스템(1624)은 무선 인터페이스에 더하여 또는 그 대신에 유선 네트워크 연결(예를 들어, 이더넷)을 제공할 수 있다.

[0221] 일부 실시예들에 있어서, 통신 서브시스템(1624)은 또한 컴퓨터 시스템(1600)을 사용할 수 있는 하나 이상의 사용자들을 대표하여 구조화된 및/또는 구조화되지 않은 데이터 피드들(1626), 이벤트 스트림들(1628), 이벤트 업데이트들(1630), 및 그와 유사한 것의 형태의 입력 통신을 수신할 수 있다.

[0222] 예로서, 통신 서브시스템(1624)은 소셜 네트워크들 및/또는 다른 통신 서비스들의 사용자들로부터의 실-시간 데이터 피드들(1626) 예컨대 Twitter® 피드들, Facebook® 업데이트들, 웹 피드들 예컨대 리치 사이트 서머리(Rich Site Summary; RSS) 피드들, 및/또는 하나 이상의 제3 자 정보 소스들로부터의 실-시간 업데이트들을 수신하도록 구성될 수 있다.

[0223] 추가적으로, 통신 서브시스템(1624)은 또한, 사실상 명시적인 종료를 갖지 않는 제한이 없거나 또는 연속적일 수 있는 실-시간 이벤트들 및/또는 이벤트 업데이트들(1630)의 이벤트 스트림들(1628)을 포함할 수 있는 연속적인 데이터 스트림들의 형태로 데이터를 수신하도록 구성될 수 있다. 연속적인 데이터를 생성하는 애플리케이션들의 예들은, 예를 들어, 센서 데이터 애플리케이션들, 금융 시세표시기들, 네트워크 성능 측정 툴들(예를 들어, 네트워크 모니터링 및 트래픽 관리 애플리케이션들), 클릭스트림 분석 툴들, 자동차 트래픽 모니터링, 및 유사한 것을 포함할 수 있다.

[0224] 통신 서브시스템(1624)은 또한, 구조화되거나 및/또는 구조화되지 않은 데이터 피드들(1626), 이벤트 스트림들(1628), 이벤트 업데이트들(1630), 및 유사한 것을 컴퓨터 시스템(1600)에 결합된 하나 이상의 스트리밍 데이터 소스 컴퓨터들과 통신할 수 있는 하나 이상의 데이터베이스들로 출력하도록 구성될 수 있다.

[0225] 컴퓨터 시스템(1600)은, 핸드헬드 휴대용 디바이스(예를 들어, iPhone® 셀룰러 폰, iPad® 컴퓨팅 태블릿, PDA), 웨어러블 디바이스(예를 들어, Google Glass® 머리 착용형 디스플레이), PC, 워크스테이션, 메인프레임, 키오스크, 서버 랙, 또는 임의의 다른 데이터 프로세싱 시스템을 포함하는 다양한 유형들 중 임의의 유형일 수 있다.

[0226] 컴퓨터들 및 네트워크들의 계속해서 변화하는 성질에 기인하여, 도면에 도시된 컴퓨터 시스템(1600)의 설명은 오로지 특정한 일 예로서만 의도된다. 도면에 도시된 시스템보다 더 많거나 또는 더 적은 구성요소들을 갖는 다수의 다른 구성들이 가능하다. 예를 들어, 커스텀화된 하드웨어가 또한 사용될 수 있거나 및/또는 특정 요소들이 하드웨어, 펌웨어, (애플릿(applet)들을 포함하는) 소프트웨어, 또는 이들의 조합으로 구현될 수 있다. 추가로, 네트워크 입력/출력 디바이스들과 같은 다른 컴퓨팅 디바이스들에 대한 연결이 이용될 수 있다. 본원에 제공되는 개시 및 교시들에 기초하여, 당업자는 다양한 실시예들을 구현하기 위한 다른 방식들 및/또는 방법들을 인식할 것이다.

[0227] 특정 실시예들이 설명되었지만, 다양한 수정예들, 대안예들, 대안적인 구성들, 및 등가물들이 또한 본 개시의 범위 내에 포괄된다. 실시예들은 정확한 특정 데이터 프로세싱 환경들 내에서 동작하도록 제한되는 것이 아니라, 복수의 데이터 프로세싱 환경들 내에서 자유롭게 동작할 수 있다. 추가적으로, 실시예들이 특정한 일련의 트랜잭션(transaction)들 및 단계들을 사용하여 설명되었지만, 본 개시의 범위가 설명된 일련의 트랜잭션들 및 단계들로 한정되지 않는다는 것이 당업자들에게 명백할 것이다. 이상에서 설명된 실시예들의 다양한 특징들 및 측면들이 개별적으로 또는 함께 사용될 수 있다.

[0228] 추가로, 실시예들이 하드웨어 및 소프트웨어의 특정한 조합을 사용하여 설명되었지만, 하드웨어 및 소프트웨어의 다른 조합들이 또한 본 개시의 범위 내에 있다는 것이 이해되어야 한다. 실시예들은 오로지 하드웨어로만, 또는 오로지 소프트웨어로만, 또는 이들의 조합들을 사용하여 구현될 수 있다. 본 명세서에서 설명된 다양한 프로세스들은 동일한 프로세서 또는 임의의 조합의 상이한 프로세서들 상에 구현될 수 있다. 따라서, 구성요소들 또는 모듈들이 특정 동작들을 수행하도록 구성된 것으로 설명되는 경우, 이러한 구성은, 예를 들어, 동작을 수행하기 위한 전자 회로들을 설계함으로써, 동작을 수행하기 위하여 프로그램가능 전자 회로들(예컨대 마이크로프로세서들)을 프로그래밍함으로써, 또는 이들의 임의의 조합에 의해 달성될 수 있다. 프로세스들은 비제한적으로 프로세스-간 통신을 위한 통상적을 기술을 포함하는 다양한 기술들을 사용하여 통신할 수 있으며, 프로세스들의 상이한 쌍들이 상이한 기술들을 사용할 수 있거나, 또는 프로세스들의 동일한 쌍이 상이한 시점에 상이한 기술들을 사용할 수 있다.

[0229] 따라서 본 명세서 및 도면은 제한적인 의미라기보다는 예시적인 의미로 간주되어야 한다. 그러나, 청구항들에 기술되는 바와 같은 광범위한 사상 및 범위로부터 벗어나지 않고 이에 대한 추가들, 대체들, 삭제들, 및 다른 수정들 및 변화들이 이루어질 수 있다는 것이 명백할 것이다. 따라서, 특정 개시 실시예들이 설명되었지만, 이들은 제한적인 것으로 의도되지 않는다. 다양한 수정예들 및 등가물들이 다음의 청구항들의 범위 내에 속한다.

[0230] 개시된 실시예들을 설명하는 맥락에서(특히 다음의 청구항들의 맥락에서) 용어들 "일(a 및 an)"과 "상기(the)" 및 유사한 지시대상들의 사용은, 본원에서 달리 표시되거나 또는 문맥에 의해 명백히 모순되지 않는 한 단수형 및 복수형 둘 모두를 커버하는 것으로 해석되어야 한다. 용어들 "포함하는", "갖는", "구성되는", 및 "함유하는"은, 달리 언급되지 않는 한 개방적인 용어(즉, "포함하지만 이에 한정되지 않는")으로 해석되어야 한다. 용어 "연결되는"은, 어떤 개재물이 있는 경우에도, 부분적으로 또는 전체적으로 어떤 것 내에 포함되거나, 또는 이에 부착되거나, 또는 이와 함께 결합되는 것으로 해석되어야 한다. 본 명세서에서 값들의 범위를 언급하는 것은, 본 명세서에서 달리 표시되지 않는 한, 단지 범위 내에 속하는 각각의 개별 값을 개별적으로 참조하는 속기 방법으로서 역할하도록 의도되며, 각각의 개별적인 값은 마치 이것이 본원에서 개별적으로 언급되는 것과 같이 명세서에 통합된다. 본 명세서에서 설명된 모든 방법은, 본 명세서에서 달리 표시되지 않거나 문맥상 명백히 모순되지 않는 한 임의의 적절한 순서로 수행될 수 있다. 본원에서 제공되는 임의의 및 모든 예들 또는 예시적인 언어(예를 들어, "~와 같은")의 사용은 단지 실시예들을 더 양호하게 조명하도록 의도되며, 달리 청구되지 않는 한, 본 개시의 범위에 제한을 두지 않는다. 본 명세서의 어떤 언어도 본 개시의 실시에 필수적인 임의의 청구되지 않은 요소를 나타내는 것으로 해석되지 않아야 한다.

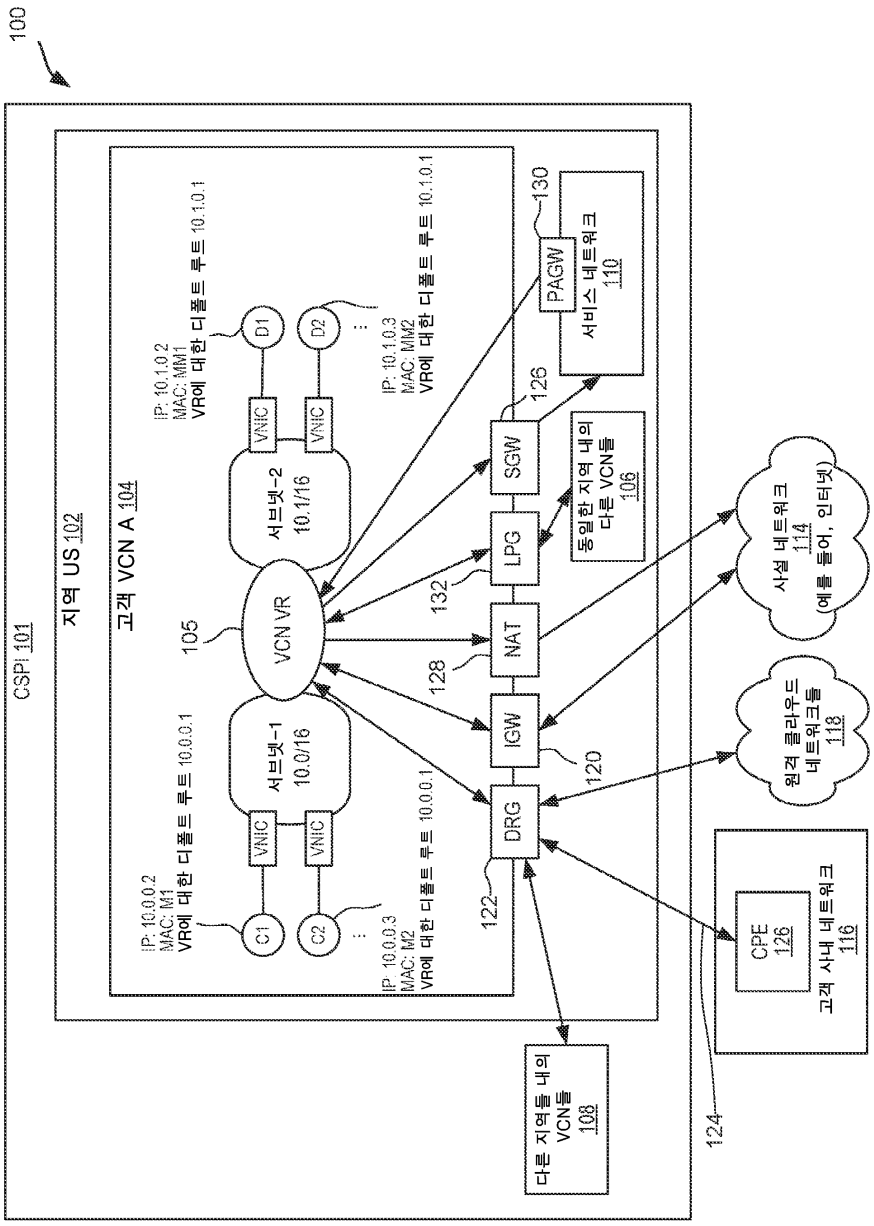
[0231] 구절 "X, Y, 또는 Z 중 적어도 하나"와 같은 이접적 언어는, 특별히 달리 명시되지 않는 한, 항목, 용어 등이 X, Y, 또는 Z 중 하나, 또는 이들의 조합(예를 들어, X, Y, 및/또는 Z)일 수 있음을 나타내기 위해 일반적으로 사용되는 문맥 내에서 이해되도록 의도된다. 따라서, 그러한 이접적 언어는 일반적으로, 특정 실시예가 각각 존재하기 위해 X 중 적어도 하나, Y 중 적어도 하나, 또는 Z 중 적어도 하나가 필요하다는 것을 의미하도록 의도되지 않는다.

[0232] 본 개시내용을 수행하기 위해 알려진 최적 모드를 포함하여 본 개시내용의 선호되는 실시예들이 본 명세서에서 설명되었다. 이러한 선호되는 실시예들의 변형들은 이상의 설명을 숙독할 때 당업자들에게 명백해질 것이다. 당업자들은 적절한 바와 같이 이러한 변형들을 이용할 수 있어야 하며, 본 개시내용은 본 명세서에서 구체적으로 설명된 것과 달리 실시될 수 있다. 따라서, 본 개시내용은 적용가능한 법률에 의해 허용되는 바와 같이 본 명세서에 첨부된 청구항들에 언급된 주제의 모든 수정예들 및 등가물들을 포함한다. 또한, 이의 모든 가능한 변형예들에서 이상에서 설명된 요소들의 임의의 조합은 따라서 본원에서 달리 표시되지 않는 한 본 개시내용에 의해 포괄된다.

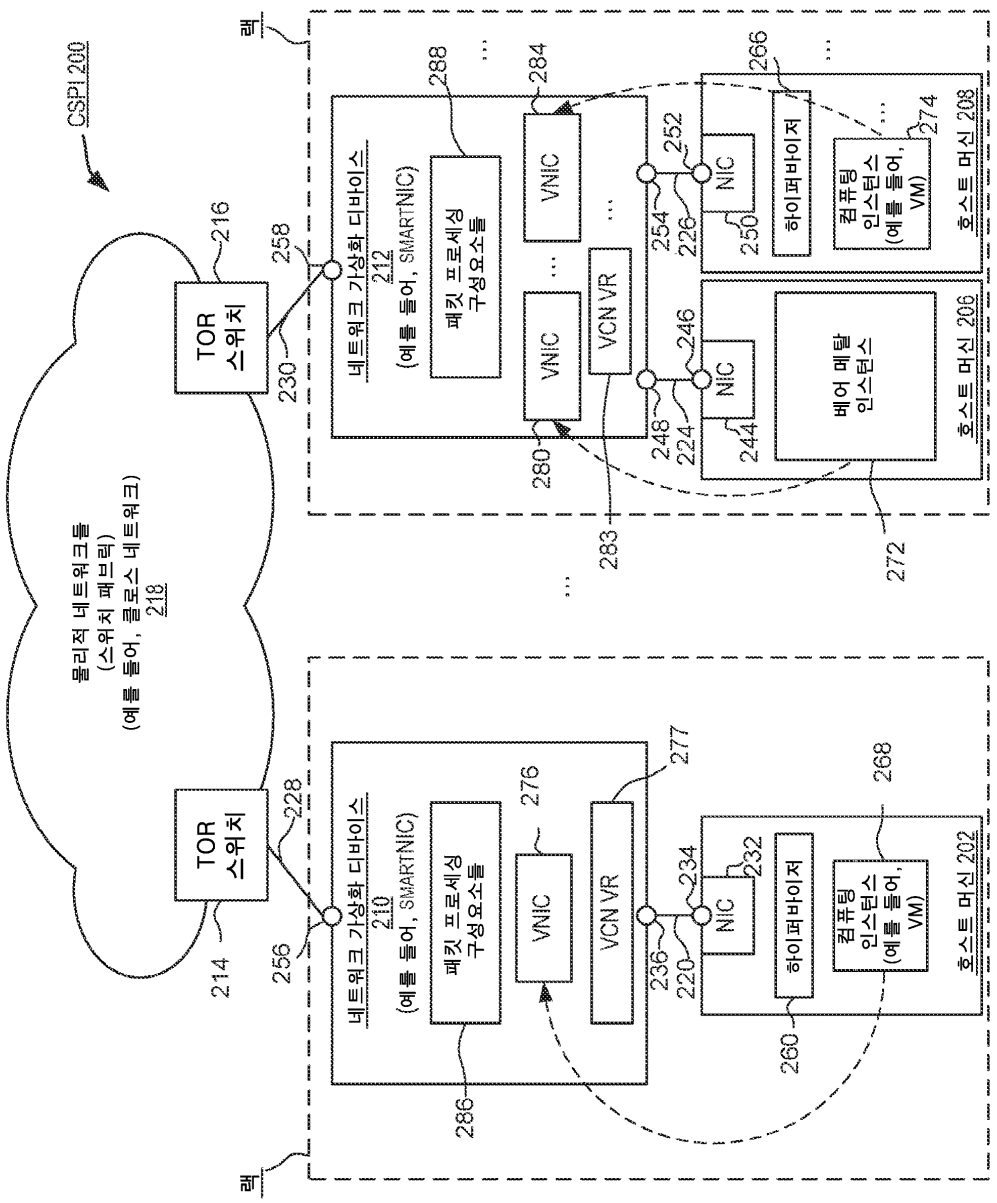
[0233] 본 명세서에서 인용된 간행물들, 특허 출원들 및 특허들을 포함하는 모든 참조문헌들은, 이로써 각각의 참조문헌이 참조로서 포함되도록 개별적으로 그리고 구체적으로 표시되고 그 전체가 본 명세서에서 기술된 것과 동일한 정도로 참조로서 본원에 포함된다. 이상의 명세서에서, 본 개시내용의 측면들이 본 개시의 특정 실시예들을 참조하여 설명되었지만, 당업자들은 본 개시내용이 이에 한정되지 않는다는 것을 인식할 것이다. 이상에서 설명된 개시내용의 다양한 특징들 및 측면들이 개별적으로 또는 함께 사용될 수 있다. 추가로, 실시예들은 본 명세서의 광범위한 사상 및 범위로부터 벗어나지 않고 본 명세서에서 설명된 것들을 넘어 임의의 수의 환경들 및 애플리케이션들에서 사용될 수 있다. 이에 따라, 본 명세서 및 도면들은 제한적인 것이 아니라 예시적인 것으로 간주되어야 한다.

도면

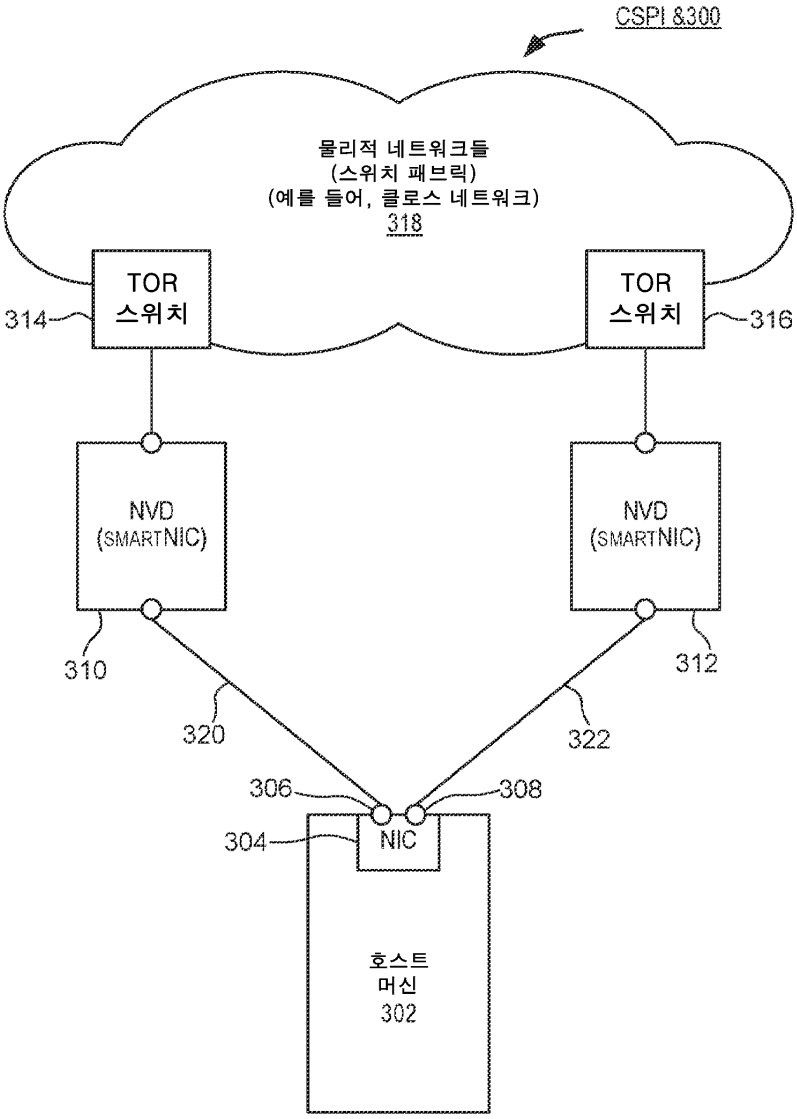
도면1


100
CSP! 101
지역 US 102
고객 VCN A 104
105
VCN VR
서브넷-1
10.0/16
서브넷-2
10.1/16
VNIC
C1
C2
D1
D2
IP: 10.0.0.2
MAC: M1
VR에 대한 디폴트 루트 10.0.0.1
IP: 10.0.0.3
MAC: M2
VR에 대한 디폴트 루트 10.0.0.1
IP: 10.1.0.2
MAC: MM1
VR에 대한 디폴트 루트 10.1.0.1
IP: 10.1.0.3
MAC: MM2
VR에 대한 디폴트 루트 10.1.0.1
DRG
122
IGW
120
NAT
128
LPG
132
SGW
126
PAGW
130
서비스 네트워크
110
동일한 지역 내의
다른 VCN들
106
사설 네트워크
114
(예를 들어, 인터넷)
원격 클라우드
네트워크들
118
CPE
126
고객 사내 네트워크
116
124
다른 지역들 내의
VCN들
108

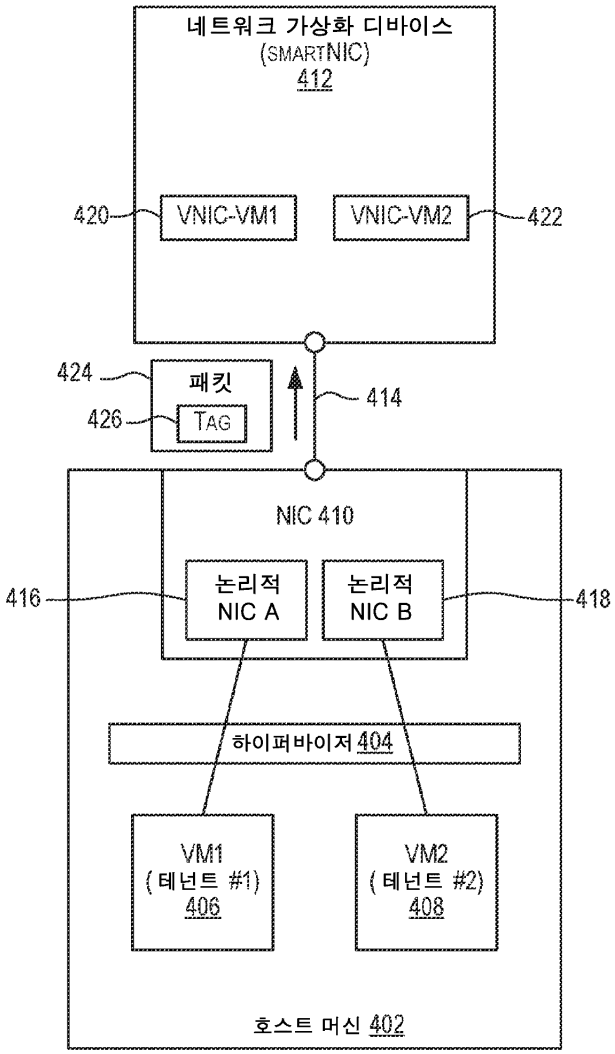
도면2


CSP1 200
물리적 네트워크들
(스위치 패브릭)
(예를 들어, 클로스 네트워크)
218
TOR 스위치
216
214
TOR 스위치
228
230
258
256
랙
랙
네트워크 가상화 디바이스
212
(예를 들어, SMARTNIC)
네트워크 가상화 디바이스
210
(예를 들어, SMARTNIC)
패킷 프로세싱 구성요소들
288
패킷 프로세싱 구성요소들
286
VNIC
284
VNIC
280
VNIC
276
VCN VR
VCN VR
277
NIC
252
NIC
246
NIC
244
NIC
234
NIC
232
하이퍼바이저
266
하이퍼바이저
260
베어 메탈 인스턴스
272
컴퓨팅 인스턴스 (예를 들어, VM)
274
컴퓨팅 인스턴스 (예를 들어, VM)
268
254
226
248
224
236
220
283
호스트 머신 208
호스트 머신 206
호스트 머신 202

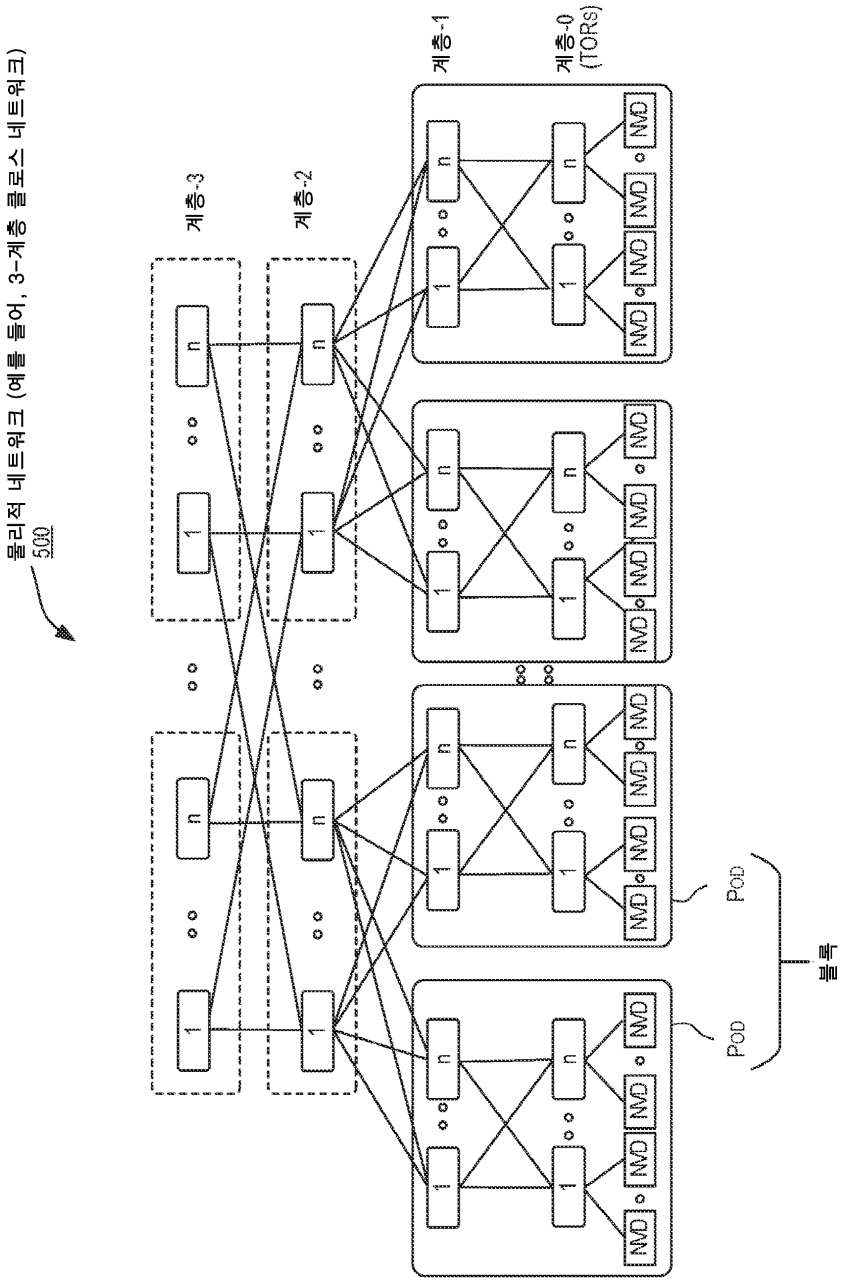
도면3


CSPI &300
물리적 네트워크들
(스위치 패브릭)
(예를 들어, 클로스 네트워크)
318
TOR
스위치
314
TOR
스위치
316
NVD
(SMARTNIC)
NVD
(SMARTNIC)
310
312
320
322
306
308
304
NIC
호스트
머신
302

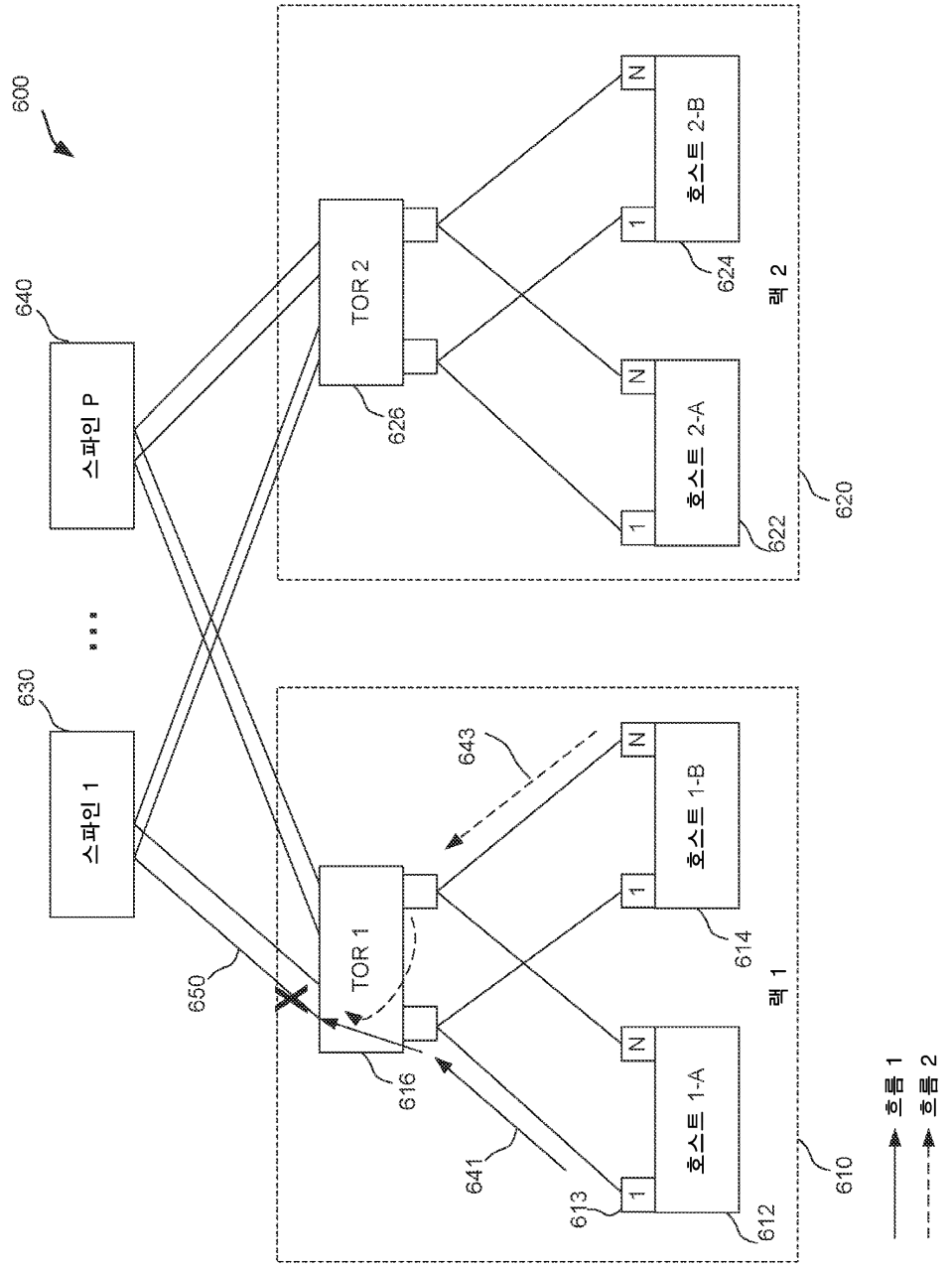
도면4


네트워크 가상화 디바이스
(SMARTNIC)
412
420
VNIC-VM1
VNIC-VM2
422
424
패킷
426
TAG
414
NIC 410
416
논리적
NIC A
논리적
NIC B
418
하이퍼바이저 404
VM1
(테넌트 #1)
406
VM2
(테넌트 #2)
408
호스트 머신 402

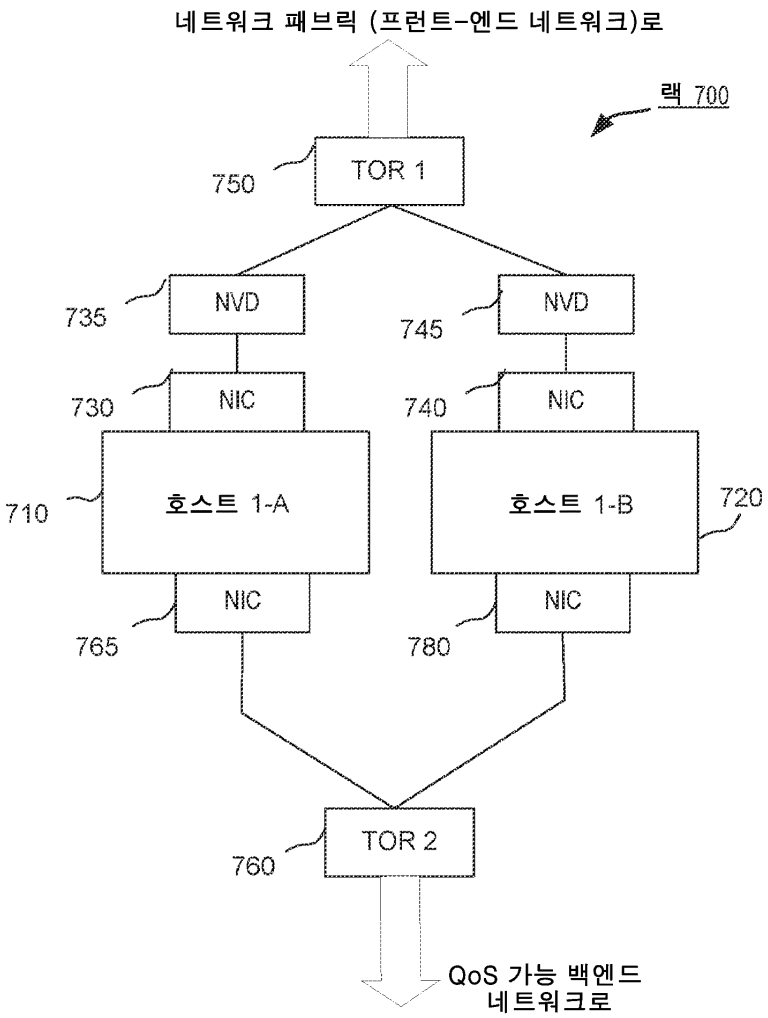
도면5


물리적 네트워크 (예를 들어, 3-계층 클로스 네트워크)
500
계층-3
계층-2
계층-1
계층-0
(TORs)
n
1
NMD
POD
블록

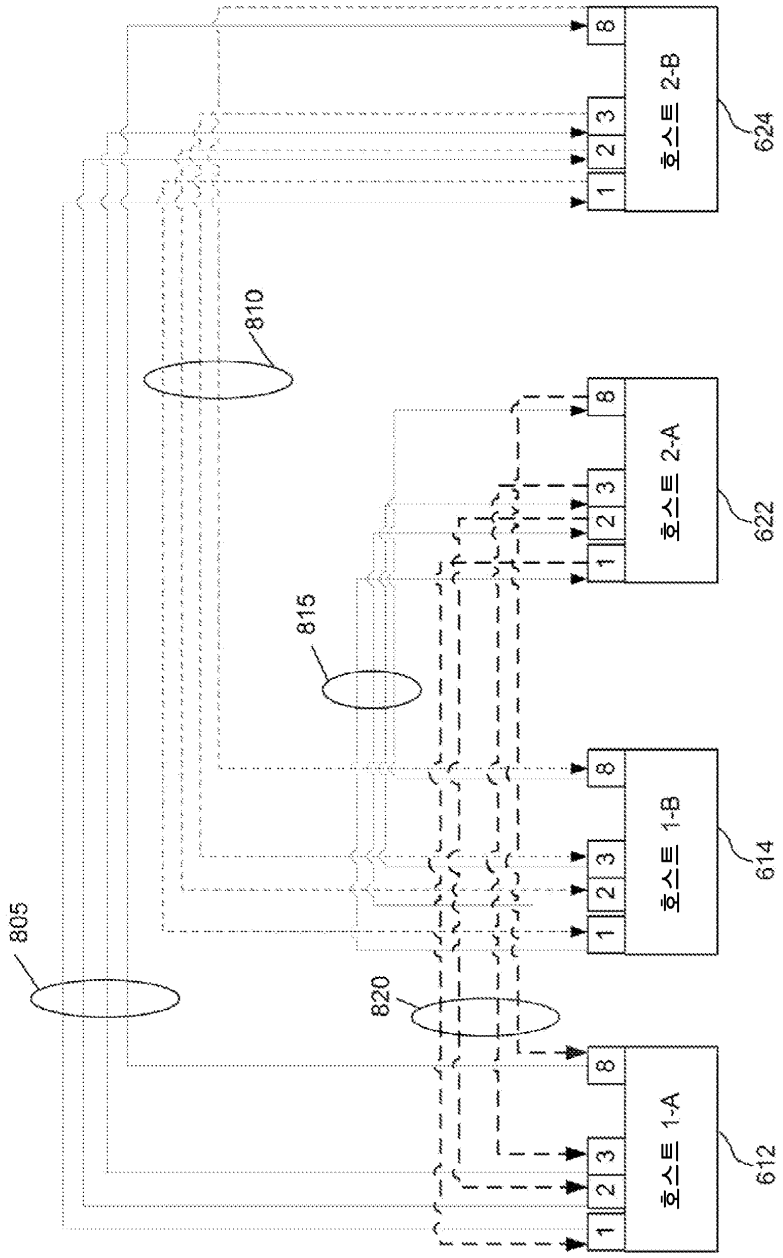
도면6


600
640
스파인 P
630
스파인 1
650
TOR 2
626
TOR 1
616
643
641
613
호스트 2-B
624
호스트 2-A
622
620
랙 2
호스트 1-B
614
호스트 1-A
612
610
랙 1
N
1
흐름 1
흐름 2

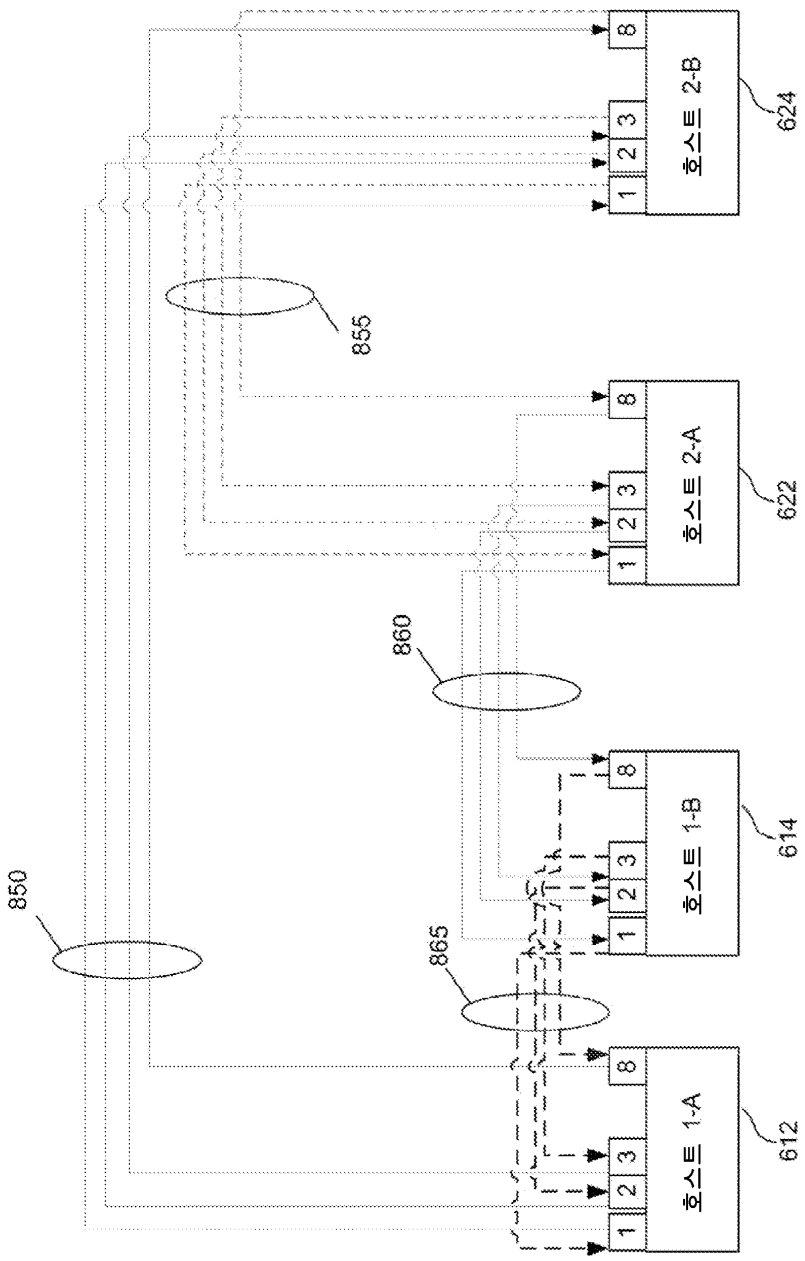
도면7


네트워크 패브릭 (프런트-엔드 네트워크)로
랙 700
750
TOR 1
735
NVD
745
NVD
730
NIC
740
NIC
710
호스트 1-A
호스트 1-B
720
NIC
NIC
765
780
760
TOR 2
QoS 가능 백엔드
네트워크로

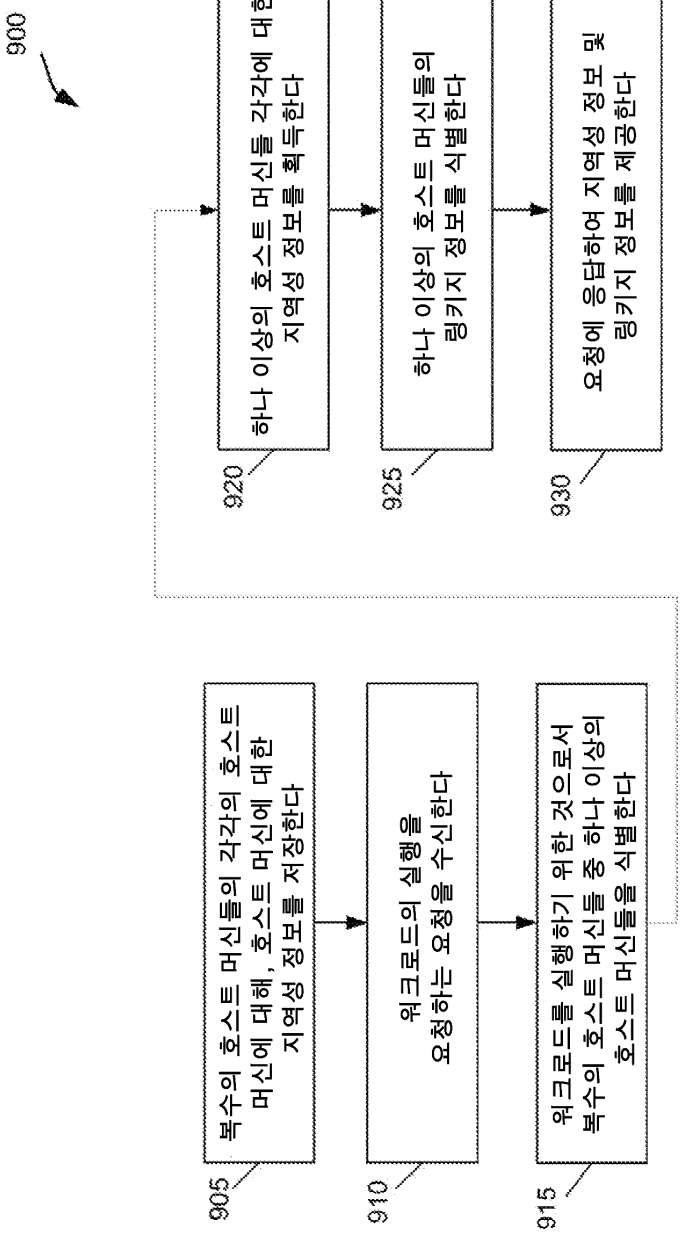
도면8a



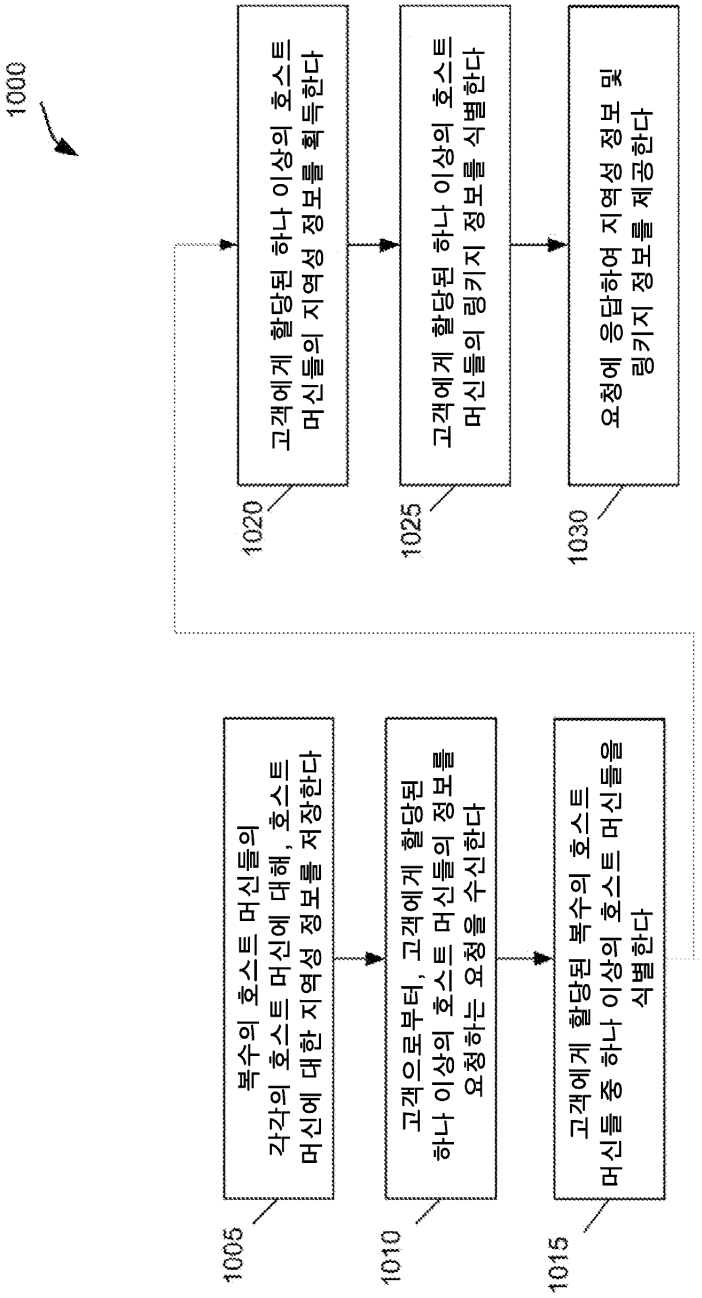
도면8b



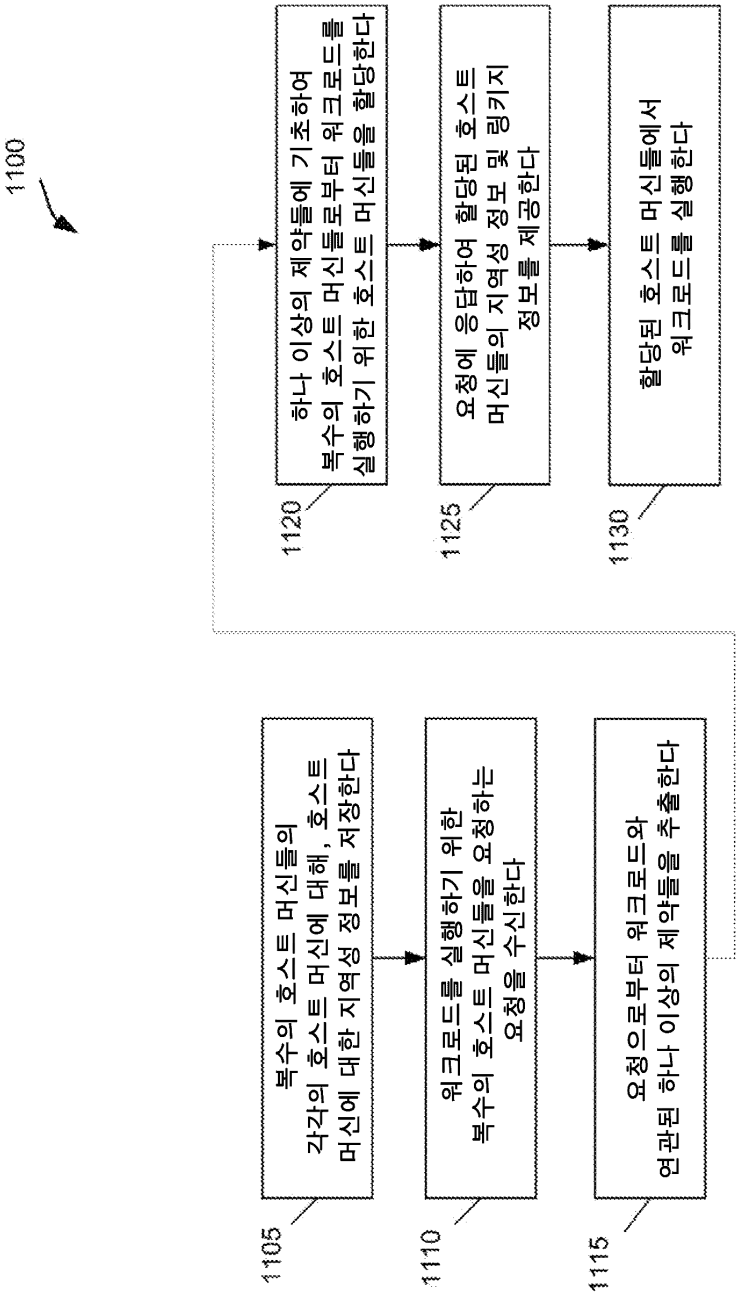
도면9


900
905
복수의 호스트 머신들의 각각의 호스트 머신에 대해, 호스트 머신에 대한 지역성 정보를 저장한다
910
워크로드의 실행을 요청하는 요청을 수신한다
915
워크로드를 실행하기 위한 것으로서 복수의 호스트 머신들 중 하나 이상의 호스트 머신들을 식별한다
920
하나 이상의 호스트 머신들 각각에 대한 지역성 정보를 획득한다
925
하나 이상의 호스트 머신들의 링키지 정보를 식별한다
930
요청에 응답하여 지역성 정보 및 링키지 정보를 제공한다

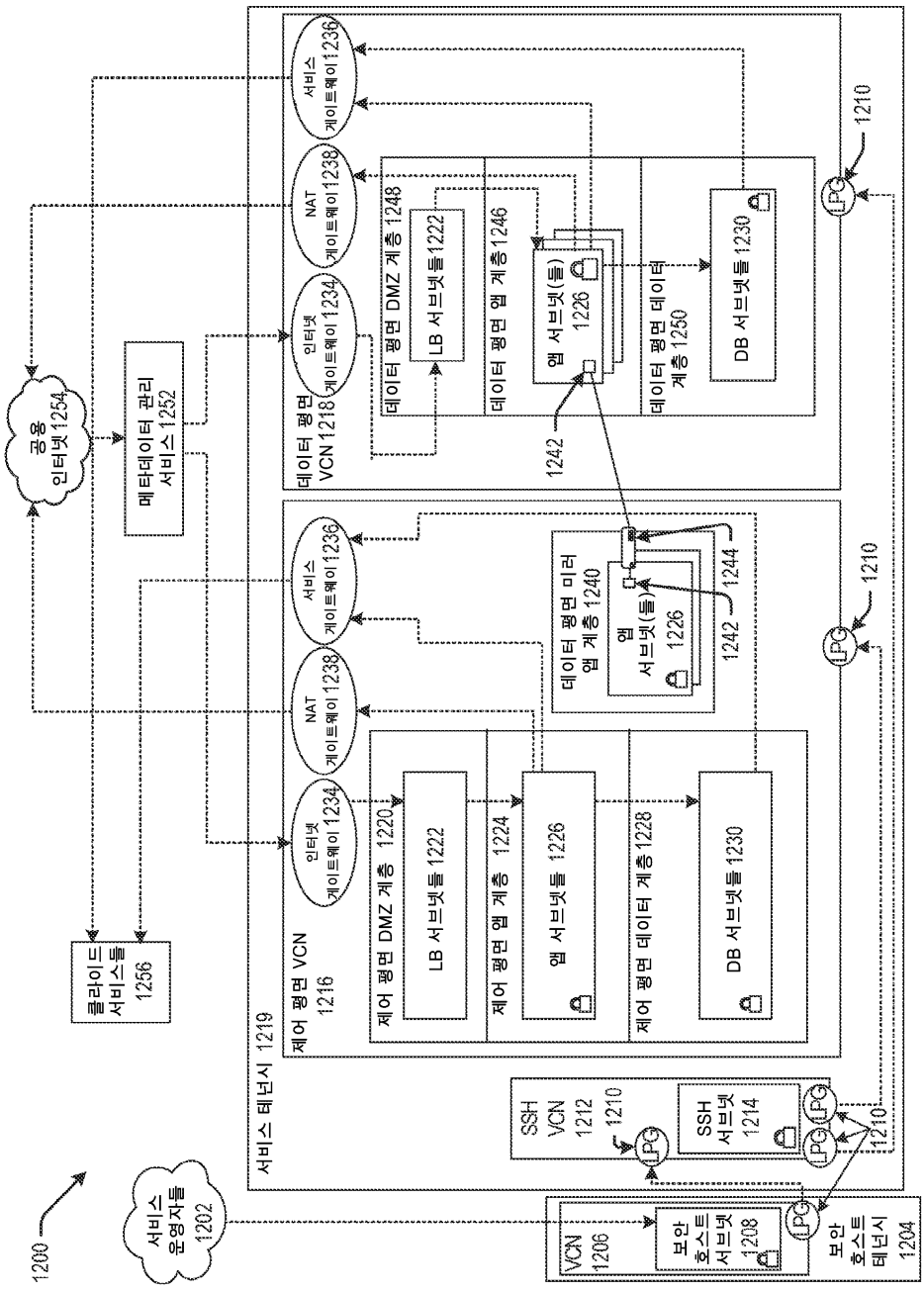
도면10


1000
복수의 호스트 머신들의 각각의 호스트 머신에 대해, 호스트 머신에 대한 지역성 정보를 저장한다
1005
고객으로부터, 고객에게 할당된 하나 이상의 호스트 머신들의 정보를 요청하는 요청을 수신한다
1010
고객에게 할당된 복수의 호스트 머신들 중 하나 이상의 호스트 머신들을 식별한다
1015
고객에게 할당된 하나 이상의 호스트 머신들의 지역성 정보를 획득한다
1020
고객에게 할당된 하나 이상의 호스트 머신들의 링키지 정보를 식별한다
1025
요청에 응답하여 지역성 정보 및 링키지 정보를 제공한다
1030

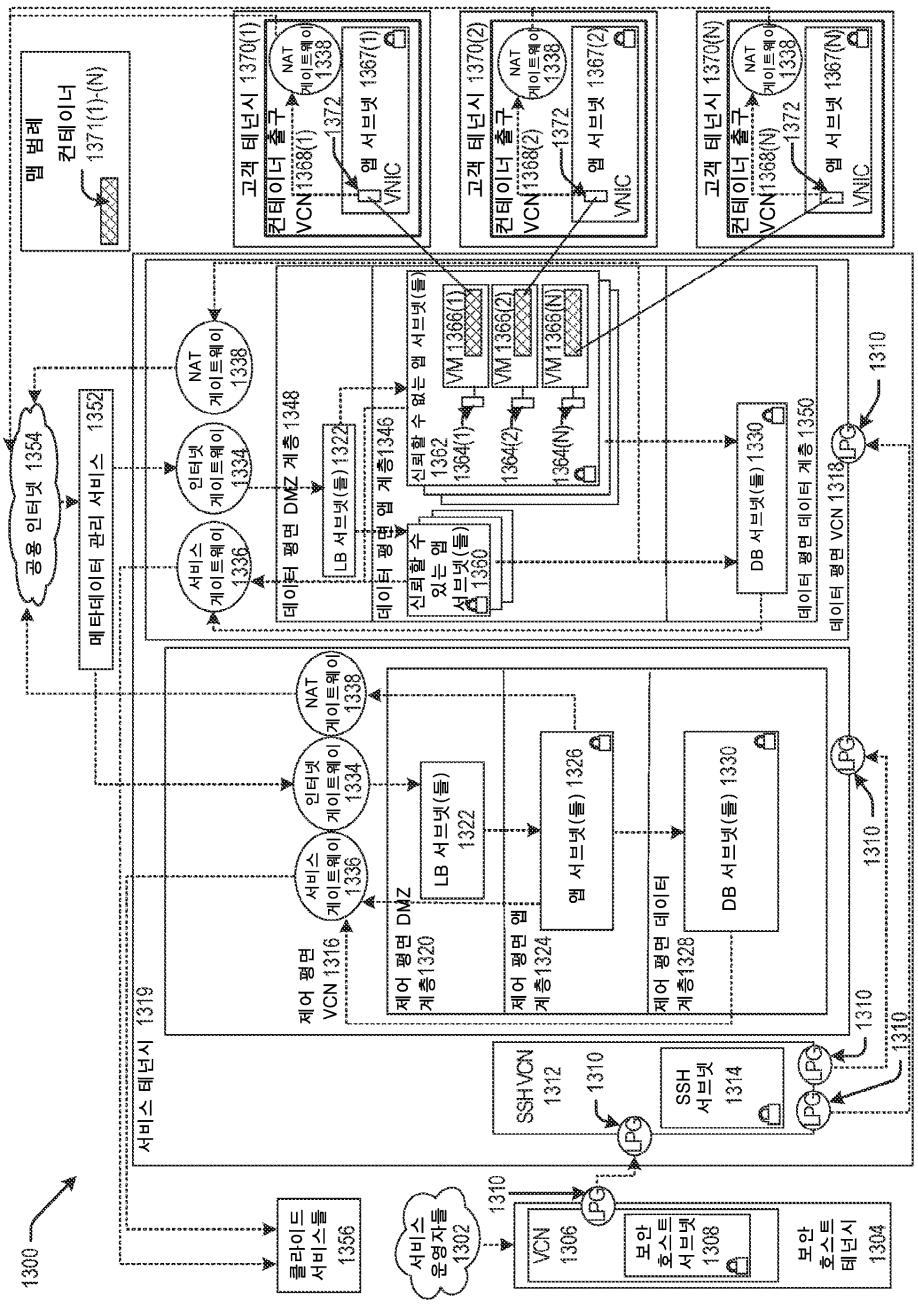
도면11


1100
1105
복수의 호스트 머신들의 각각의 호스트 머신에 대해, 호스트 머신에 대한 지역성 정보를 저장한다
1110
워크로드를 실행하기 위한 복수의 호스트 머신들을 요청하는 요청을 수신한다
1115
요청으로부터 워크로드와 연관된 하나 이상의 제약들을 추출한다
1120
하나 이상의 제약들에 기초하여 복수의 호스트 머신들로부터 워크로드를 실행하기 위한 호스트 머신들을 할당한다
1125
요청에 응답하여 할당된 호스트 머신들의 지역성 정보 및 링키지 정보를 제공한다
1130
할당된 호스트 머신들에서 워크로드를 실행한다

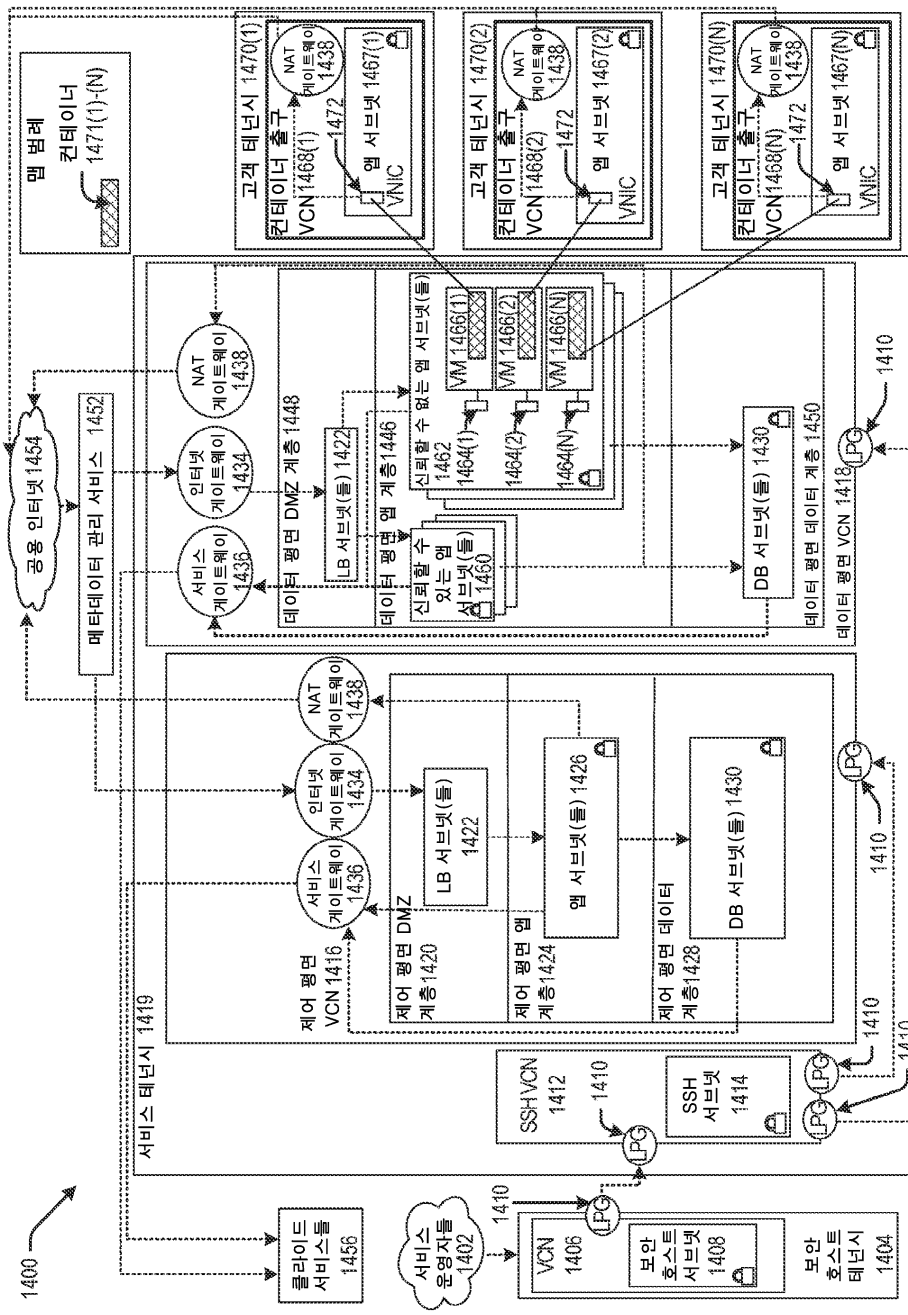
도면12


1200
서비스 운영자들 1202
서비스 테넌시 1219
클라이드 서비스들 1256
공용 인터넷 1254
메타데이터 관리 서비스 1252
제어 평면 VCN 1216
제어 평면 DMZ 계층 1220
LB 서브넷들 1222
제어 평면 앱 계층 1224
앱 서브넷들 1226
제어 평면 데이터 계층 1228
DB 서브넷들 1230
인터넷 게이트웨이 1234
NAT 게이트웨이 1238
서비스 게이트웨이 1236
데이터 평면 미러 앱 계층 1240
앱 서브넷(들) 1226
1242
1244
데이터 평면 VCN 1218
데이터 평면 DMZ 계층 1248
데이터 평면 앱 계층 1246
데이터 평면 데이터 계층 1250
SSH VCN 1212
SSH 서브넷 1214
VCN 1206
보안 호스트 서브넷 1208
보안 호스트 테넌시 1204
LPG
1210

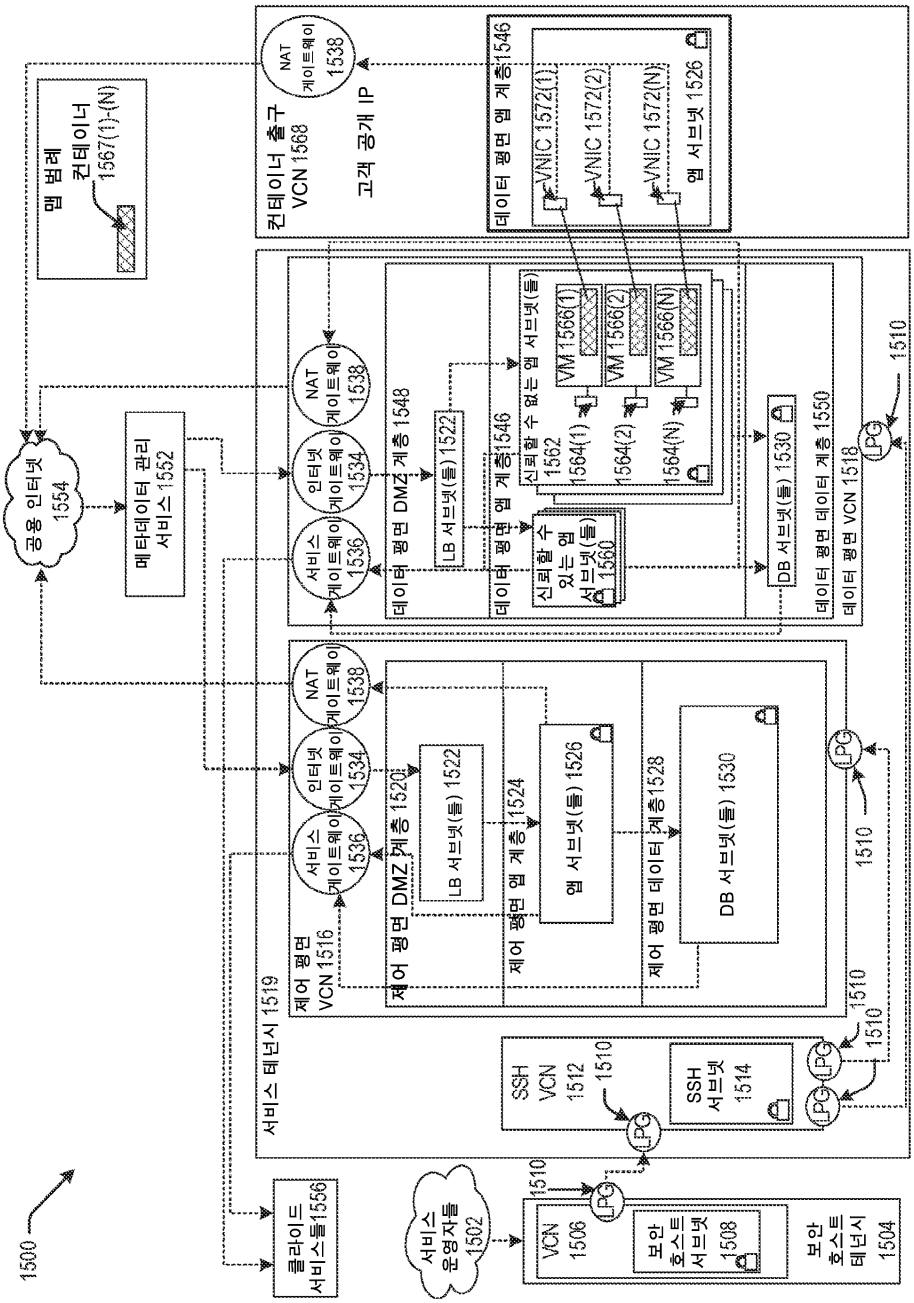
도면13


맵 범례
컨테이너
1371(1)-(N)
고객 테넌시 1370(1)
컨테이너 출구
VCN 1368(1)
1372
앱 서브넷 1367(1)
VNIC
NAT 게이트웨이 1338
고객 테넌시 1370(2)
컨테이너 출구
VCN 1368(2)
1372
앱 서브넷 1367(2)
VNIC
NAT 게이트웨이 1338
고객 테넌시 1370(N)
컨테이너 출구
VCN 1368(N)
1372
앱 서브넷 1367(N)
VNIC
NAT 게이트웨이 1338
공용 인터넷 1354
메타데이터 관리 서비스 1352
서비스 게이트웨이 1336
인터넷 게이트웨이 1334
NAT 게이트웨이 1338
데이터 평면 DMZ 계층 1348
LB 서브넷(들) 1322
데이터 평면 앱 계층 1346
신뢰할 수 없는 앱 서브넷(들) 1362
VM 1366(1)
VM 1366(2)
VM 1366(N)
1364(1)
1364(2)
1364(N)
신뢰할 수 있는 앱 서브넷(들) 1360
데이터 평면 데이터 계층 1350
DB 서브넷(들) 1330
데이터 평면 VCN 1318
1310
서비스 테넌시 1319
제어 평면 VCN 1316
제어 평면 DMZ 계층 1320
제어 평면 앱 계층 1324
제어 평면 데이터 계층 1328
LB 서브넷(들) 1322
앱 서브넷(들) 1326
DB 서브넷(들) 1330
SSH VCN 1312
SSH 서브넷 1314
PG
1300
클라이드 서비스들 1356
서비스 운영자들 1302
VCN 1306
보안 호스트 서브넷 1308
보안 호스트 테넌시 1304

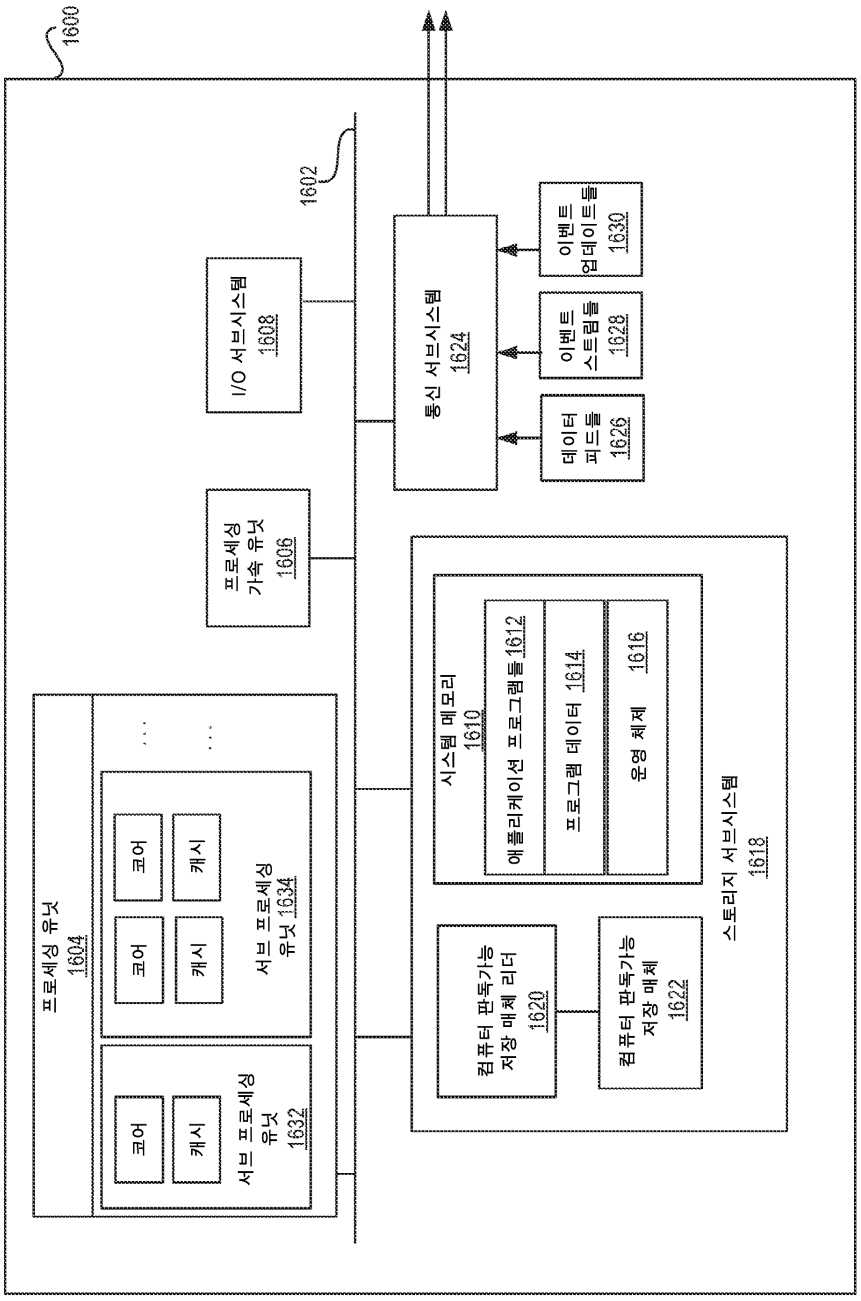
도면14


맵 범례
컨테이너
1471(1)-(N)
고객 테넌시 1470(1)
컨테이너 출구
VCN 1468(1)
1472
앱 서브넷 1467(1)
VNIC
NAT 게이트웨이 1438
고객 테넌시 1470(2)
컨테이너 출구
VCN 1468(2)
1472
앱 서브넷 1467(2)
VNIC
NAT 게이트웨이 1438
고객 테넌시 1470(N)
컨테이너 출구
VCN 1468(N)
1472
앱 서브넷 1467(N)
VNIC
NAT 게이트웨이 1438
공용 인터넷 1454
메타데이터 관리 서비스 1452
서비스 게이트웨이 1436
인터넷 게이트웨이 1434
NAT 게이트웨이 1438
데이터 평면 DMZ 계층 1448
LB 서브넷(들) 1422
데이터 평면 앱 계층 1446
신뢰할 수 없는 앱 서브넷(들) 1462
VM 1466(1) 1464(1)
VM 1466(2) 1464(2)
VM 1466(N) 1464(N)
신뢰할 수 있는 앱 서브넷(들) 1460
DB 서브넷(들) 1430
데이터 평면 데이터 계층 1450
데이터 평면 VCN 1418
1410
LPG
서비스 테넌시 1419
제어 평면 VCN 1416
제어 평면 DMZ 계층 1420
LB 서브넷(들) 1422
제어 평면 앱 계층 1424
앱 서브넷(들) 1426
제어 평면 데이터 계층 1428
DB 서브넷(들) 1430
SSH VCN 1412
SSH 서브넷 1414
1400
클라이언트 서비스들 1456
서비스 운영자들 1402
VCN 1406
보안 호스트 서브넷 1408
보안 호스트 테넌시 1404

도면15


1500
맵 범례
컨테이너
1567(1)-(N)
컨테이너 출구
VCN 1568
고객 공개 IP
NAT
게이트웨이
1538
데이터 평면 앱 계층1546
VNIC 1572(1)
VNIC 1572(2)
VNIC 1572(N)
앱 서브넷 1526
공용 인터넷
1554
메타데이터 관리
서비스 1552
인터넷
게이트웨이
1534
서비스
게이트웨이
1536
데이터 평면 DMZ 계층 1548
LB 서브넷(들) 1522
데이터 평면 앱 계층1546
신뢰할 수 없는 앱 서브넷(들)
VM 1566(1)
VM 1566(2)
VM 1566(N)
1562
1564(1)
1564(2)
1564(N)
신뢰할 수
있는 앱
서브넷(들)
1560
DB 서브넷(들) 1530
데이터 평면 데이터 계층 1550
데이터 평면 VCN 1518
PG
1510
서비스 테넌시 1519
제어 평면
VCN 1516
제어 평면 DMZ 계층 1520
제어 평면 앱 계층 1524
제어 평면 데이터 계층1528
SSH
VCN
1512
SSH
서브넷
1514
클라이언트
서비스들1556
서비스
운영자들
1502
VCN
1506
보안
호스트
서브넷
1508
보안
호스트
테넌시
1504

도면16


1600
1602
I/O 서브시스템
1608
프로세싱
가속 유닛
1606
통신 서브시스템
1624
이벤트
업데이트들
1630
이벤트
스트림들
1628
데이터
피드들
1626
프로세싱 유닛
1604
서브 프로세싱
유닛 1634
코어
캐시
서브 프로세싱
유닛
1632
시스템 메모리
1610
애플리케이션 프로그램들 1612
프로그램 데이터 1614
운영 체제 1616
컴퓨터 판독가능
저장 매체 리더
1620
컴퓨터 판독가능
저장 매체
1622
스토리지 서브시스템
1618