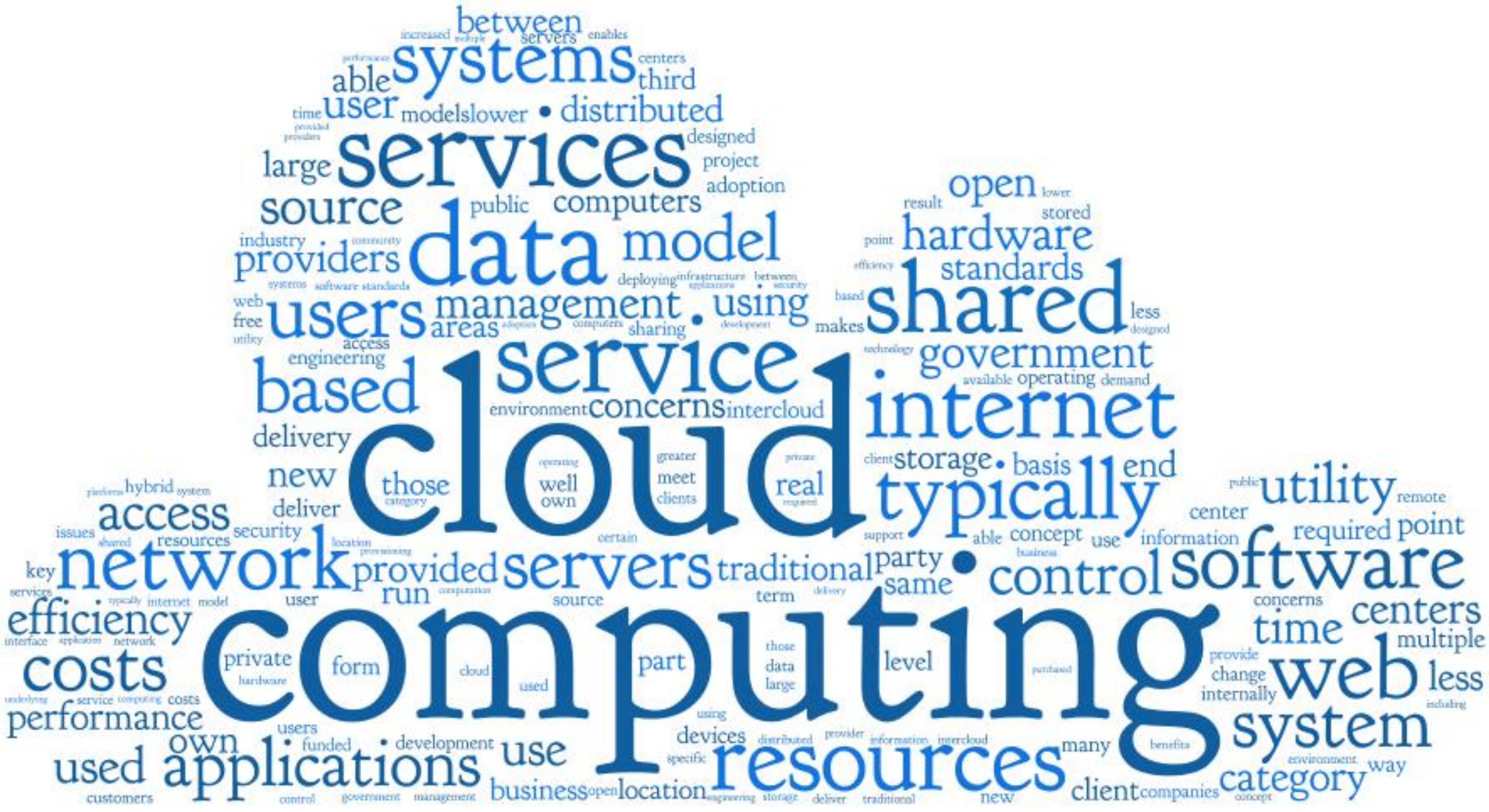
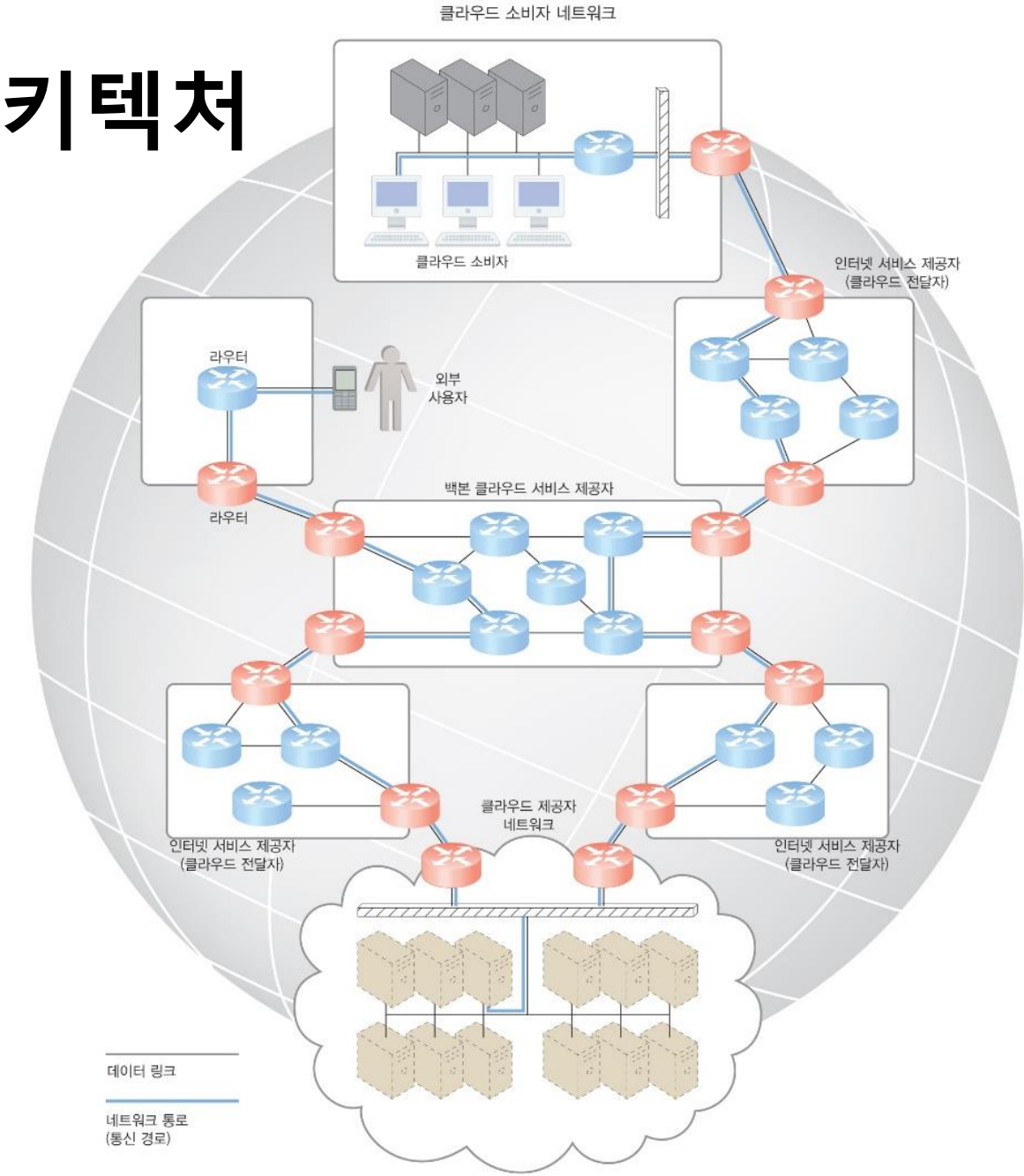

cloud
computing
services
data
service
systems
shared
internet
typically
network
software
resources
applications
users
servers
management
model
based
access
costs
efficiency
web
system
utility
control
provided
traditional
hardware
government
between
source
providers
distributed
performance
used
use
storage
centers
time
open
standards
using
delivery
new
those
private
form
part
level
category
client
location
business
own
development
real
well
meet
concerns
intercloud
environment
computers
public
large
user
able
third
designed
project
adoption
engineering
areas
sharing
party
same
run
key
security
resources
deliver
hybrid
issues
basis
end
concept
information
required
point
center
multiple
less
change
internally
many
devices
funded
customers
companies
way
remote

02
클라우드 컴퓨팅 기술

02 Cloud Computing Technology

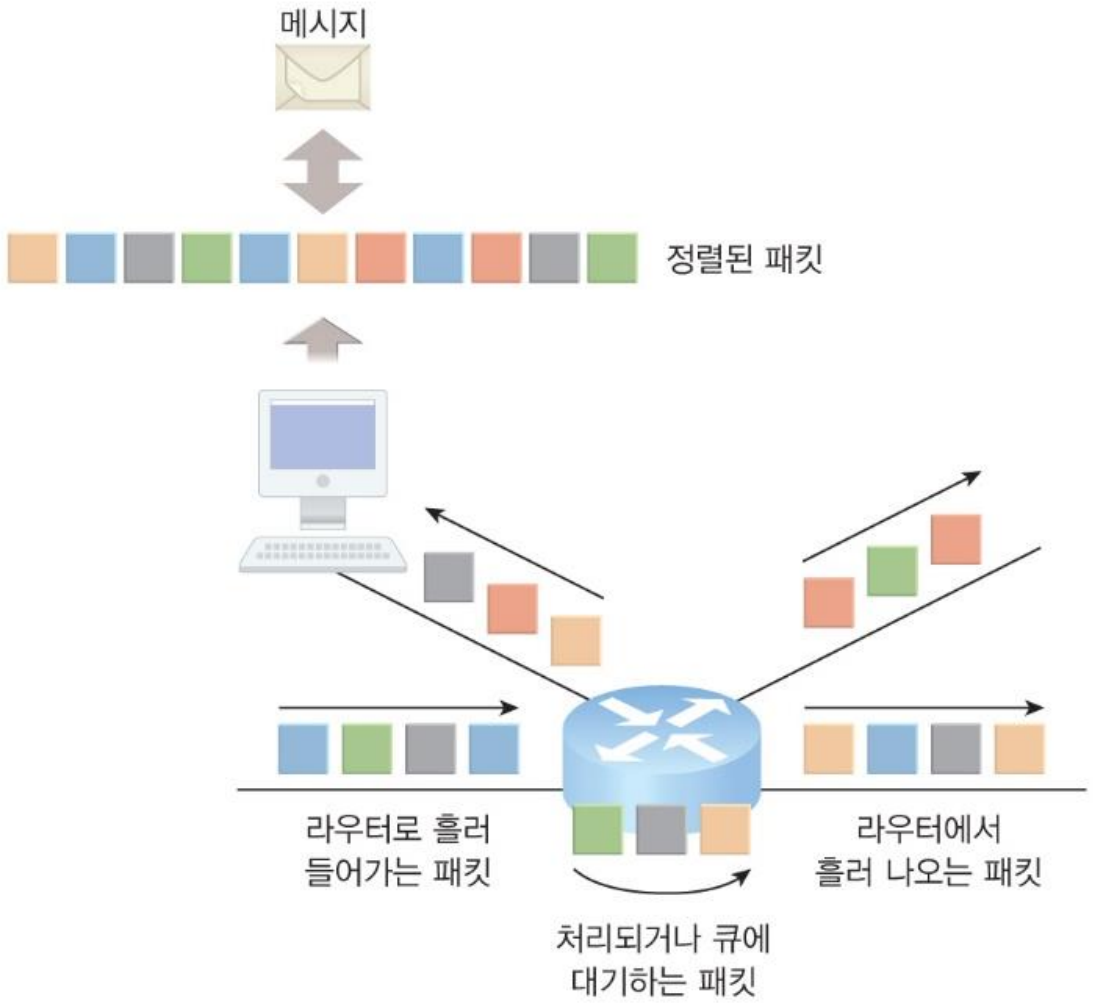
광대역 네트워크와 인터넷 아키텍처

- 모든 클라우드는 네트워크에 연결되어야 함
- 인터넷은 IT 자원의 원격 제공을 가능하게 하고 어디서나 네트워크에 접속할 수 있게 직접적으로 지원
- 대부분의 클라우드는 인터넷으로 이용 가능하지만 클라우드 소비자는 프라이빗 네트워크를 사용할지 LAN 전용선을 이용해 클라우드를 접속할지 선택 가능
- 클라우드 플랫폼의 잠재력은 인터넷 연결성과 서비스 품질의 발전과 함께 성장

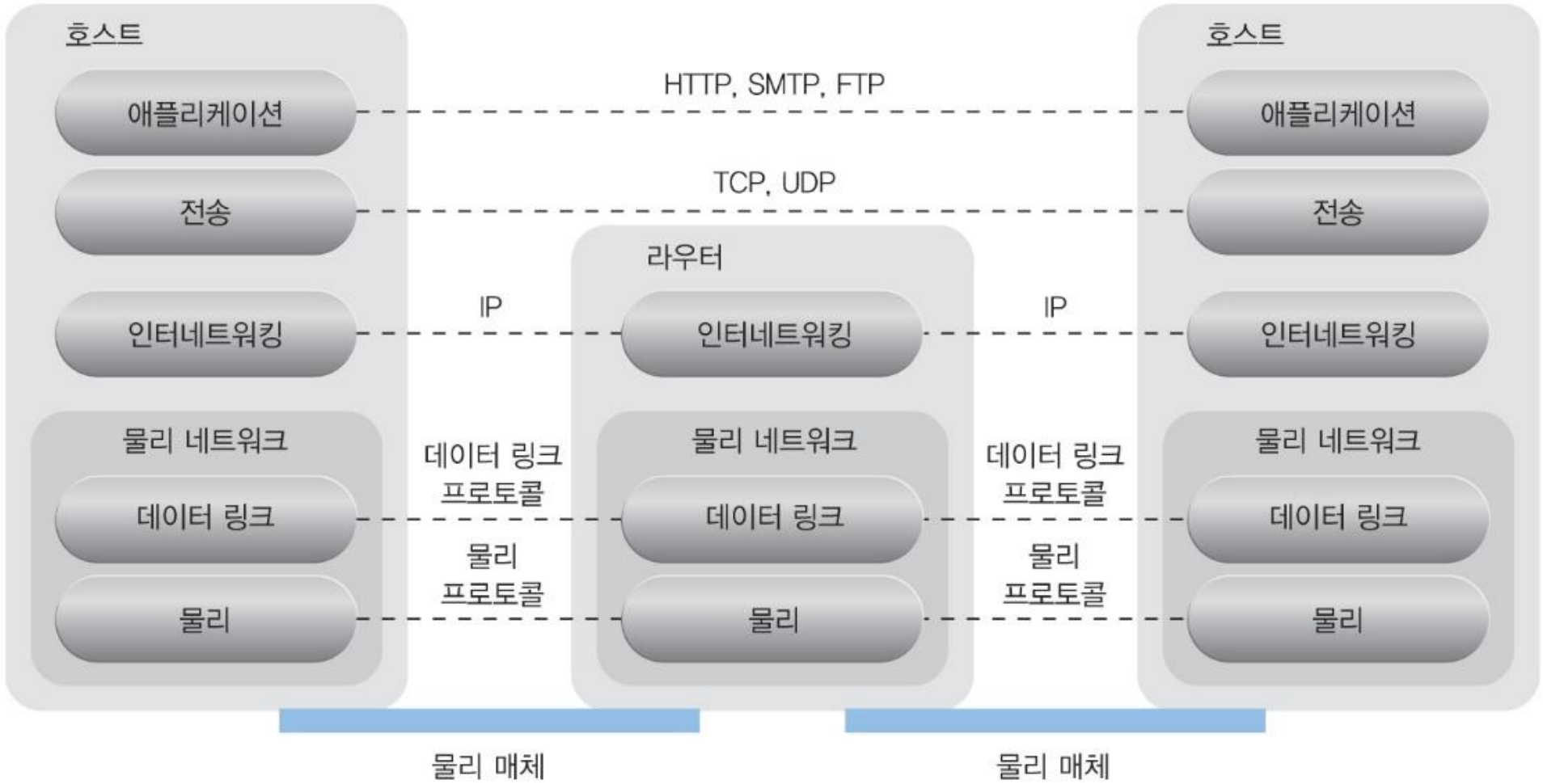


비연결형 패킷 교환

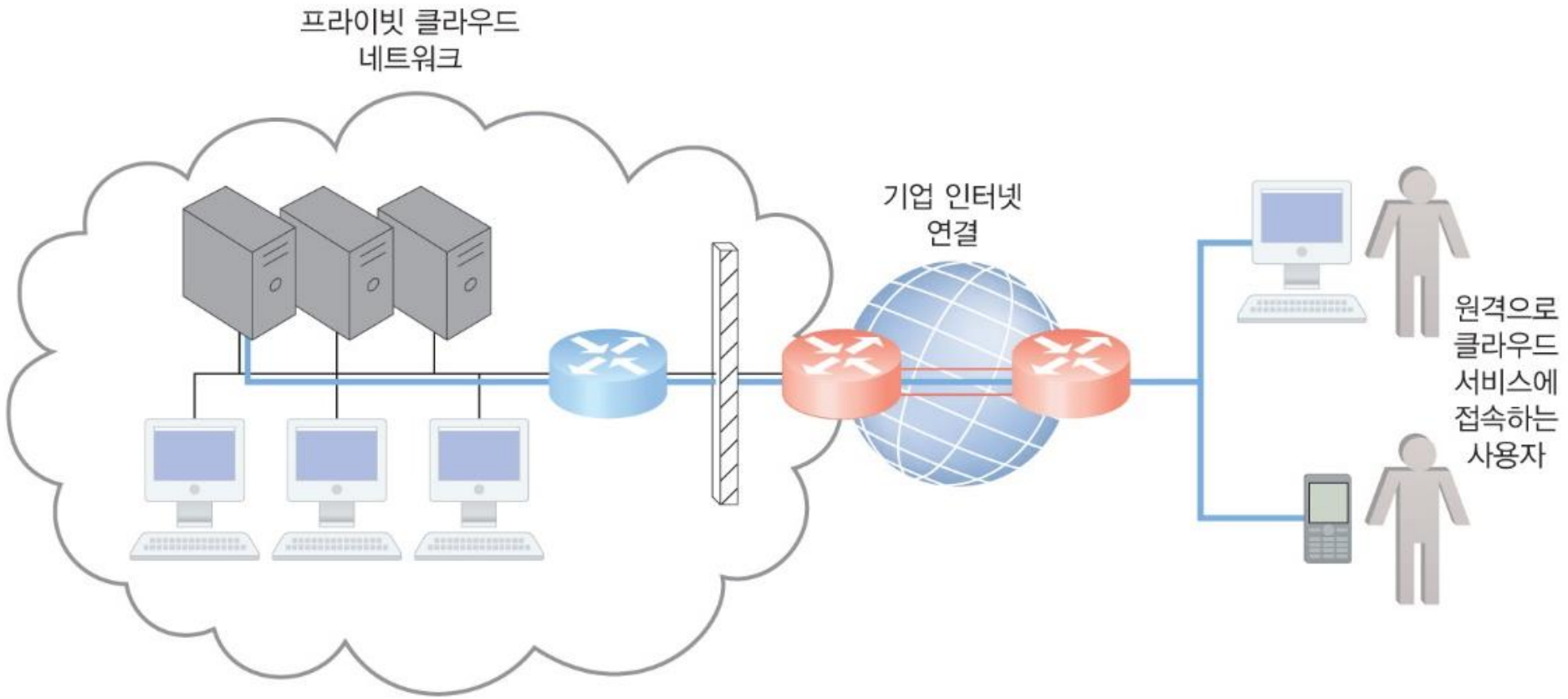
- 종단간(송신-수신 쌍) 데이터 흐름은 네트워크 스위치와 라우터를 통해 수신, 처리되는 제한된 크기의 패킷으로 나뉨
- 패킷은 큐에 대기한 후 중간 노드에서 다음 노드로 전송
- 각 패킷은 인터넷 프로토콜IP, Internet Protocol이나 맥MAC, Media Access Control 주소와 같은 필수적인 위치 정보를 가지며 모든 시작, 중간, 목적 노드에서 처리되고 전송



라우터 기반 상호 접속

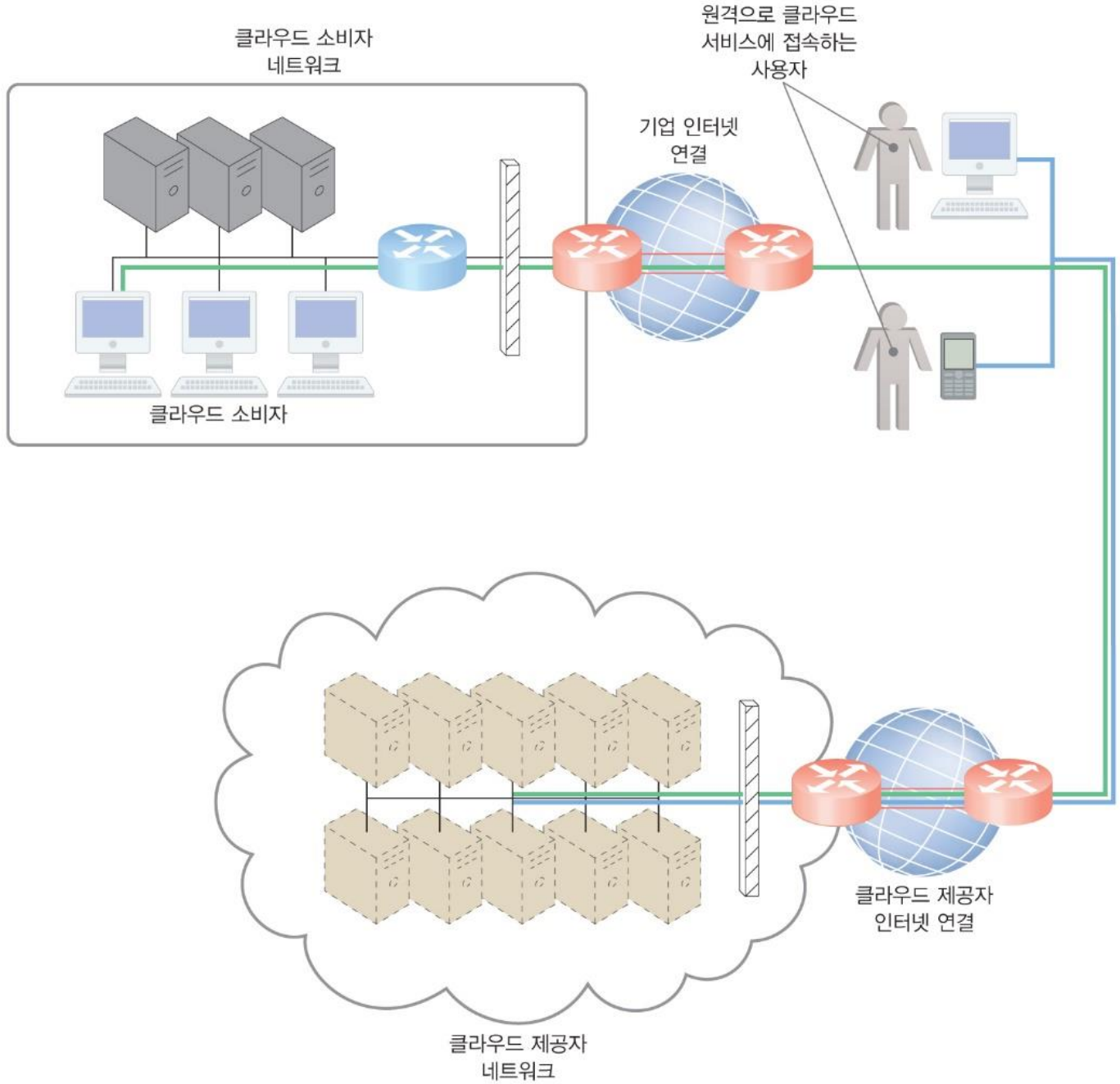

호스트
애플리케이션
전송
인터네트워킹
물리 네트워크
데이터 링크
물리
HTTP, SMTP, FTP
TCP, UDP
IP
데이터 링크
프로토콜
물리
프로토콜
라우터
인터네트워킹
물리 네트워크
데이터 링크
물리
IP
데이터 링크
프로토콜
물리
프로토콜
호스트
애플리케이션
전송
인터네트워킹
물리 네트워크
데이터 링크
물리
물리 매체
물리 매체

접속 문제



접속 문제

사내 IT 자원	내부 사용자 장치가 기업 네트워크를 통해 기업 IT 서비스에 접근
	내부 사용자가 외부 네트워크에서 로밍하면서 기업 네트워크를 통해 기업 IT 서비스에 접근
	외부 사용자가 기업 인터넷 연결을 통해 기업 IT 서비스에 접근
클라우드 기반 IT 자원	내부 사용자 장치가 인터넷 연결을 통해 기업 IT 서비스에 접근
	내부 사용자가 클라우드 제공자의 인터넷 연결을 통해 외부 네트워크에서 로밍하면서 기업 IT 서비스에 접근
	외부 사용자가 클라우드 제공자의 인터넷 연결을 통해 기업 IT 서비스에 접근



네트워크 대역폭과 대기 시간 문제

- 종단간 대역폭은 네트워크를 ISP에 연결하는 데이터 링크의 대역폭에 의해 영향을 받을 뿐 아니라 중간 노드를 연결하는 공유 데이터 링크의 전송 용량에도 영향을 받음
- ISP는 종단간 연결성을 보장하기 위해 필요한 중심 네트워크를 위해 광대역 네트워크 기술을 사용
- 대기 시간은 패킷이 한 데이터 노드에서 다른 노드로 전송되는데 걸리는 시간을 의미하며, 데이터 패킷 경로의 각 중간 노드를 지날 때마다 증가
- 서비스 품질QoS, Quality of Service을 갖는 패킷 네트워크로 트래픽의 우선순위가 정해져 있지 않을 때, 데이터 플로우는 대역폭 감소나 대기 시간 증가, 패킷 손실의 형태로 서비스 수준 저하 발생
- 대역폭은 많은 데이터가 전송되는 애플리케이션에 중요하고,
  대기 시간은 빠른 응답 시간을 요구사항으로 가진 애플리케이션에 중요

클라우드 전달자와 클라우드 제공자 선택

- 클라우드 소비자와 클라우드 제공자 간 인터넷 연결의 서비스 수준은 ISP에 의해서 결정
- ISP는 서로 다르고 경로 내에 여러 ISP 네트워크를 포함
- 여러 ISP를 포함한 QoS 관리는 사실상 매우 어려우며 종단간 서비스 수준이 요구사항을 충족시키기 위해서는 양측의 클라우드 전달자의 협조가 필요
- 클라우드 소비자와 클라우드 제공자는 클라우드 애플리케이션의 연결성과 신뢰성의 수준을 달성하기 위해서 추가 비용을 지불하여도 여러 클라우드 전달자를 사용할 필요가 있음

데이터 센터 기술 ①

- 가상화
  - 물리적 컴퓨팅 및 네트워킹 자원을 가상화된 컴포넌트로 추상화해 자원ㅇ르 좀더 손쉽게 할당, 운영, 해제, 모니터링, 통제할 수 있는 가상화 플랫폼 기반
  - 운영 및 관리 도구가 가상화 계층의 통제와 자원 추상화를 담당
- 표준화와 모듈화
  - 데이터 센터는 표준화된 하드웨어상에 구축되고 확장성, 성장, 빠른 하드웨어 교체를 지원하는 장비와 시설 인프라의 여러 독립적인 구성요소를 모은 모듈화된 아키텍처를 이용해 설계
  - 모듈화와 표준화는 조달, 획득, 배포, 운영, 유지보수에 대한 규모의 경제를 가능하게 하기 때문에 투자 및 운영 비용을 절감하는 핵심적인 요건
- 자동화
  - 데이터 센터는 감독 없이 프로비저닝, 설정, 패치, 모니터링과 같은 작업을 자동화하는 특화된 플랫폼을 가짐
  - 데이터 센터 관리와 플랫폼의 발전은 자체 설정과 자체 복구를 수행하는 자동화된 컴퓨팅 기술을 가능하게 함

데이터 센터 기술 ②

- 원격 운영과 관리
  - 데이터 센터의 IT 자원의 운영 및 관리 작업 대부분은 네트워크의 원격 콘솔과 관리 시스템을 통해 수행
  - 기술 인력들은 장비 관리나 배선 연결, 하드웨어 수준의 설치 및 유지보수와 같은 특별한 경우를 제외하고는 서버를 보유한 공간에 방문할 필요가 없음
- 높은 가용성
  - 데이터 센터의 서비스를 이용하는 조직은 정전이나 재난과 같이 비즈니스 연속성에 영향을 미치는 요소가 발생
  - 이러한 문제 요소를 고려하여 가용성을 유지하기 위한 높은 수준의 중복성을 가지도록 설계되고 운영
  - 데이터 센터는 시스템 실패에 대비하여 통신 링크와 클러스터링된 하드웨어의 로드 밸런싱과 함께 중복되고 방해받지 않는 전원 공급, 배선, 환경적 제어 서브 시스템을 갖추고 있음

데이터 센터 기술 ③

- 보안 인식 설계와 운영, 관리
  - 데이터 센터는 비즈니스 데이터를 저장하고 처리하는 중앙 집중 구조이기 때문에 물리적 및 논리적 접근 통제와 데이터 복구 전략과 같은 보안과 관련된 요구사항을 엄격하고 포괄적으로 적용
- 시설
  - 데이터 센터 시설은 특화된 컴퓨팅, 스토리지, 네트워크 장비에 맞춰 설계된 장소
  - 다양한 전원 공급, 배선 연결, 열, 통풍, 공기조절/냉난방, 화재 보호 등과 같은 관련된 서브 시스템을 조절하는 환경 제어 장치 등 몇 개의 기능적인 레이아웃 영역으로 구성

데이터 센터 기술 ④

- 컴퓨팅 하드웨어
  - 전원, 네트워크 내부 냉각을 위해 상호 연결된 표준화된 랙으로 구성된 랙 마운트 폼 팩터 서버 디자인
  - X86-32bits, x86-64, RISC와 같은 여러 하드웨어 프로세싱 아키텍처 지원
  - 표준화된 랙만큼 작은 공간에 수백 개의 프로세싱 코어를 가지고 있는 전원 효율적인 멀티코어 CPU 아키텍처
  - 하드디스크, 전원공급장치, 네트워크 인터페이스, 스토리지 컨트롤러 카드와 같이 전원 차단 없이 추가/제거될 수 있으며 중복 구성된 컴포넌트
  - 고밀도 서버 기술과 같은 컴퓨팅 아키텍처는 랙 내장 물리적 상호접속blade enclosure, 섬유(스위치), 공유 전원 공급 단위 및 냉각 팬을 사용
  - 상호접속은 물리적 공간과 전원을 최적화하면서 컴포넌트 간 네트워킹과 관리를 향상
  - 전형적으로 개별적인 서버의 전원 차단 없는 교체hot-swapping, 확장, 교체, 유지보수를 지원하는데, 이는 컴퓨터 클러스터를 기반으로 한 고장방지 시스템의 배포를 용이하게 함

데이터 센터 기술 ⑤

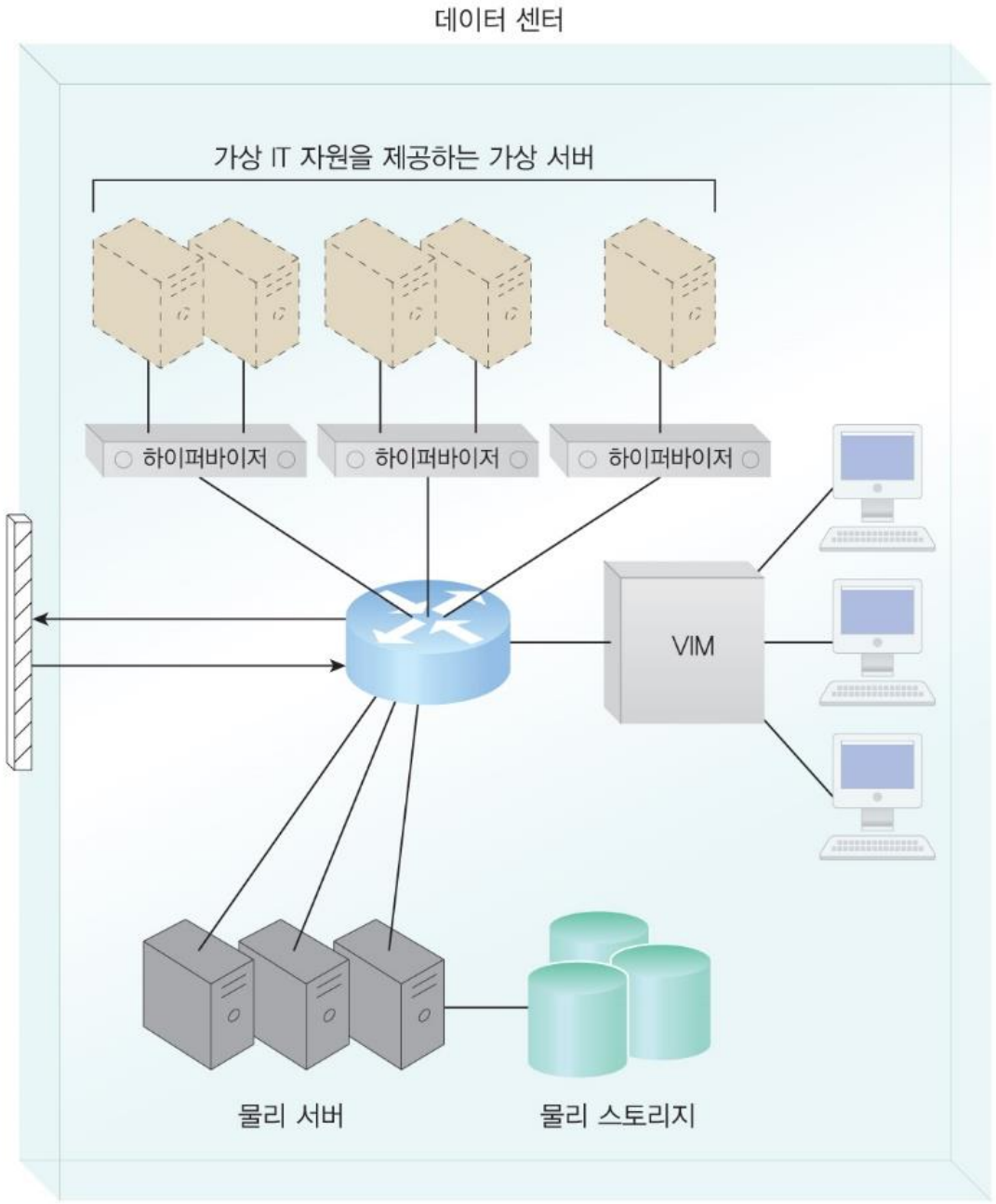
- 스토리지 하드웨어
  - 하드디스크 배열: 배열은 여러 물리적 드라이버 사이에 데이터를 나누고 복제하며 여분의 디스크를 포함하여 성능과 중복성을 향상시킴 (레이드RAID, Redundant Arrays of Independent Disks 사용)
  - 입출력 캐싱: 데이터 캐싱을 이용해 디스크 접근 시간과 성능을 향상시키는 하드 디스크 배열 컨트롤러를 이용해 수행
  - 전원 차단 없이 교체 가능한 하드디스크: 전원 차단 없이 배열에서 안전하게 제거 가능
  - 스토리지 가상화: 가상 하드디스크와 스토리지 공유로 이루어짐
  - 빠른 데이터 복제 메커니즘: 리로딩이나 가상 또는 물리 하드디스크 용량 및 파티션을 복사하는 용량 복제volume cloning를 위해 가상 머신의 메모리에서 하이퍼바이저가 읽을 수 있는 파일ㅇ 형태로 저장하는 스냅샷 기능 포함
  - SANStorage Area Network: 물리적 데이터 스토리지 미디어는 전용 네트워크를 통해 연결되며 SCSI와 같은 업계 표준 프로토콜을 이용하여 블럭 수준의 접근을 제공
  - NASNetwork Attached Storage: 하드 드라이브 배열은 네트워크를 통해 연결하고 네트워크 파일 시스템이나 서버 메시지 블록과 같은 파일 중심의 데이터 접근 프로토콜을 이용하여 데이터 접근

데이터 센터 기술 ⑥

- 네트워크 하드웨어
  - 전달자와 외부 네트워크 상호접속: 인터네트워킹 인프라와 관련된 서브시스템으로 상호접속은 대개 외부 WAN과 데이터 센터의 LAN 사이의 라우팅을 제공하는 백본 라우터와 방화벽, VPN 게이트웨이 같은 주변 네트워크 보안 장치로 구성
  - 웹 계층 로드 밸런싱과 가속화: XML 전처리기, 암복호화 기기, 내용 인식 라우팅을 수행하는 7계층 스위칭 장치와 같은 웹 가속화 장치로 이루어짐
  - LAN 섬유: 내부 LAN을 구성하며 높은 성능과 데이터 센터의 네트워크 연결 가능한 IT 자원에 대한 중복 연결성 제공. 초당 10기가 비트 속도까지 운영하는 여러 네트워크 스위치로 구현
  - SAN 섬유: 파이버 채널FC, Fibre Channel, 파이퍼 채널 오버 이더넷FCoE, Fibre Channel over Ethernet, 인피니밴드InfiniBand 네트워크 스위치
  - NAS 게이트웨이: NAS 기반 스토리지 장치에 부착점을 제공하고 SAN과 NAS 장치 사이의 데이터 전송을 돕는 프로토콜 전환 하드웨어를 구현

가상화 기술

- 하드웨어 독립성
- 서버 통합
- 자원 복제
- 운영체제 기반 가상화
- 하드웨어 기반 가상화
- 가상화 관리



가상화 가능한 IT 자원

- 서버: 물리 서버는 가상 서버로 추상화
- 스토리지: 물리 스토리지 장치는 가상 스토리지 장치나 가상 디스크로 추상화
- 네트워크: 물리 라우터와 스위치는 VLAN과 같은 논리적인 네트워크 섬유로 추상화
- 전원: 물리적 UPS와 전원 분배 장치는 가상 UPS라 불리는 장비로 추상화

운영체제 기반 가상화


가상 머신
(게스트 운영체제와
애플리케이션
소프트웨어)
가상 머신
(게스트 운영체제와
애플리케이션
소프트웨어)
가상 머신
(게스트 운영체제와
애플리케이션
소프트웨어)
가상 머신 관리
운영체제
(호스트 운용 체제)
하드웨어
(가상 호스트)

하드웨어 기반 가상화


가상 머신
(게스트 운영체제와
애플리케이션
소프트웨어)
가상 머신
(게스트 운영체제와
애플리케이션
소프트웨어)
가상 머신
(게스트 운영체제와
애플리케이션
소프트웨어)
가상 머신 관리 하이퍼바이저
하드웨어(가상 호스트)

하이퍼바이저

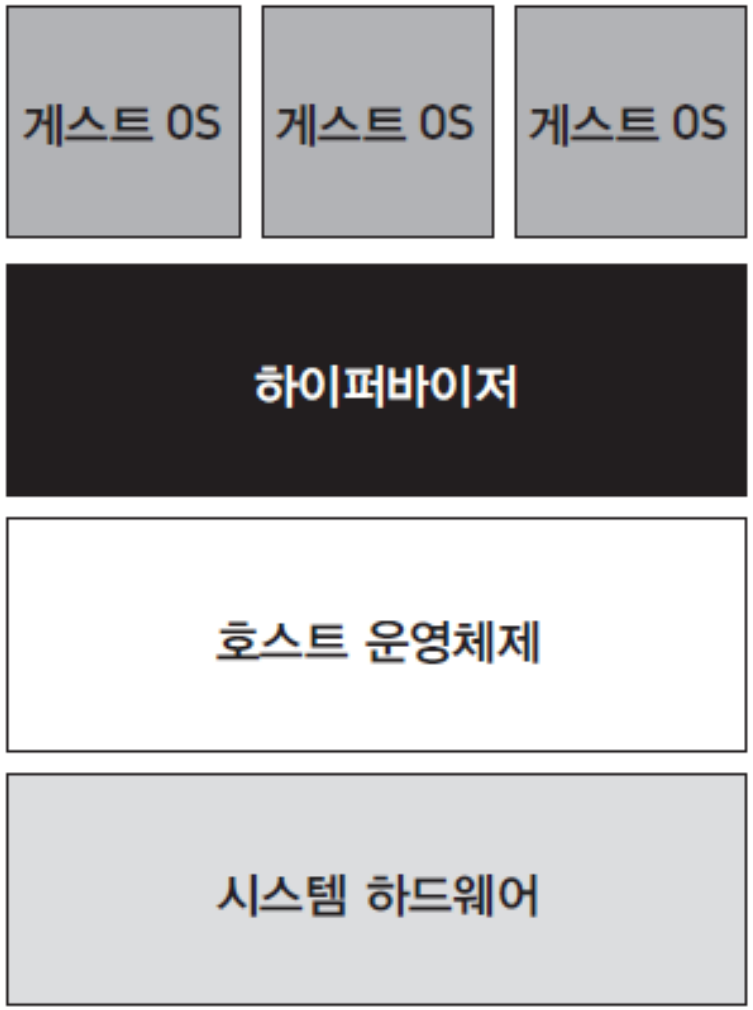
- 로드밸런싱은 논리적 주소를 물리적 주소로 매핑시켜서 시스템과 자원을 가상화
- 특정 자원의 모음과 함께 주어진 컴퓨터 시스템은 가상머신을 생성하기 위한 자원의 일부를 따로 구분할 수 있음
- 애플리케이션이나 사용자의 관점에서 가상머신은 물리적 시스템의 모든 속성과 특성을 가지지만, 물리적 머신을 에뮬레이트하는 엄격한 소프트웨어
- 시스템 가상머신(또는 하드웨어 가상머신)은 스스로의 가상 기기 드라이버, 프로세서 자원 할당, 가상 기기 드라이버를 이용한 기기 I/O를 가짐
- 가상머신은 가상머신이 실행되고 있는 물리적 컴퓨터로부터 분리된 또 하나의 컴퓨터
- 이것은 가상머신 기술이 오래된 버전의 운영체제를 실행하는 것과 샌드박스에서 애플리케이션을 테스트하는 것을 매우 유용하게 함
- 클라우드 컴퓨팅의 경우에는 워크로드가 할당될 수 있는 가상머신 인스턴스를 생성하는데 유용

가상머신의 종류 ①

- 저수준 프로그램은 가상머신에 시스템 자원에 대한 접근을 제공해야 하고, 이 프로그램은 하이퍼바이저 또는 VMM(Virtual Machine Monitor)이라고 함
- 베어메탈(bare metal) 상에서 실행되는 하이퍼바이저는 Type1 VM 또는 네이티브 VM
- Type1 VM은 베어시스템에 설치되기 때문에 호스트 운영체제가 없으며 하드웨어 상에서 실행되는 하드웨어의 시뮬레이션이 완벽하기 때문에 완벽한 가상화
- 어떤 하이퍼바이저는 운영체제상에 설치되는 Type2이거나 호스트된 VM
- Type2 가상머신은 호스트 운영체제 위에 설치

가상머신의 종류 ②

Type 1 하이퍼바이저

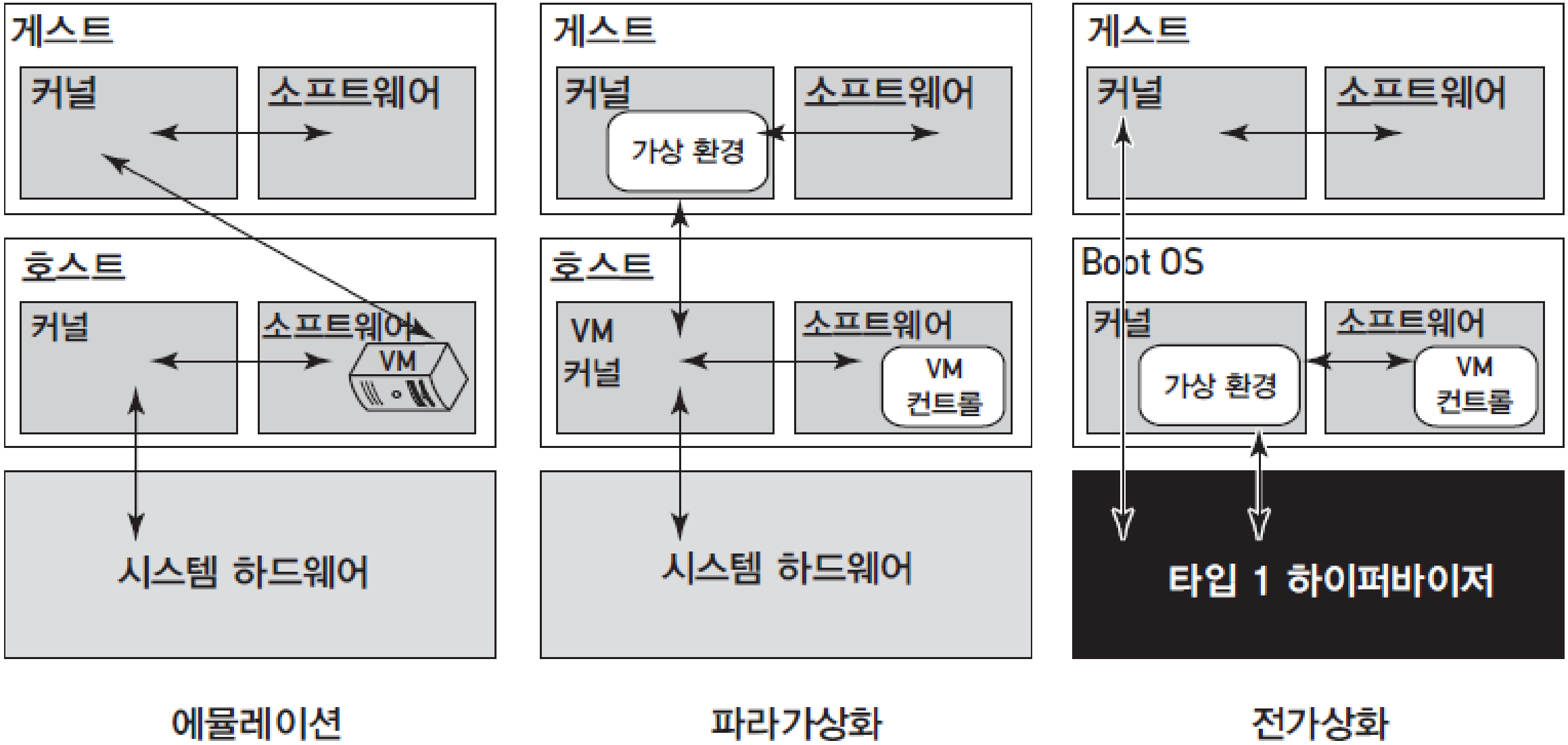


Type 2 하이퍼바이저

가상머신의 종류 ③

- Type2 VM에서 소프트웨어 인터페이스는 시스템이 일반적으로 상호작용하는 기기를 에뮬레이트하도록 생성
- 이 추상화는 가상환경의 내부에서 기기 I/O를 실행하는 것보다 더 쉽고, 효율적이게 함
- 이러한 가상화는 반가상화(paravirtualization)라고 하고, 마이크로소프트의 Hyper-V와 Xen 같은 하이퍼바이저에 적용

가상머신의 종류 ④


게스트
커널
소프트웨어
호스트
커널
소프트웨어
VM
시스템 하드웨어
에뮬레이션
게스트
커널
가상 환경
소프트웨어
호스트
VM
커널
소프트웨어
VM
컨트롤
시스템 하드웨어
파라가상화
게스트
커널
소프트웨어
Boot OS
커널
가상 환경
소프트웨어
VM
컨트롤
타입 1 하이퍼바이저
전가상화

가상머신의 종류 ⑤

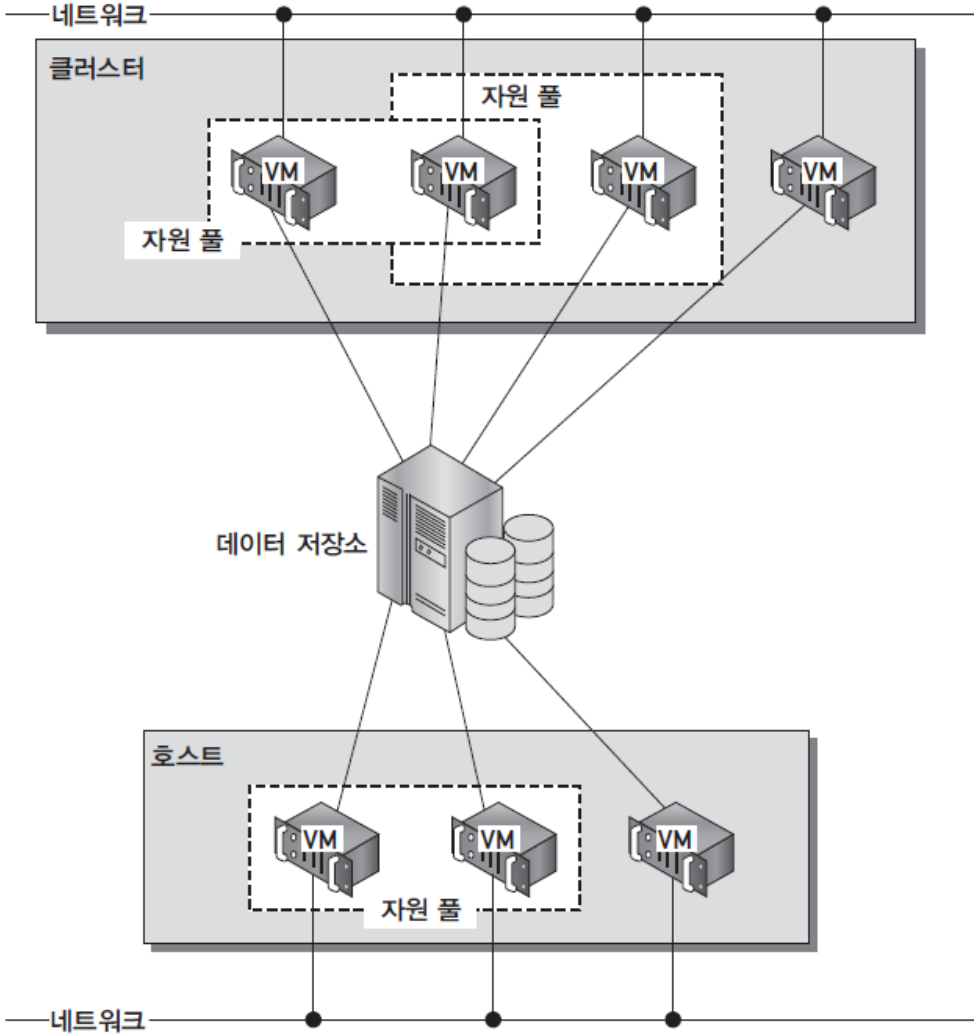
- 에뮬레이션에서 가상 머신은 하드웨어를 시뮬레이션함으로써 시스템 하드웨어에 독립적으로 동작
- 반가상화를 위해서는 호스트 운영체제가 게스트 운영체제를 위한 가상머신 인터페이스를 제공하는 것이 필요하고, 게스트 운영체제는 호스트 VM을 통하여 하드웨어에 접근
- 전체 가상화 구조에서 VM은 하드웨어에 Type1 하이퍼바이저로 직접 설치
- 전체가상화의 모든 운영체제는 VM 하이퍼바이저와 직접 통신하고, 게스트 운영체제는 수정될 필요가 없음
- VMI(Virtual Machine Interface)는 VMware에서 제안한 반가상화 API의 예를 표준(vmi.ncsa.uiuc.edu)으로 개방
- 최신 버전의 VMI는 2.1이고, 다양한 버전의 리눅스 운영체제에 기본 설치로 포함

가상머신의 종류 ⑥

- 애플리케이션 가상머신 내부에서 실행되는 애플리케이션은 일반적으로 느림
- 그러나 휴대성을 제공하고 다양한 프로그래밍 언어를 제공하고 많은 발전된 기능 사용 가능
- 해당 프로그램이 플랫폼 독립적으로 실행될 수 있게 하기 때문에 사용자들이 선호하는 편

가상 인프라 요소들

- 중앙에 있는 데이터 저장소는 공유 저장장치 자원
- 이 저장장치 자원은 SCSI, SAS를 사용하는 서버의 DAS(Direct Attached Storage)이거나 SATA 연결, Fiber Channel disk arrays/SANs, iSCSI disk 배열, 또는 NAS(Network Attached Storage) 디스크 배열



저장장치 & 네트워크 가상화

- 저장장치 가상화는 대부분 논리적 저장장치 주소가 물리적 저장장치 주소의 어디로 이동되느냐하는 매핑 메커니즘을 통해 구현되는 것이 일반적
- SAN에서 사용되는 블록 기반 저장장치는 LUN(Logical Unit Identifier)이라고 불리는 기능을 사용
- 네트워크 가상화는 네트워킹 하드웨어와 소프트웨어를 관리할 수 있는 가상 네트워크 안으로 추상화
- 가상 네트워크는 VNICs(virtual network interfaces) 또는 VLANs(virtual LANs)을 생성할 수 있고 하이퍼바이저, 운영체제, 또는 외부 관리 콘솔에 의해 관리될 수 있음
- 클라우드 컴퓨팅 솔루션을 구현하기 위한 매우 매력적인 가상 인프라를 만드는 핵심 기능은 유동성

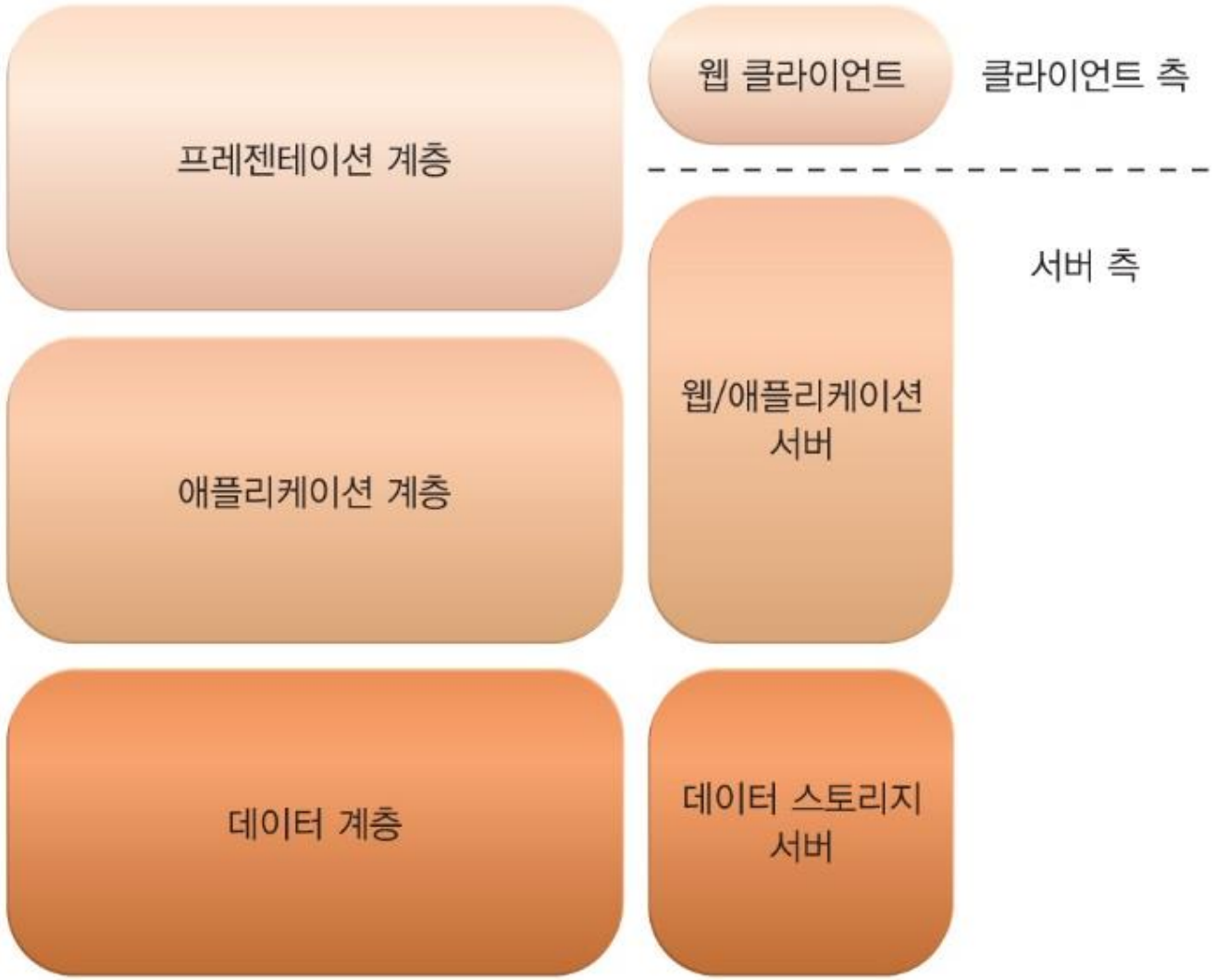
머신 이미지

- 시스템 이미지는 단일 컨테이너 안에 전체 컴퓨터 시스템의 복제본을 파일로 만드는 일이며, 시스템 이미징 프로그램은 이 이미지를 만들고 나중에 복구할 수 있음
- 시스템 이미지가 클라우드 컴퓨팅 아키텍처에서 어떻게 사용되는가를 보여주는 중요한 예는 가상머신의 복사본을 저장하기 위해 아마존 웹서비스에서 사용하는 AMI(Amazon Machine Image)
- AMI는 운영체제, 모든 올바른 기기 드라이버, 애플리케이션, 가상 머신이 작동했을 때의 상태정보를 포함하는 파일 시스템
- 아마존 웹 서비스를 사용한다면, 수백 개의 미리 만들어진 AMI 중의 하나를 선택해서 사용할 수 있고 사용자 정의 시스템을 생성하여 AMI로 만들 수 있음
- AMI는 자유 배포 라이센스 아래에서 공개적으로 사용할 수 있고 운영체제는 사용한 만큼 지불해야 하고 접근 권한이 주어진 사용자인 EC2 사용자에 의해 공유

웹 기술

- URL(Uniform Resource Locator): 웹 기반 자원을 가리키는 식별자를 생성하는 표준 문법으로 논리적 네트워크 위치를 사용하여 구성
- HTTP(HyperText Transfer Protocol): WWW를 통해 내용과 데이터를 교환하기 위해 사용되는 기본적인 통신 프로토콜 (URL은 HTTP를 통해 전송)
- 마크업 언어: 마크업 언어는 웹 중심 데이터와 메타데이터를 표현하는 수단을 제공
  - HTML: 웹 페이지를 표현하기 위해 사용
  - XML: 메타데이터를 통한 웹 기반 데이터와 사용되는 단어의 정의를 위해 사용

웹 애플리케이션


프레젠테이션 계층
애플리케이션 계층
데이터 계층
웹 클라이언트
클라이언트 측
서버 측
웹/애플리케이션 서버
데이터 스토리지 서버

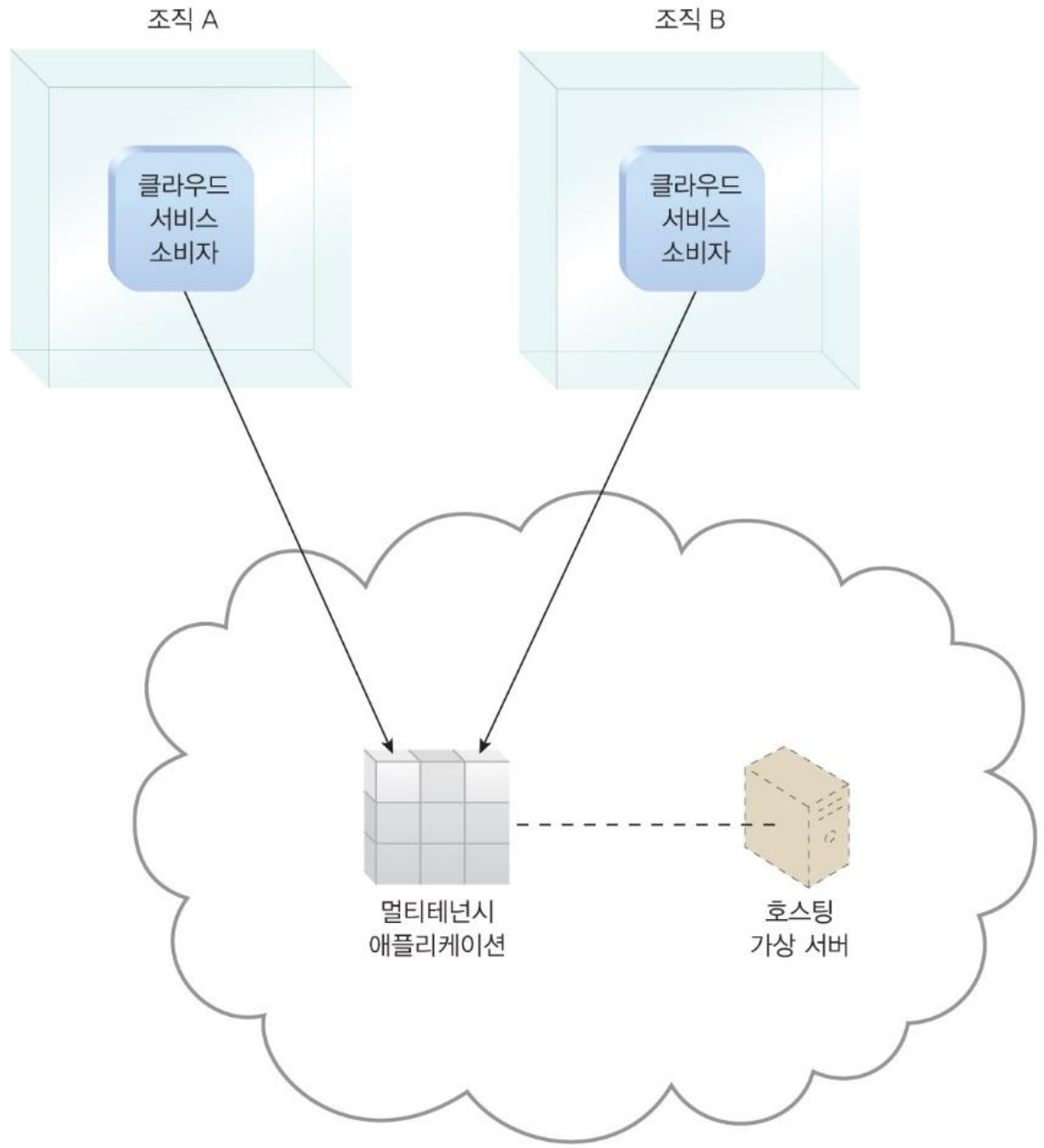
멀티테넌시 기술

- 멀티테넌시 애플리케이션 설계는 여러 테넌트가 같은 애플리케이션 로직에 동시 접근이 가능하게 만드는 기술
- 각 테넌트는 같은 애플리케이션을 사용하고 있다는 것을 인식하지 못한 채 사용하고 관리
- 테넌트는 다음과 같은 애플리케이션 특성을 개별적으로 원하는 대로 생성
  - 사용자 인터페이스: 특화된 'Look and Feel'로 정의 가능
  - 비즈니스 프로세스: 비즈니스 프로세서의 규칙, 로직, 워크플로우를 원하는 대로 생성 가능
  - 데이터 모델: 데이터 구조에 있는 필드를 포함하거나 제외, 재명명하기 위해 데이터 스키마 확장 가능
  - 접근 제어: 사용자와 그룹에 대한 접근 권한을 독립적으로 통제 가능

멀티테넌시 애플리케이션의 특징

- 사용 분리: 한 테넌트의 사용 행위가 다른 테넌트의 애플리케이션의 가용성과 성능에 영향을 주지 않음
- 데이터 보안: 테넌트는 다른 테넌트에 속한 데이터에 접근할 수 없음
- 복구: 백업과 복구 처리는 각 테넌트의 데이터에 독립적으로 수행
- 애플리케이션 업그레이드: 테넌트는 공유 소프트웨어 산출물의 동시 업그레이드에 의해 영향을 받지 않음
- 확장성: 애플리케이션은 존재하는 테넌트의 사용량 증가나 테넌트의 수의 증가를 수용하기 위해 확장할 수 있음
- 사용량 측정: 테넌트는 실제 소비된 애플케이션 프로세싱과 특징에 대해서만 과금
- 데이터 계층 분리: 테넌트는 다른 테넌트와 독립적인 데이터베이스, 테이블, 스키마를 가짐 (의도적으로 데이터베이스, 테이블, 스키마가 공유되도록 설계할 수는 있음)

멀티테넌시 애플리케이션



가상화 vs. 멀티테넌시

- 호스트 역할을 하는물리 서버 내에 무엇이 여러 개 존재하는지가 차이점
- 가상화
  - 서버 환경의 여러 가상 서버는 하나의 물리 서버에 의해 제공 가능
  - 가상 서버는 다른 사용자에게 제공될 수 있음
  - 가상 서버는 독립적으로 설정될 수 있고 각 운영체제와 애플리케이션을 포함
- 멀티테넌시
  - 애플리케이션을 제공하는 물리 또는 가상 서버가 여러 다른 사용자가 사용할 수 있도록 설계
  - 각 사용자는 애플리케이션을 베타적으로 사용한다고 느낌

웹 서비스

- 웹 서비스 기술 언어WSDL, Web Service Description Language
  - 웹 서비스의 개별 동작과 각 동작의 입력과 출력 메시지를 포함하는 API 정의
- XML 스키마 정의 언어
  - 웹 서비스에서 교환되는 메시지는 XML을 사용해 표현
  - XML 기반 웹 서비스에 의해 교환되는 메시지의 데이터 구조를 정의
- 단순 객체 접근 프로토콜SOAP, Simple Object Access Protocol
  - 웹 서비스에 의해 교환되는 요청 및 응답 메시지에 사용되는 일반적인 메시징 형식
  - SOAP 메시지는 바디와 헤더 부분으로 구성되며, 바디는 메시지 컨텐츠를 저장하고, 헤더는 런타임 시 처리되는 메타데이터를 포함
- 전역 비즈니스 레지스트리UDDI, Universal Description, Discovery, and Integration
  - 서비스 카탈로그의 일부로 공개될 수 있는 서비스 레지스트리를 규제

REST 서비스

- REST 서비스는 독립적인 기술 인터페이스를 갖지 않고 대신에 HTTP를 통해 수립되는 공통 계약uniform contract 으로 알려진 공통 기술 인터페이스를 공유
- REST 설계의 제약
  - 클라이언트 서버
  - 상태 비보존
  - 캐쉬
  - 인터페이스/공통 계약
  - 계층적 시스템
  - 주문형 코드

서비스 에이전트

- 서비스 에이전트는 런타임 시 메시지를 가로채도록 설계된 이벤트 기반 프로그램
- 능동형 서비스 에이전트: 주로 메시지 컨텐츠를 변경시키거나 메시지 경로 자체를 변경
- 수동형 서비스 에이전트: 메시지 컨텐츠는 변경하지 않고, 대신 메시지를 읽고 모니터링이나 로깅, 보고의 목적으로 컨텐츠의 특정 부분을 캡처

서비스 미들웨어

- 메시지 기반 미들웨어MOM, Messaging-Oriented Middleware는 복잡한 서비스 구성을 수용하도록 설계된 정교한 서비스 미들웨어 플랫폼 통합을 용이하게 함
- 엔터프라이즈 서비스 버스EBS, Enterprise Service Bus: 서비스 브로커, 라우팅, 메시지 대기와 같은 다양한 중재 프로세싱 특징을 수반
- 통합 플랫폼: 서비스의 런타임 구성요소들을 구동시키는 워크플로우 로직을 제공하고 실행하기 위해 설계