



Clouding Computing

목차

00. 개요

01. 클라우드 컴퓨팅의 이해

02. 클라우드 컴퓨팅의 핵심 기술

03. 클라우드 네이티브

04. 클라우드 컴퓨팅의 활용

학습목표

- 클라우드 컴퓨팅의 개념을 이해한다.
- 클라우드 컴퓨팅의 종류를 이해한다.
- 클라우드 컴퓨팅의 핵심 기술을 이해한다.
- 클라우드 컴퓨팅의 활용 방안을 이해한다.

00. 개요

■ 클라우드 컴퓨팅에 대해서 알아보기

01. 클라우드 컴퓨팅의 이해

■ 클라우드 컴퓨팅의 개념

- 클라우드
 - 인터넷을 의미하고 컴퓨팅은 넓은 의미로 컴퓨팅 자원을 사용하는 모든 활동
 - 컴퓨터 네트워크 구성도에서 인터넷을 구름 모양으로 나타내는 데에서 유래
- 클라우드 컴퓨팅
 - 인터넷을 기반으로 하는 것으로, 자신의 컴퓨터가 아닌 인터넷에 연결된 컴퓨팅 자원을 이용하는 것을 의미
 - 컴퓨팅 자원: 네트워크, 데이터베이스, 서버, 스토리지, 애플리케이션 등
- 클라우드 컴퓨팅의 문헌별 정의
 - 위키피디아: 인터넷을 통해 가상화된 컴퓨팅 자원을 요구하는 즉시 제공(온디맨드)하는 것
 - 가트너: 인터넷 기술을 이용해 컴퓨팅 자원을 유연하고 확장 가능하게 제공하는 서비스
 - IBM: 애플리케이션부터 데이터센터까지 모든 온디맨드 컴퓨팅 자원을 인터넷을 통해 이용 단위 부과 방식으로 제공
 - 아마존: 컴퓨팅 자원을 인터넷을 통해 온디맨드로 제공하고 이용한 만큼만 비용을 받는 것

01. 클라우드 컴퓨팅의 이해

■ 클라우드 컴퓨팅의 개념

■ 클라우드 컴퓨팅의 이용 방식

- 가장 큰 차이는 컴퓨팅 자원을 소유하는 게 아니라 공유한다는 점
 - 전통 방식의 컴퓨팅: 컴퓨팅 자원을 구매해서 소유하는 개념
 - 클라우드 컴퓨팅: 내가 원하는 자원을 일정 기간 동안 이용하고 이용한 만큼 서비스 이용료를 지불하는 방식
- 사용자는 컴퓨팅 자원의 배치와 할당을 신경쓰지 않고 빌린 컴퓨팅 자원을 통해 실제 수행하려는 작업에 집중



그림 6-1 클라우드 컴퓨팅의 개요

01. 클라우드 컴퓨팅의 이해

■ 클라우드 컴퓨팅의 개념

■ 클라우드 컴퓨팅의 장단점

• 장점

- 클라우드 컴퓨팅을 도입하면 사용자 컴퓨팅 자원의 유지보수, 확장, 관리의 부담을 줄일 수 있음

• 단점

- 클라우드 컴퓨팅 자원은 인터넷에 연결되어 있기 때문에 해킹으로 사용자 개인정보가 유출될 수도 있음
- 사용자 데이터가 물리적으로 어디에 위치해 있는지 특정 짓기 어려움
- 클라우드 컴퓨팅의 특징은 개인정보는 반드시 자국 내에 위치한 서버에 보관해야 한다는 법과 충돌

01. 클라우드 컴퓨팅의 이해

■ 클라우드 컴퓨팅의 역사

- 클라우드 컴퓨팅의 개념은 오래전에 등장

- 1960~70년대의 호스트-터미널 구조는 모든 컴퓨팅 자원을 데이터센터에서 중앙집중식으로 구성
- 사용자 컴퓨터는 단순히 데이터센터에 연결하는 단말기 역할 수행

■ 메인프레임 시대

- 메인프레임에 터미널이 접속해서 필요한 자원을 활용하는 방식
- 메인프레임: 강력한 컴퓨팅 자원을 갖춘 컴퓨터
- 터미널: 적은 리소스의 단말, 콘솔 화면에서 메인프레임에 접속해 키보드로 문자를 입력해 지시를 내리는 단순 기기
- 메인프레임 시대는 PC의 보급으로 종결



그림 6-2 IBM 704 메인프레임(1964)(출처: 위키피디아)

01. 클라우드 컴퓨팅의 이해

■ 클라우드 컴퓨팅의 역사

■ 클라이언트-서버 시대

- 1980년대에 PC가 등장하면서 메인프레임의 일부 역할을 PC에서 실행할 수 있게 되면서 클라이언트와 서버 구조로 분산
- 클라이언트: 서비스를 사용하는 사용자 단말 / 서버: 다수의 클라이언트를 지원하기 위해 컴퓨팅 자원을 모아둔 컴퓨터
- 역할 분산은 주로 데이터는 서버에 배치 하고 애플리케이션은 클라이언트에서 실행하는 구조가 일반적
- 클라이언트와 서버 사이의 통신은 인터넷을 이용
- 클라이언트가 필요한 기능을 요청하면 서버가 응답하는 방식으로 상호 작용

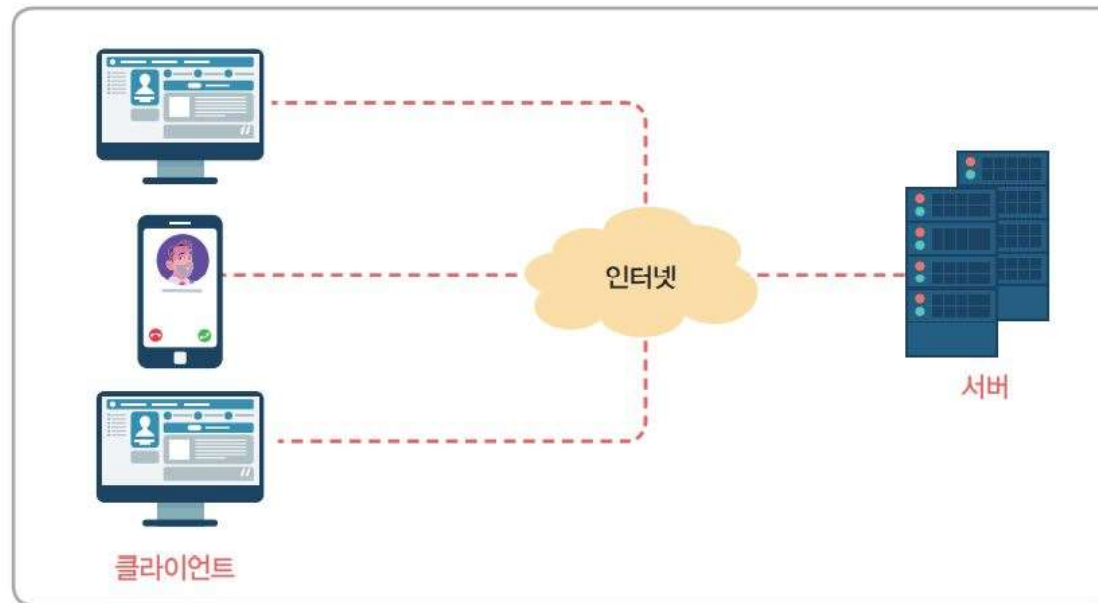


그림 6-3 클라이언트-서버 모델

01. 클라우드 컴퓨팅의 이해

■ 클라우드 컴퓨팅의 역사

■ 클라우드 컴퓨팅 시대

- 초고속 인터넷의 보급으로 웹서비스가 폭발적으로 증가하면서 서버의 수요도 증가
- 서비스 사용자가 많아질수록 서버의 과부하를 줄이기 위해 컴퓨팅 자원을 확장했으나 관리의 복잡도와 난이도는 증가
- 클라이언트-서버 구조로는 이러한 상황에 대처하기가 버거워지면서 여러 가지 공유 개념이 해결책으로 등장하기 시작
- 그리드 컴퓨팅: 대용량의 컴퓨팅 자원을 언제든지 필요할 때 제공한다는 개념, 후에 유틸리티 컴퓨팅으로 발전
- 유틸리티 컴퓨팅: 사용한 만큼 요금을 지불하는 전기나 수도처럼 종량제로 컴퓨팅 자원을 제공
- 그리드 컴퓨팅과 유틸리티 컴퓨팅의 개념은 가상화 기술과 분산 처리 기술의 발전에 힘입어 클라우드 컴퓨팅으로 발전
- 초기에는 소프트웨어 자체를 제공하는 SaaS에서 시작해 PaaS, IaaS 형태의 모델로 성공적으로 안착



그림 6-4 클라우드 컴퓨팅의 변천사

01. 클라우드 컴퓨팅의 이해

■ 클라우드 컴퓨팅의 성장

■ 민첩성

- 클라우드 컴퓨팅은 컴퓨팅 자원을 다양한 사양으로 구성하며 다양한 플랫폼 기능을 제공
- 데이터베이스, 사물인터넷, 머신러닝, 빅데이터 분석 등의 플랫폼 기능
- 전문 지식이 부족하더라도 여러 플랫폼 기능을 활용하면 비즈니스 로직에 더 집중할 수 있음
- 인터넷 쇼핑몰을 구축하면서 구매자에게 맞춤형 제품을 추천하는 경우
 - » 기존 방식: 추천 알고리즘을 연구하고 구현해서 애플리케이션을 만드는 작업을 직접 해야만 했음
 - » 클라우드 컴퓨팅: 플랫폼 기능으로 제공하는 추천 알고리즘을 사용해 자동으로 추천 아이템을 얻을 수 있음

■ 탄력성

- 클라우드 컴퓨팅은 사용자가 원하는 시간에 원하는 만큼 자원을 변경, 추가, 삭제할 수 있는 기능을 서비스 제공자가 제공
- 처음 설정한 컴퓨팅 자원을 중간에 늘리거나 줄이는 것을 쉽고 빠르게 할 수 있음
- 자동으로 자원을 늘리거나 줄이는 기능을 제공
- 웹서비스나 쇼핑몰에서 빈번하게 공지하던 새벽 중 정기점검이나 패치 작업이 클라우드 컴퓨팅 시대에는 거의 사라짐
- 인터넷 쇼핑몰에서 갑자기 한 제품이 인기를 끌며 트래픽이 몰려 과부하가 걸릴경우
 - » 기존 방식: 수작업으로 인프라를 변경하는 동안 고객은 쇼핑몰을 이용할 수 없음
 - » 클라우드 컴퓨팅: 곧바로 자원을 자동으로 증가 시켜 고객이 불편을 겪지 않도록 조치할 수 있음

01. 클라우드 컴퓨팅의 이해

■ 클라우드 컴퓨팅의 성장

■ 비용 절감

- 전통적인 컴퓨팅 환경을 직접 구축하기 위해 컴퓨팅 자원을 개별적으로 구매할 때는 가장 많이 사용하는 최대 용량에 맞춤
- 클라우드 컴퓨팅으로 쇼핑몰을 구축한다면 초기에는 필요한 컴퓨팅 자원을 최소로 구축
- 사용자가 많아지거나 구매가 늘어나면 실시간으로 필요한 만큼 늘렸다가 다시 원래대로 줄일 수 있음
- 초기 구축 때 꾸준히 이용할 것이 확실한 자원에 대해서는 클라우드 사업자가 할인을 제공하기도 하므로 비용 절감 효과를 극대화할 수 있음

■ 글로벌 배포

- 클라우드 컴퓨팅 자원을 요청할 때는 어느 지역에 배포할 것인지 선택할 수 있음
- 글로벌 동영상 스트리밍 서비스를 하고 있는 넷플릭스의 경우 클라우드 컴퓨팅을 가장 잘 사용하는 기업 중 하나

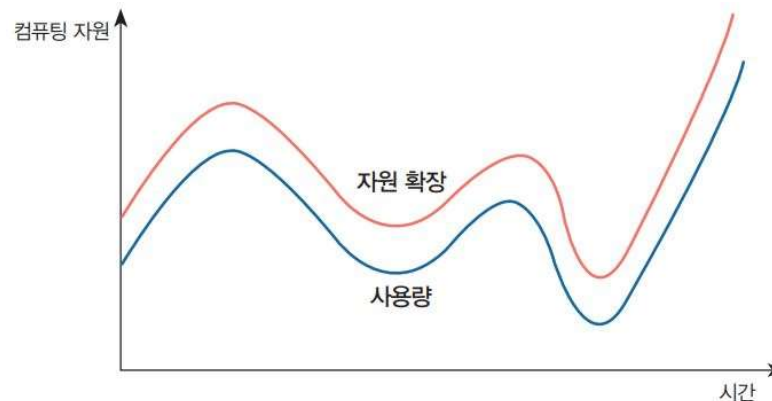


그림 6-5 사용량에 따른 컴퓨팅 자원 비용 최적화

01. 클라우드 컴퓨팅의 이해

■ 클라우드 컴퓨팅의 종류

■ 클라우드 컴퓨팅 서비스 모델

- IaaS(Infrastructure as a Service)
 - 서버의 CPU, 네트워크, 스토리지 등의 컴퓨팅 자원을 인터넷으로 제공하는 형태
 - 구성된 서버에 어떤 소프트웨어를 설치하거나 인터넷으로 접근이 가능 하도록 네트워크를 설정하는 것이 가능
 - 클라우드 컴퓨팅의 스토리지 서비스를 이용해 파일을 백업 및 복구하는 목적으로 사용하는 것도 IaaS에 속함
- PaaS(Platform as a Service)
 - 애플리케이션을 구동하는 플랫폼을 서비스로 제공하는 형태
 - IaaS만큼 자유도가 높지는 않지만 PaaS가 제공하는 서비스를 이용해 고객이 원하는 애플리케이션을 개발하거나 구동
 - IaaS가 CPU, 네트워크, 스토리지 등의 컴퓨팅 자원 자체를 빌리는 데 초점을 맞추고 있다면 PaaS 는 컴퓨팅 자원에 OS, 미들웨어와 같은 좀 더 고도화된 기능을 함께 제공
- SaaS(Software as a Service)
 - IaaS, PaaS와 달리 컴퓨팅 자원이 아니라 소프트웨어를 서비스로 제공하는 형태
 - SaaS는 IaaS, PaaS보다 자유도는 떨어지지만 제공하는 기능은 가장 많음

01. 클라우드 컴퓨팅의 이해

■ 클라우드 컴퓨팅의 종류

■ 클라우드 컴퓨팅 서비스 모델

• FaaS(Function-as-a-Service)

- 서버 인프라를 제공하는 IaaS와 달리 애플리케이션을 실행하는 컴퓨팅 시간만큼만 과금하는 서비스 모델
- 사용자가 함수만 만들어 올려두면 구동에 필요한 빌드, 실행, 모니터링 등의 나머지 과정을 자동화해 제공하는 방식
- 서버의 유휴 시간에는 별도의 과금이 없으며 최소한의 자원을 사용할 필요가 없음.
- 애플리케이션 로직만 올리기 때문에 업데이트 과정도 단순하고 빠름
- 용자가 서버 세팅을 하지 않고 애플리케이션만 등록하기 때문에 서버리스 컴퓨팅이라고도 부름



그림 6-6 IaaS, PaaS, SaaS 비교

01. 클라우드 컴퓨팅의 이해

■ 클라우드 컴퓨팅의 종류

■ 퍼블릭 클라우드와 프라이빗 클라우드

• 퍼블릭 클라우드

- 불특정 다수의 개인이나 기업에 컴퓨팅 자원을 빌려주는 형태
- 물리적으로 떨어진 곳에 서버가 있다는 사실로 인한 보안 우려가 있음
- 개인정보보호 관련 법률에 의해 데이터 보관 장소가 엄격히 제한되는 경우 퍼블릭 클라우드 도입이 어려움
- 퍼블릭 클라우드의 특정 기술에 종속되면 향후에 온프레미스로 옮기거나 다른 퍼블릭 클라우드 업체로 전환하기 어려움
- 클라우드 컴퓨팅에서의 클라우드는 일반적으로 퍼블릭 클라우드를 의미

• 프라이빗 클라우드

- 클라우드 컴퓨팅 개념을 자사 인프라에 도입해 데이터센터를 고도화하는 것
- 퍼블릭 클라우드의 단점인 물리적 보안에 대한 컨트롤이 가능하고 데이터 저장소를 명확히 확인할 수 있음
- 특정 퍼블릭 클라우드 기술에 대한 종속성을 없앨 수 있음
- 온프레미스에 클라우드 컴퓨팅 기술을 도입하는 것이기 때문에 운영이나 구축 비용에 대한 절감 효과를 보기 어려운 한계
- 퍼블릭 클라우드 도입이 어려운 경우에는 프라이빗 클라우드를 구축해서 사용
- 둘 이상의 퍼블릭 클라우드 또는 퍼블릭 클라우드와 프라이빗 클라우드를 함께 이용할 수도 있음 (하이브리드 클라우드)

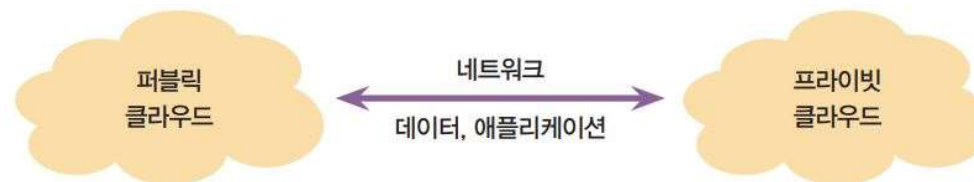


그림 6-7 퍼블릭 클라우드와 프라이빗 클라우드

02. 클라우드 컴퓨팅의 핵심 기술

■ 클라우드 컴퓨팅의 주요 특징

- 확장성
 - 자원의 확장 및 축소가 사용자가 원하는 시간에 빠르게 이루어짐
- 가용성
 - 물리적인 하드웨어의 고장이나 장애에도 사용자에게 할당된 자원은 사용 가능
- 접근성
 - 네트워크를 통해 접속이 가능
- 측정 가능성
 - 사용자가 요청해서 사용한 리소스를 측정해서 과금
- 경제성
 - 물리적인 하드웨어를 최대한 효율적으로 사용자에게 배분
- 보안성
 - 사용자의 데이터와 컴퓨팅 자원을 안전하게 보호

02. 클라우드 컴퓨팅의 핵심 기술

■ 클라우드 컴퓨팅의 핵심 기술

- 서버와 네트워크를 가상으로 연결해주는 가상화 기술
- 대용량 스토리지 클라우드를 제공하기 위한 분산 스토리지 기술
- 가상화의 최근 트렌드인 컨테이너 기술

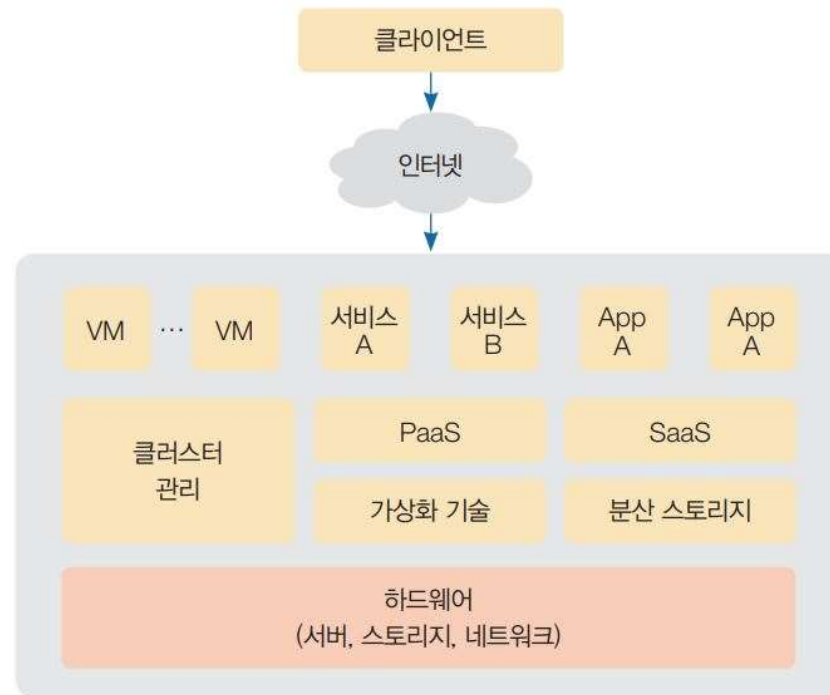


그림 6-8 클라우드 컴퓨팅의 시스템 구조

02. 클라우드 컴퓨팅의 핵심 기술

■ 가상화 기술

■ 가상화 기술의 개념

- 물리적인 컴퓨팅 자원을 논리적인 단위로 추상화하는 것
- 여러 하드웨어 장비를 하나로 묶거나 하나의 커다란 장비를 여러 대의 작은 장비인 것처럼 나눌 수 있음
- 가상화 기술은 클라우드 컴퓨팅을 위해 개발된 것이 아니라 유휴 컴퓨팅 자원의 재활용을 목적으로 개발
- 클라우드 컴퓨팅은 가상화 기술을 사용해서 물리적인 컴퓨팅 자원을 최대한 많은 사용자에게 효율적으로 할당

■ 서버 가상화

- 가상화 기술이 적용되는 컴퓨팅 자원은 CPU, 메모리, 네트워크, 스토리지를 포함
- 가상화된 서버 구조에서는 하이퍼바이저라고 부르는 가상화 소프트웨어를 통해 가상머신을 관리



(a) 일반적인 서버 구조

(b) 가상화된 서버 구조

그림 6-9 가상화된 서버 구조

02. 클라우드 컴퓨팅의 핵심 기술

■ 가상화 기술

■ 컨테이너 기술

- 컨테이너: 모듈화되고 격리된 컴퓨팅 환경
- 가상머신도 넓은 의미로는 운영체제와 애플리케이션으로 구성된 컨테이너
- 가상머신과 달리 컨테이너는 애플리케이션만 격리하므로 가상머신보다 가볍고 빠르게 동작할 수 있음
- 컨테이너의 개념은 2000년대 리눅스에서 LXC 기술이 도입되면서 시작
- 프로그램을 다른 구동 환경에서 돌리더라도 안정적으로 실행시키기 위해 고안
- 최근 컨테이너 기술 기반의 가상화가 각광받으며 가상머신 기반의 가상화를 대체



그림 6-10 가상머신과 컨테이너 기술

02. 클라우드 컴퓨팅의 핵심 기술

■ 가상화 기술

■ 도커와 쿠버네티스

• 도커

- 컨테이너 기반의 오픈소스 가상화 플랫폼
- 애플리케이션 개발과 테스트를 거쳐 컨테이너 이미지를 만들어 상용 환경에 배포하기 위한 준비를 담당
- 도커에서 가장 중요한 개념은 이미지와 컨테이너
 - » 이미지: 컨테이너 구동에 필요한 실행 파일과 관련 라이브러리, 설정값 등을 포함하는 패키지
 - » 컨테이너: 이미지를 실행한 상태

• 쿠버네티스

- 컨테이너를 쉽고 빠르게 배포하고 자동으로 관리해주는 오픈소스 플랫폼
- 쿠버네티스의 주요기능
- 서비스 디스커버리 & 로드밸런싱: 네트워크 트래픽을 컨테이너별로 분산
- 스토리지 관리: 로컬 저장소와 클라우드 저장소를 탑재
- 자동 롤아웃 & 롤백: 신규 컨테이너를 생성하고 기존 컨테이너를 소멸
- 자동화된 빈패킹: 컨테이너에 효율적으로 자원을 할당
- 자동 복구: 오류가 발생한 컨테이너는 자동으로 재시작
- 보안: 암호, 인증키와 같은 중요한 정보를 저장하고 관리

02. 클라우드 컴퓨팅의 핵심 기술

■ 분산 스토리지 기술

- 스토리지 가상화

- 다수의 사용자가 대량의 데이터를 저장하는 용도로 클라우드 스토리지를 활용
- 단일 스토리지로는 구현할 수 없기에 당연히 여러 스토리지를 묶어서 활용하는 방식이 고안
- 스토리지 가상화는 가상화 소프트웨어를 통해 스토리지 자원과 애플리케이션이 통신
- 서버와 스토리지 사이의 중간 계층에서 작동
- 서버 입장에서는 단일 스토리지처럼, 스토리지 입장에서는 단일 서버처럼 인식하도록 도움
- 서버와 스토리지의 변화에 유연하게 대처할 수 있음

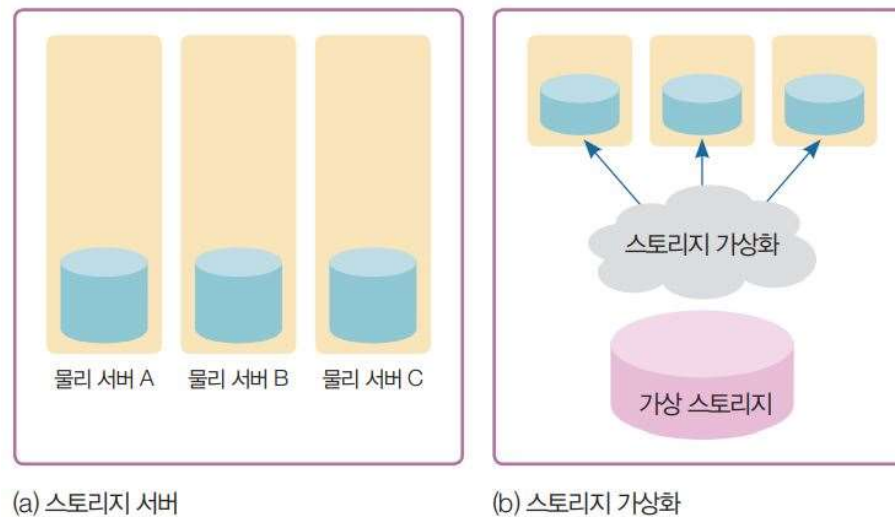


그림 6-11 전통적인 스토리지 서버와 스토리지 가상화

02. 클라우드 컴퓨팅의 핵심 기술

■ 분산 스토리지 기술

- 스토리지 가상화의 장점
 - 스토리지 활용률을 높일 수 있음
 - 입출력(I/O) 성능을 향상시킬 수 있음
 - 장애가 발생하더라도 쉽게 복구할 수 있음
 - 남은 자원을 재활용하므로 스토리지 자원 추가 비용을 절감할 수 있음
 - 장애 복구와 스토리지 추가 및 확장에 유연하게 대처할 수 있음

02. 클라우드 컴퓨팅의 핵심 기술

■ 클러스터 관리 기술

■ 클러스터

- 대량의 데이터를 저장하거나 많은 연산을 처리하기 위해 여러 대의 컴퓨팅 자원을 하나로 묶은 시스템
- 클러스터 관리의 주요 역할은 노드 관리, 업데이트 및 설치의 자동화, 프로비저닝, 로드밸런싱 등

■ 노드 관리

- 노드: 클러스터를 구성하는 개별 시스템
- 노드 관리의 목적은 시스템 가용성을 유지하는 것
- 클라우드 컴퓨팅 환경에서 노드는 가상화된 컴퓨팅 자원을 의미
- 물리 노드의 상태와 자원 활용도를 지속적으로 모니터링하면서 어떤 물리 노드에 가상 노드를 연결할지 결정

• 업데이트 및 설치의 자동화

- 클라우드 컴퓨팅 서비스 제공자들은 고객이 원하는 만큼의 컴퓨팅 자원을 안전하고 중단없이 제공해야 함
- 자동화를 통하여 클러스터에 보안 패치를 적용하거나 프로그램을 설치 및 삭제 할 수 있음

02. 클라우드 컴퓨팅의 핵심 기술

■ 클러스터 관리 기술

■ 프로비저닝

- 클러스터 내의 유휴 자원을 파악하고 필요한 순간에 자원을 빠르게 할당해서 제공하기 위한 기술
- 프로비저닝의 전제 조건은 자원의 상태를 정확하고 빠르게 모니터링 하는 것
- 프로비저닝은 실시간으로 변하는 사용자 요청에 빠르게 대처할 수 있는 방법

■ 로드밸런싱

- 시스템 부하를 적절히 나눠서 컴퓨팅 자원을 효율적으로 사용하는 기술
- 주로 네트워크 부하를 분배하거나 각 서버가 일정하게 부하를 나눠서 처리할 수 있음
- 사용자가 자동 확장의 옵션의 범위를 정하면 해당 범위 안에서 클라우드 컴퓨팅 업체는 트래픽에 따라 자동으로 컴퓨팅 자원을 확장 혹은 축소
- 예: 10대의 웹 서버로 운영 중인 쇼핑몰의 트래픽이 10대로는 감당이 안될 경우 자동으로 컴퓨팅 자원을 확장
혹은 5대로 감당할 정도로 트래픽이 적으면 자원을 축소

03. 클라우드 네이티브

■ 클라우드 네이티브의 개념

- 클라우드 네이티브의 정의와 목적
 - 클라우드 컴퓨팅의 장점을 활용하는 애플리케이션 개발 및 운영 방식을 의미
 - 개발 생산성과 운영 효율성을 극대화하는 데 목적
 - 가장 기초적인 활용은 전통적인 컴퓨팅 환경에서 동작하는 소프트웨어를 수정없이 그대로 클라우드 환경에 배포하는 것
 - 가장 이상적인 모델은 컨테이너를 이용하면서 소프트웨어를 클라우드 환경의 장점을 최대한 이용하도록 변경하는 것

표 6-1 클라우드 네이티브와 기존 방식의 차이점

	클라우드 네이티브	기존 방식
확장성	필요할 때 동적으로 확장 가능	정적이고 확장이 어려움
가용성	장애가 발생하더라도 서비스 가능	장애 발생 시 서비스 중단
업데이트	무중단 자동 업데이트	서비스 중단 후 수동 업데이트
멀티테넌시	가상 서버와 물리 서버 지원	물리 서버에서만 동작하는 경우가 있음

03. 클라우드 네이티브

■ 클라우드 네이티브의 4요소



그림 6-12 클라우드 네이티브의 4요소

03. 클라우드 네이티브

■ 클라우드 네이티브의 4요소

■ 데브옵스

- 개발(Development)과 운영(Operations)의 합성어
- 소프트웨어 개발자와 정보기술 전문가 간의 소통과 협업을 강조하는 개발환경
- 전통적인 소프트웨어 개발 방법에서 개발과 운영은 분리된 개념
- 클라우드 환경에서는 개발과 운영의 경계가 모호해졌으며 개발자가 직접 운영하는 것이 가능해짐
- 소프트웨어 개발론에서 개발과 운영을 동시에 하면서 빠르게 개발 사이클을 가져가는 애자일 방식이 각광받음

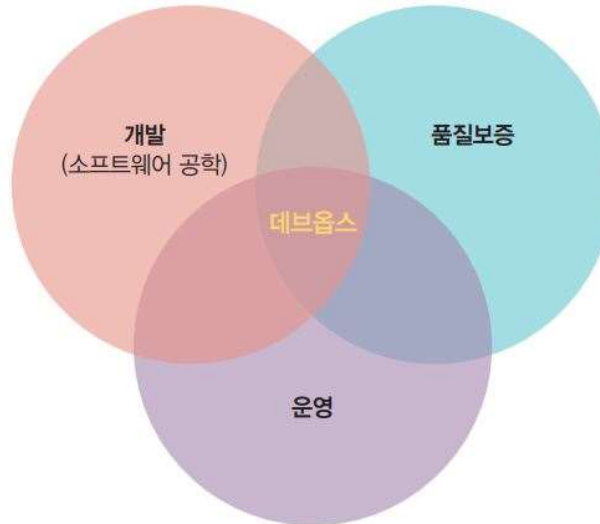


그림 6-13 데브옵스

03. 클라우드 네이티브

■ 클라우드 네이티브의 4요소

■ 지속적인 통합/배포(CI/CD)

- 소프트웨어 개발은 코딩, 빌드, 테스트, 배포 단계를 거쳐 진행
- 지속적인 통합/배포: 지속적으로 소프트웨어 품질을 점검하는 프로세스를 의미
- CI: 소프트웨어 개발자를 위한 자동화 프로세스
- CD: 개발된 소프트웨어 를 안정적으로 관리하는 운영자 과정을 위한 자동화 프로세스

03. 클라우드 네이티브

■ 클라우드 네이티브의 4요소

■ 마이크로 서비스 아키텍처(MSA)

• 모노리틱 아키텍처

- 전통적인 방식의 소프트웨어는 여러 가지 기능을 하나의 애플리케이션으로 구현하는 모노리틱 아키텍처를 사용
- 소스 코드 베이스 하나로 애플리케이션을 개발하기 때문에 빌드, 테스트, 배포 과정이 단순한 장점이 있음
- 애플리케이션이 복잡해질수록 새로운 기능을 추가할 때 소요되는 인력과 시간이 비례해서 증가
- 유지 보수 측면에서도 애플리케이션을 업데이트하는 과정에서 기능에 버그가 발생하면 다른 기능에도 영향을 끼침

• 마이크로 서비스 아키텍처의 등장

- 모노리틱 아키텍처의 단점을 극복하기 위해 애플리케이션을 기능별로 잘게 나눠 서비스로 제작
- 서비스 사이에서는 HTTP API 호출을 통해 필요한 기능을 불러 쓰는 방식을 도입

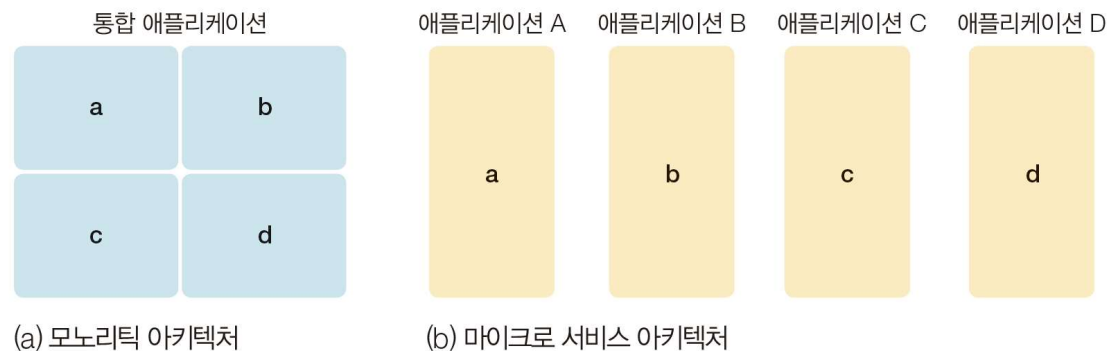


그림 6-14 모노리틱과 마이크로 서비스 아키텍처

03. 클라우드 네이티브

■ 클라우드 네이티브의 4요소

■ 마이크로 서비스 아키텍처(MSA)

• 마이크로 서비스 아키텍처의 장점

- 가장 큰 장점은 확장성
- 다양한 기능이 독립적인 서비스로 나뉘어 있어서 기능별로 배포할 수 있음
- 단일 서비스에 장애가 발생해도 해당 서비스 외의 다른 기능에 큰 영향을 주지 않음
- 개별 서비스에 새로운 기술을 적용하는 것도 상대적으로 쉬움
- 서비스 단위가 모노리틱 아키텍처에 비해 작기 때문에 새로 합류한 개발자가 기능을 이해하기도 쉬움
- 컨테이너 기술을 사용해 개별 서비스를 독립적으로 배포하면 자원을 보다 효율적으로 활용할 수 있음

• 마이크로 서비스 아키텍처의 단점

- 하나의 애플리케이션을 너무 작은 서비스 단위로 잘게 쪼개면 네트워크에 메시지를 보내는 트래픽이 많아지면서 지연
- 통합 테스트에서 문제가 발생하면 여러 서비스 중 어떤 서비스에서 발생한 문제인지 찾아내는 과정이 추가로 필요
- 데이터베이스가 분산되어 있으므로 여러 데이터베이스에 흩어진 데이터를 관리하기 어려움

■ 컨테이너

- 가상머신보다 효율적이고 빠른 속도로 애플리케이션을 개발하고 배포하는 방식으로 선호
- 쿠버네티스와 같은 오케스트레이션 툴과 조합하여 사용하면 클라우드 환경에서 더 큰 장점으로 작용

04. 클라우드 컴퓨팅의 활용

■ 퍼블릭 클라우드의 예



그림 6-15 주요 퍼블릭 클라우드 기업

■ 아마존웹서비스(AWS)

- 2006년에 클라우드를 통해 스토리지, 컴퓨팅 자원을 서비스로 제공하는 IaaS로 시작
- 스토리지나 서버 자원을 빌려서 쓰는 것부터 시작해 점차 캐시 서버, 데이터베이스 등의 플랫폼 기능을 갖추기 시작
- 최근에는 빅데이터 분석, 머신러닝과 같은 보다 발전된 형태의 서비스까지 제공
- 컴퓨팅 관련 서비스: 아마존 EC2 가상머신, ECS 컨테이너 서비스, 가상 프라이빗 서버인 라이트세일, 서버리스 컴퓨팅
- 스토리지 서비스: 심플 스토리지 서비스(S3), 일래스틱 블록 서비스(EBS), 아카이브 서비스 인 글래시어
- 데이터베이스 서비스: 관계형 데이터베이스인 오로라, MySQL용 RDS, PostgreSQL, 오라클, NoSQL인 다이나모DB

04. 클라우드 컴퓨팅의 활용

■ 퍼블릭 클라우드의 예

■ 마이크로소프트 애저

- 마이크로소프트는 IaaS, PaaS, SaaS를 모두 제공하는 업체
- 애저의 기능은 AWS와 유사
- 애저는 AWS와 차별성을 강조하기 위해 하이브리드 클라우드 컴퓨팅에 초점을 맞춤
- 마이크로소프트 기반 워크로드나 기존 마이크로소프트 애플리케이션의 퍼블릭 클라우드 부분을 활용하는 데 적합
- 컴퓨팅 서비스: 가상머신에서 애저 컨테이너 서비스, 서버리스 플랫폼인 펄션스, 마이크로 서비스 기반 애플리케이션을 연결하는 플랫폼인 서비스 패브릭, 배치 프로세싱 서비스
- 스토리지 서비스: 블롭 객체 스토리지 서비스, VM용 디스크 스토리지
- 데이터베이스 서비스: 관리형 관계 SQL 데이터베이스, 도큐먼트DB NoSQL 데이터베이스, 데이터 팩토리등의 데이터 관리 도구, 레디스 캐시 플랫폼
- 기타: 자체 사물인터넷 플랫폼, 보안 및 접근 관리 플랫폼, 개발자 도구, 모니터링 및 관리용 상품

04. 클라우드 컴퓨팅의 활용

■ 퍼블릭 클라우드의 예

■ 구글 클라우드 플랫폼(GCP)

- 구글은 PaaS 기능으로 시작하여 IaaS 기능을 지원하기 시작
- GCP의 장점은 빅데이터, 머신러닝, 컨테이너 기술에서 우위에 있음
- 기능면에서는 GCP도 AWS나 애저와 유사한 기능을 제공
- 구글은 GCP를 통해 검증된 오픈소스를 내놓으면서 기술력을 자랑(텐서플로, 쿠버네티스)
- 컴퓨팅 서비스: 가상머신, 컨테이너 엔진과 레지스트리, FaaS인 클라우드 펑션스
- 데이터베이스 서비스: 클라우드 빅테이블, 클라우드 스페너
- 빅데이터 서비스: 클라우드 데이터플로, 데이터레이크용 빅쿼리

04. 클라우드 컴퓨팅의 활용

■ 대표적인 퍼블릭 클라우드 활용 서비스

■ 비디오 스트리밍 서비스

- 퍼블릭 클라우드를 활용해 비디오 스트리밍 서비스를 제공하는 대표적인 회사에는 넷플릭스가 있음
- 넷플릭스가 빠르게 성장한 요인은 사업 초기에 퍼블릭 클라우드로 전격 전환했기 때문
- 클라우드 관련한 오픈소스를 다수 개발하여 공개(카오스 몽키, 넷플릭스 내부에서 사용하는 다양한 프레임워크)

■ 금융 서비스

- 전 세계에서 24시간 동안 발생하는 대용량 데이터를 실시간으로 분석하기 위해서는 클라우드가 적합
- 보안을 중요시하는 보수적인 은행권에서도 핵심 기능에는 프라이빗 클라우드를 도입
- 비 핵심 기능에는 퍼블릭 클라우드를 활용한 하이브리드 클라우드 도입
- 중소 은행의 경우 뱅킹 시스템 같은 핵심 시스템까지 클라우드로 이전

04. 클라우드 컴퓨팅의 활용

■ 대표적인 퍼블릭 클라우드 활용 서비스

■ 헬스케어 서비스

- 나이키는 2010년부터 기존의 사업을 확장해 종합 헬스케어 서비스로 변신하면서 클라우드를 적극 활용 (NTC)
- 전 세계 고객의 데이터를 관리 하고 분석하기 위해 퍼블릭 클라우드를 기반으로 인프라를 이전
- 필립스는 클라우드를 활용해 의료기기에서 발생하는 의료 데이터를 분석할 수 있는 HSDP를 서비스
- HSDP를 통해 데이터 분석 결과를 의사에게 제공하면 의사가 환자를 위한 맞춤형 처방을 서비스
- 의료기기에서 발생하는 데이터는 대규모이므로 클라우드 스토리지와 빅데이터 처리 기능을 활용해 서비스

■ 스마트홈 서비스

- 스마트폰 시장의 성장에 부응하여 LG전자의 가전 기기는 클라우드를 통해 연결된 사물인터넷을 지향
- LG전자의 모든 최신 가전 기기는 와이파이(Wi-Fi)를 탑재하여 출시되며 인터넷에 연결되어 클라우드를 통해 관리
- 삼성전자도 스마트홈 가전을 중심으로 클라우드를 사용



4차 산업혁명과 정보통신의 이해

감사합니다.
