

---

# AWS 101

---

## 아마존 웹서비스(AWS)와 함께 하는 클라우드 비용 최적화 전략

윤석찬

AWS코리아, 테크에반젤리스트

# 목차

1. 클라우드 컴퓨팅이란?
2. 아마존 웹 서비스(AWS)이란?
3. AWS 클라우드 비용 최적화 전략
4. AWS 비용 모니터링/분석 도구

# 클라우드 컴퓨팅 비지니스의 방식을 바꾸다!

단순한 기술 용어가 아니라 세상을 바꾸고 있습니다!



From 2014:

# Cloud computing has become the new normal

새로운 애플리케이션은  
클라우드 선택



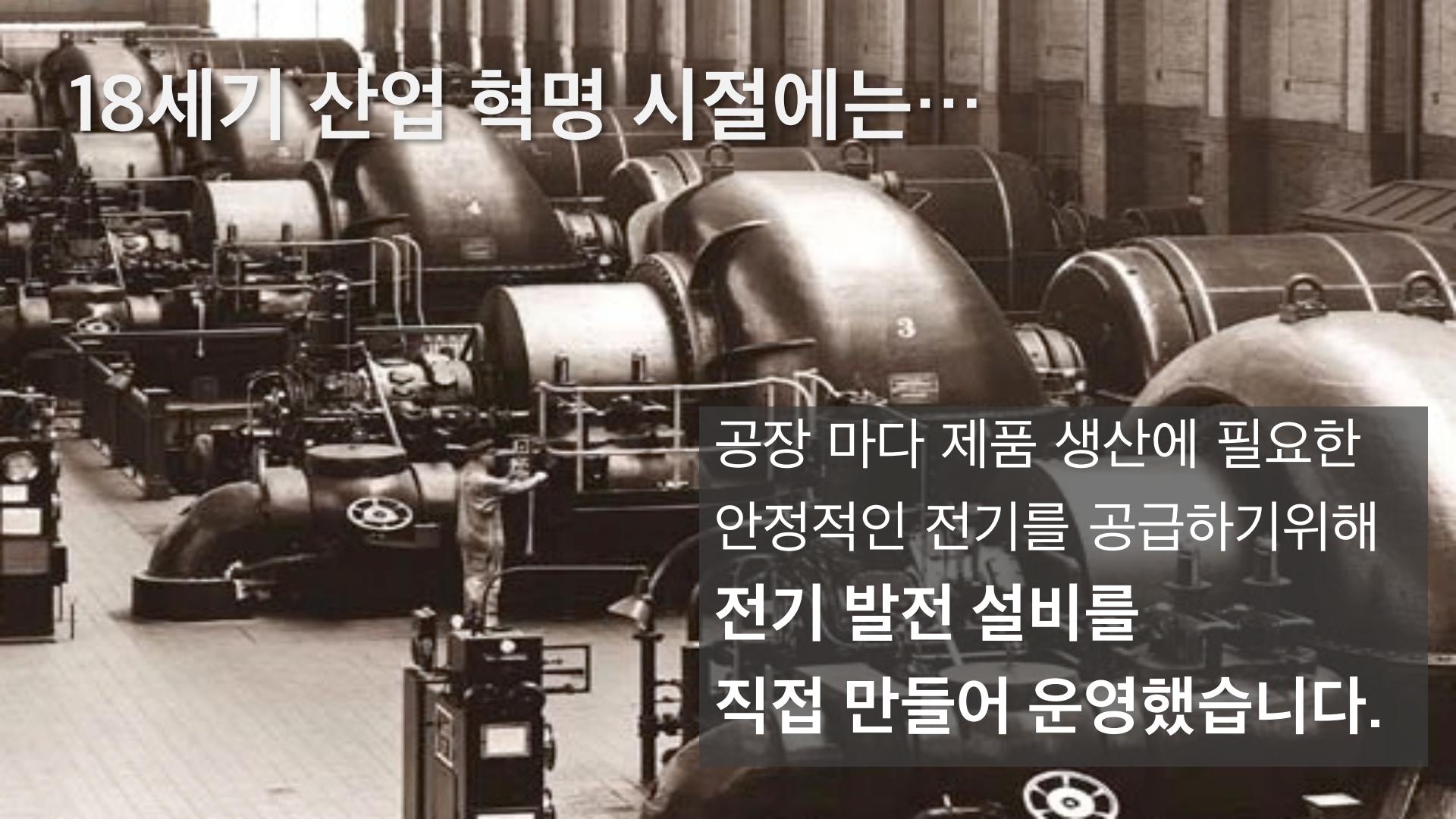
기존의 애플리케이션도  
빠르게 이전

FREEDOM  
AND  
CONTROL OVER YOUR OWN DESTINY

AWS 클라우드(Cloud)를 통하면 가능(enable)해집니다!

# 1. 클라우드 컴퓨팅이란 무엇인가요?

# 18세기 산업 혁명 시절에는...



공장마다 제품 생산에 필요한  
안정적인 전기를 공급하기 위해  
전기 발전 설비를  
직접 만들어 운영했습니다.

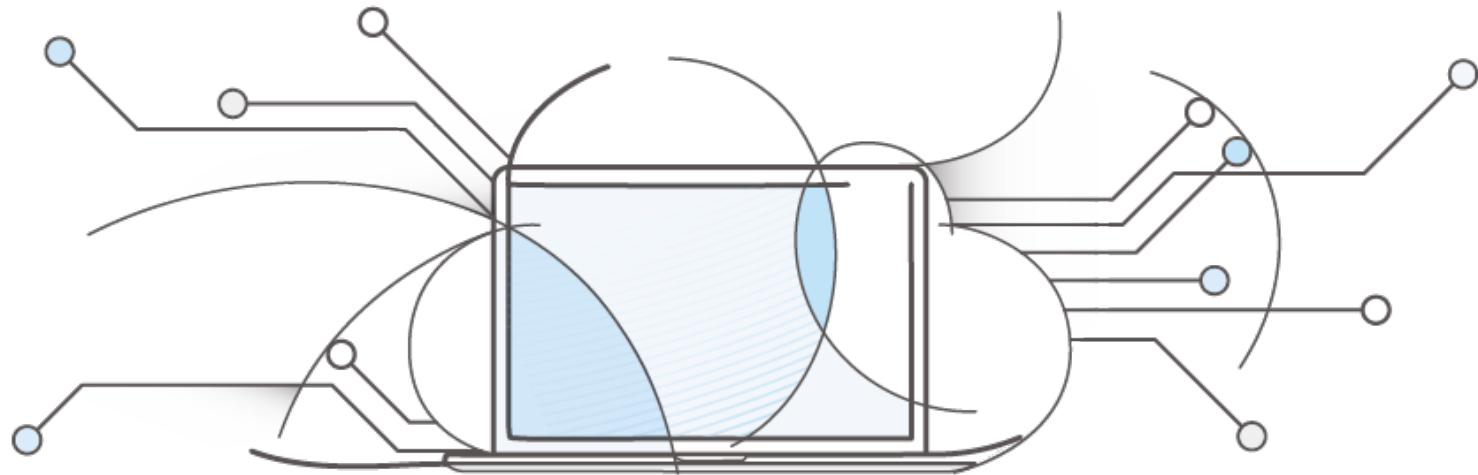
# 하지만, 21세기 현대에는...



누가 어떻게 만드는지 고려없이  
플러그를 꼽기만 하면  
언제 어디서나 쓴 만큼만  
요금을 내고 사용합니다.

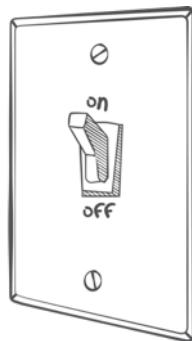
# 클라우드 컴퓨팅이란?

“인터넷을 통해 (물리적 서버와 네트워크와 같은) IT 리소스와 (데이터 분석과 같은) 애플리케이션을 원할 때 언제든지(On-demand) 사용한 만큼만 요금을 내는 서비스입니다”

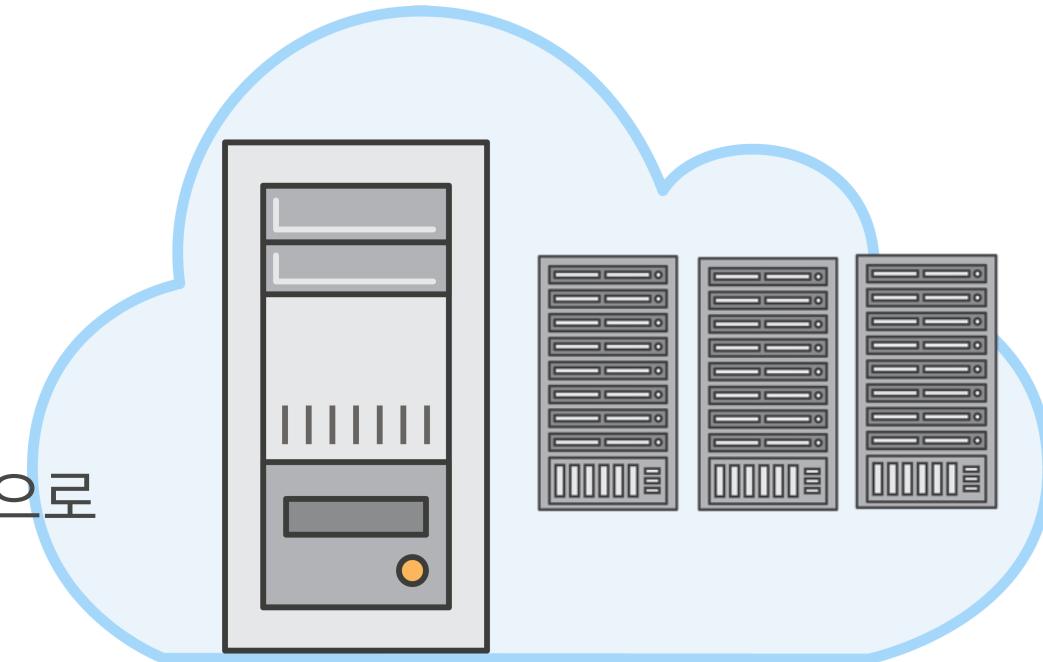


<http://aws.amazon.com/ko/what-is-cloud-computing/>

# 이제 여러분은…



몇 번의 클릭과 간단한 명령만으로  
**IT 자원을 언제 어디서나  
바로 사용할 수 있습니다!**

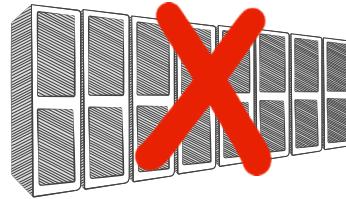


CPU, 메모리, 스토리지, 네트워크, 데이터베이스...

# 클라우드 컴퓨팅의 이점

## 1. 초기 선 투자 불필요

서비스 규모를 예측하고 미리 서버를 구매하고 관리할 필요가 없습니다.



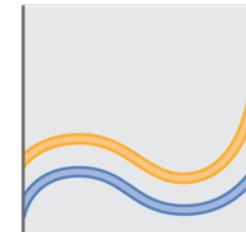
## 2. 저렴한 종량제 가격

규모의 경제를 통한 비용 절감에 따른 지속적 가격인하 및 종량제를 통한 효율적 과금 체계입니다.



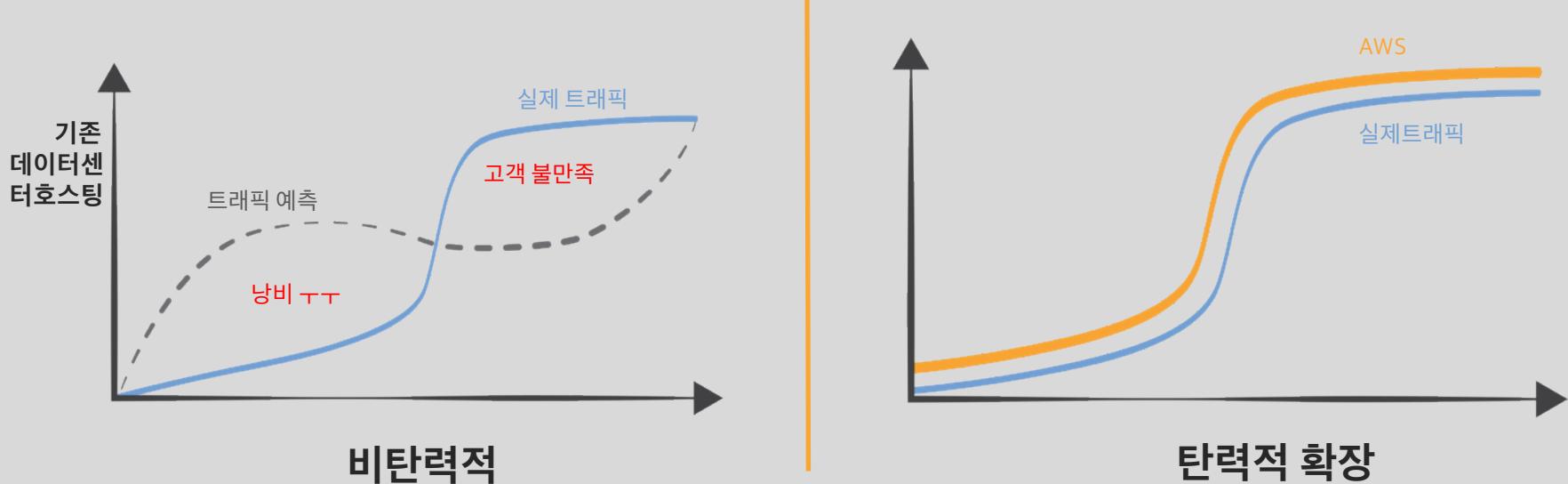
## 3. 탄력적인 운영 및 확장 가능

필요한 용량을 예측할 필요없이 트래픽 만큼만 사용하거나 손쉽게 확장할 수 있습니다.



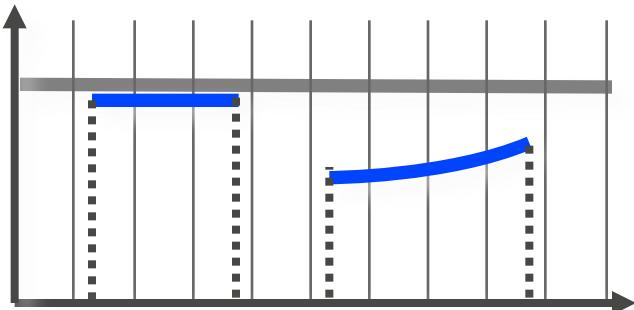


# 클라우드에서 확장성이란?

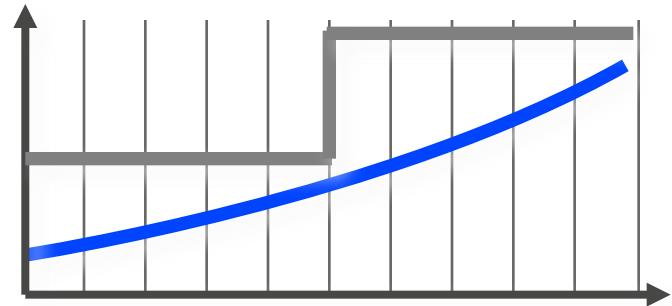


AWS에서는 프로그램 코드로 필요한 자원을 자동 증설 및 감소 할 수 있어  
자원을 손쉽게 활용할 뿐만 아니라 비용 효율적입니다.

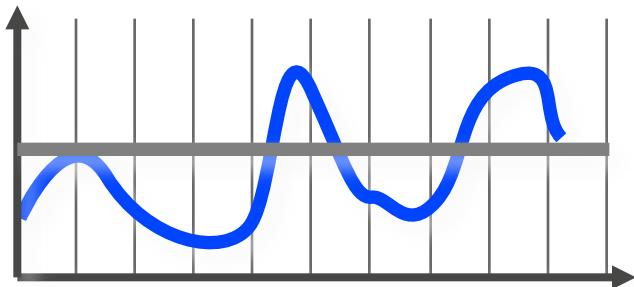
# 기존 데이터 센터 사용 패턴



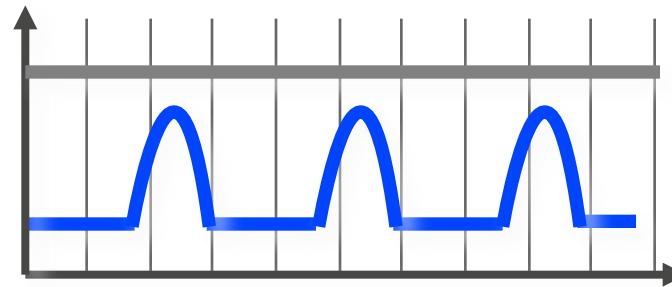
On and Off



Fast Growth

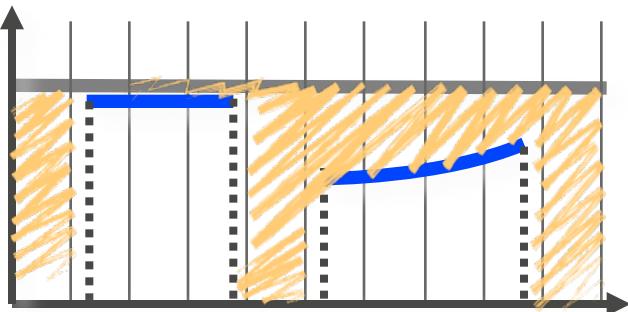


Variable peaks

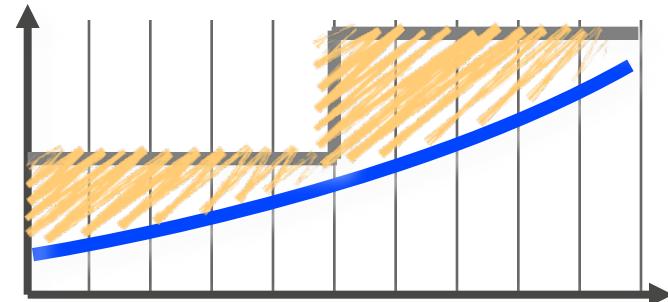


Predictable peaks

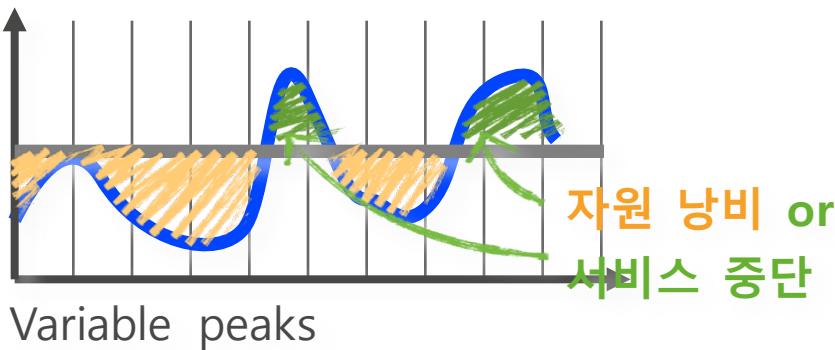
# 기존 데이터 센터 사용 패턴



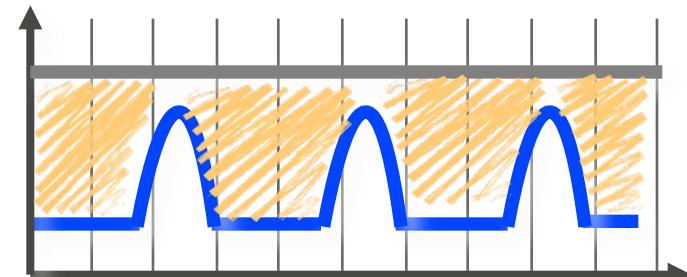
On and Off



Fast Growth

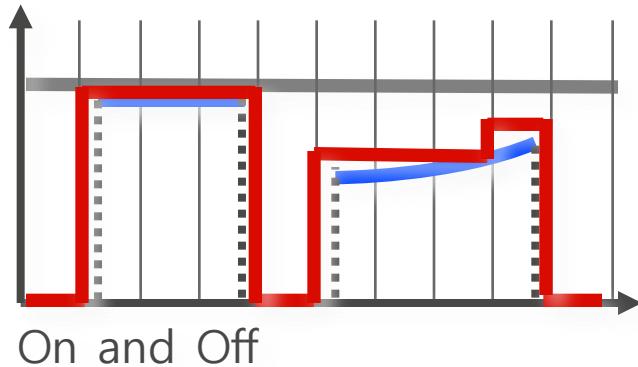


Variable peaks

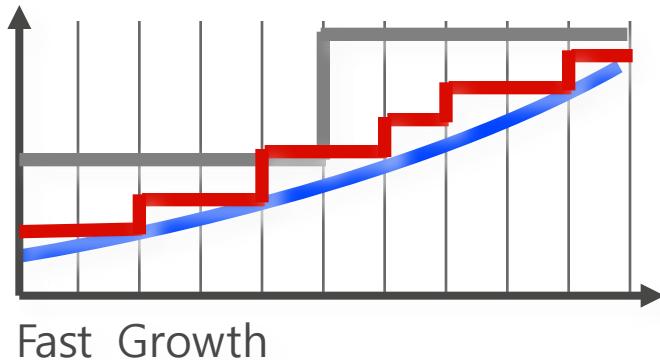


Predictable peaks

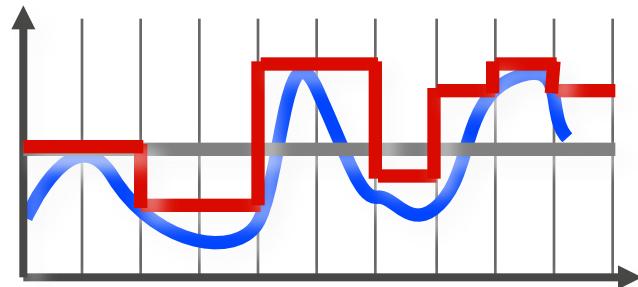
# AWS 클라우드 사용 패턴 (오토 스케일링)



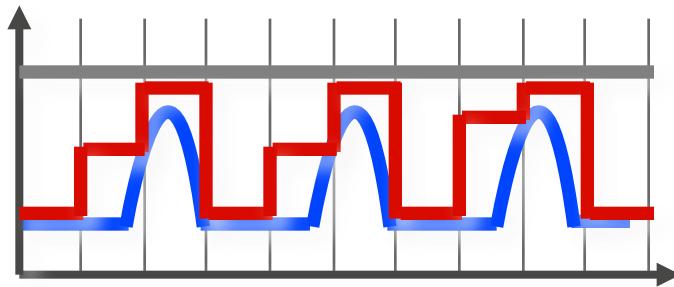
On and Off



Fast Growth



Variable peaks



Predictable peaks

# 클라우드 컴퓨팅의 이점

## 4. 속도와 민첩성

시장 상황에 빠르게 대응할 수 있는 민첩성을 통해 비지니스를 혁신할 수 있습니다.



## 5. 비지니스에만 집중 가능

인프라가 아니라 비즈니스를 차별화할 프로젝트와 고객에 집중할 수 있습니다.



## 6. 손 쉬운 글로벌 진출

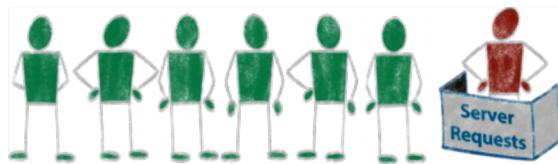
빠른 시간내에 손쉽게 글로벌 고객을 위한 서비스를 시작할 수 있습니다.



# 비지니스 민첩성- Business Agility

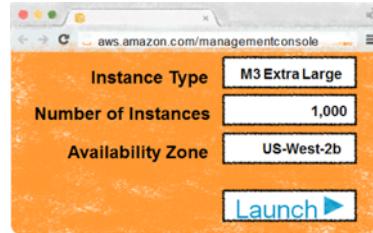
기존 방식:

인프라 준비를 위해 수 **주일** 소요



AWS:

인프라 준비를 위해 수 **분** 소요



새로운 개발 환경 추가

새로운 서비스 인프라 추가

유럽과 남미에 새로운 인프라 구성

1,000대의 서버 구성

1,000대의 서버 제거

1 PB Data Warehouse 구축

1 PB Data Warehouse 폐기

필요한 만큼 자원을 단 몇 분만에 준비 가능합니다!

# 성공을 위한 다양한 실험 가능



## On-Premises

혁신을 위한 시도가  
자주 일어나지 않고

실패의 비용이 높음

혁신 속도가 느려짐



혁신을 위한 시도를 많이  
할 수 있고

실패의 비용이 낮음

많은 혁신이 가능





# 클라우드를 통한 민첩성으로 시장 파괴

AWS 클라우드를 사용하는 스타트업의 약진!

Hotels



Storage



Music



Media



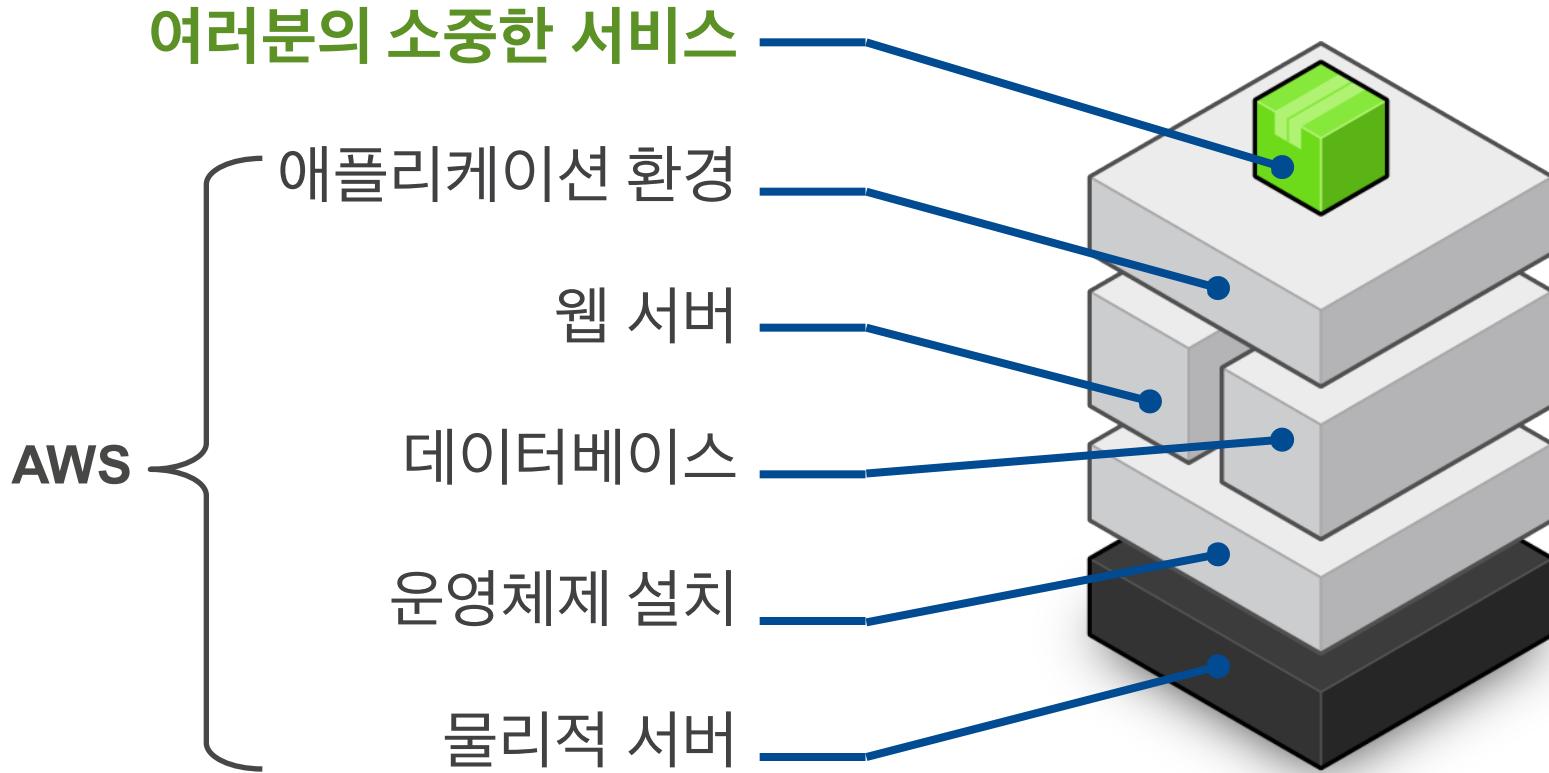
Game

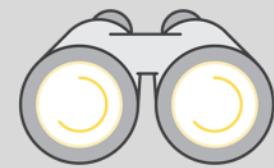


## 핵심 역량에만 집중하여

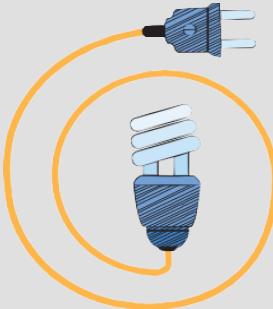
비지니스 변화에 빠르게 대처하고 있습니다!

# 우리가 필요한 것만 집중합시다!





# 클라우드 컴퓨팅이란?



필요한 만큼만  
원할때 마다



인터넷으로  
IT 자원을



사용한 만큼만  
지불하는



대용량  
서비스

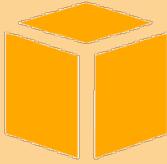
## 2. 아마존 웹 서비스란?

# 왜 Amazon이 클라우드를?

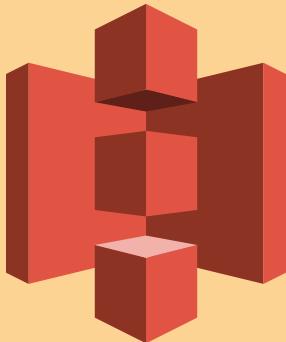


Amazon.com이라는 확장성 높은 글로벌 온라인 전자 상거래 서비스를 제공하는데 대용량 인프라 서비스 기술이 핵심 역량이라고 판단하고,

**2006년 Amazon Web Service(AWS)를 시작하였습니다.**



# Amazon Simple Storage Services (S3)



- 객체 기반의 무제한 파일 저장 스토리지
- 99.99999999% 내구성
- URL을 통해 손쉽게 파일 공유 가능
- 정적 웹사이트 호스팅 서비스 가능
- 사용(저장)한 만큼만 지불

# Amazon S3 성장세

**2,000,000,000,000** – S3 전체 파일 갯수 (2013년 4월 시점)

**400,000,000,000** – 우리 은하 내 별의 갯수

**13,500,000,000** – 중국 인구수

**매년 102% 데이터 전송량 증가**

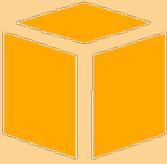
(Q3 2014 vs Q3 2015, not including Amazon use)

2012

2013

2014





# Amazon Elastic Compute Cloud (EC2)

- 가상 서버(Virtual Server) 서비스로서 다양한 운영 체제 선택 가능
- Amazon Machine Image(AMI)를 통한 손쉬운 구성 가능
- 키페어를 통한 보안 접속 및 VPC를 통한 사설 서버망 구성 가능
- 다양한 인스턴스 타입 제공



# Amazon EC2 성장세

인스턴스 사용량 88% YoY

(Q3 2014 vs Q3 2015)



2010

2011

2012

2013

2014

2015



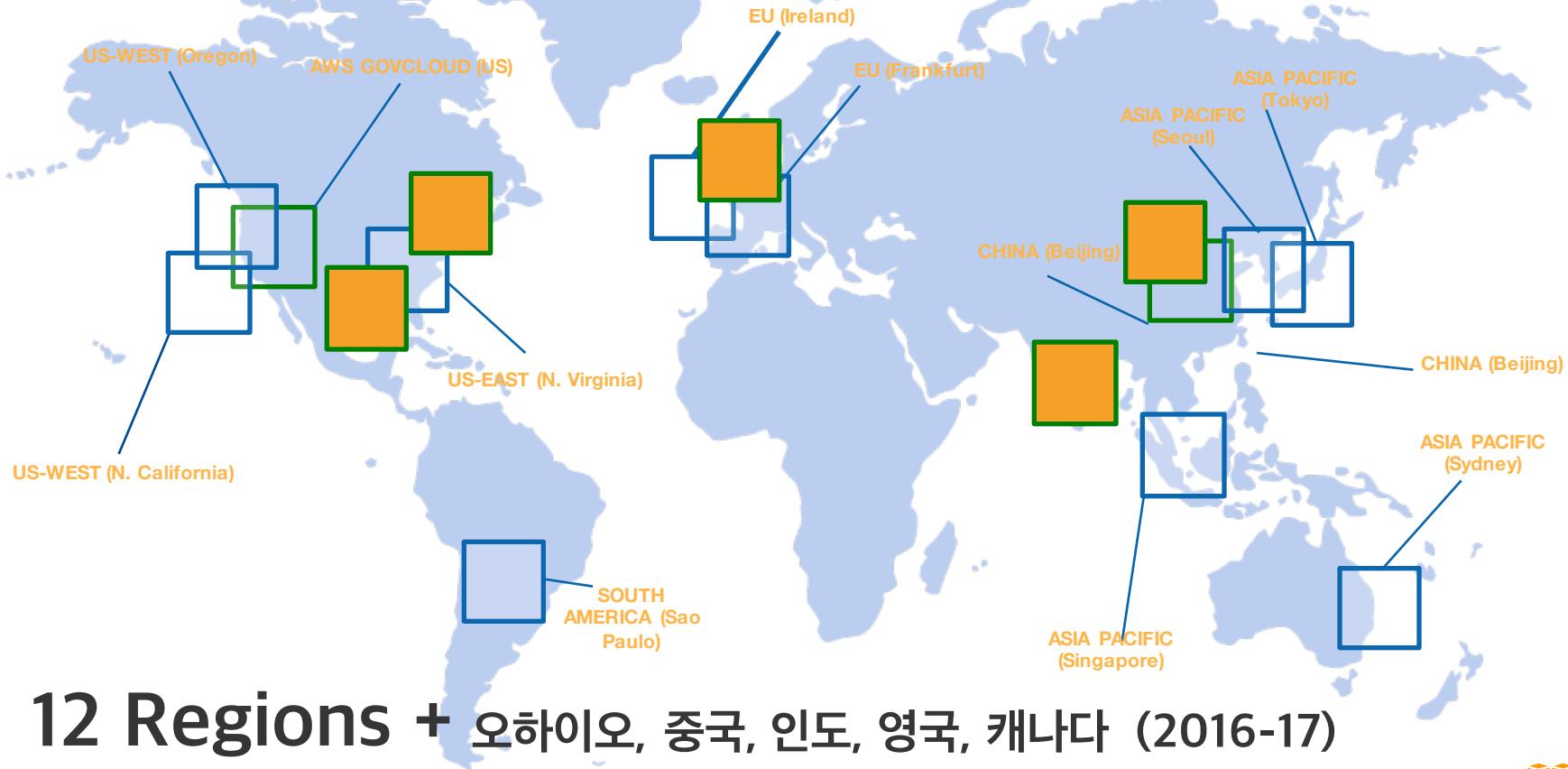
# AWS 글로벌 인프라



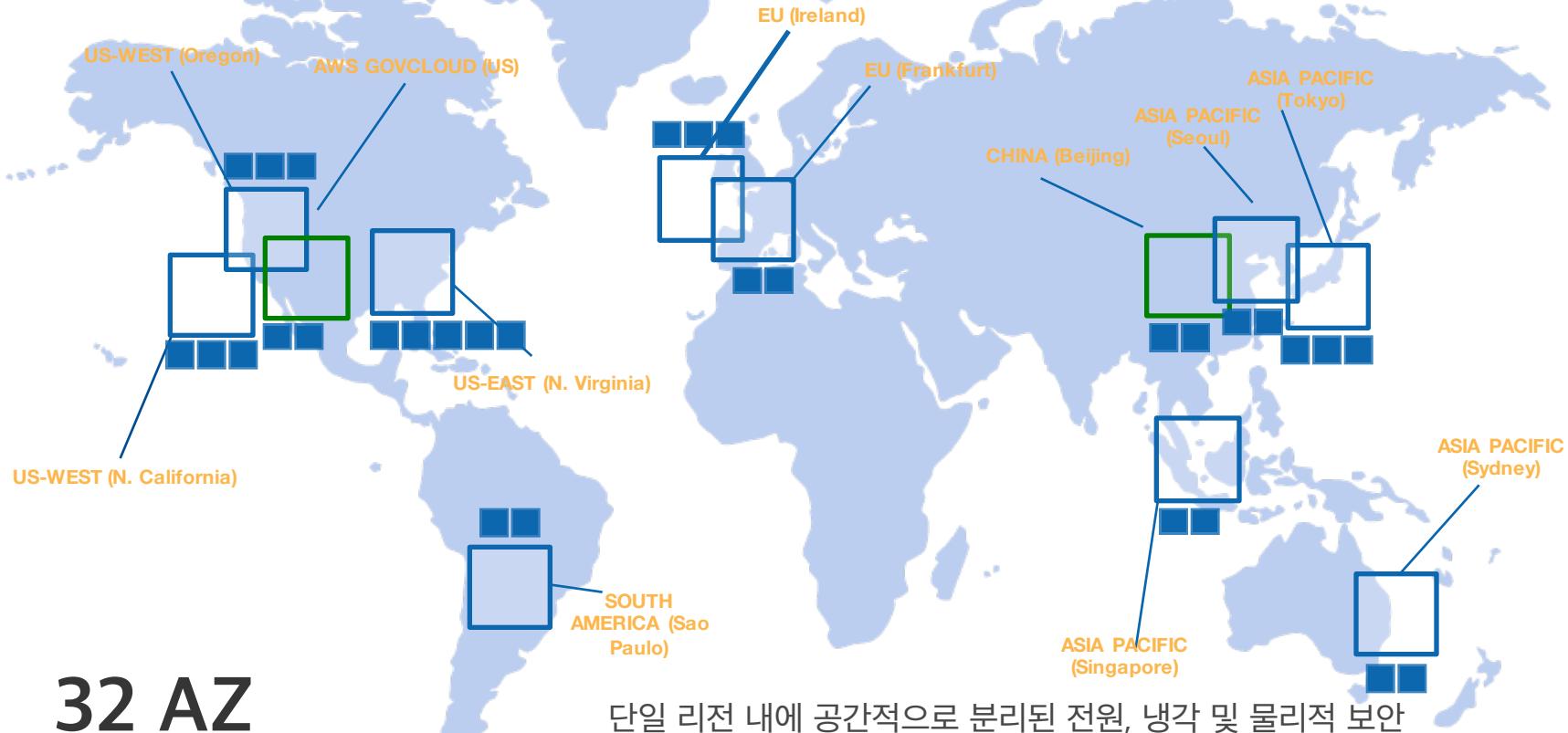
## 12 Regions

복수개의 데이터센터 클러스터의 독립적인 물리적 위치로서 대부분 AWS 자원은 리전 단위로 제공됩니다.  
각 리전은 개별 가용영역(AZ)로 구성되어 있습니다.

# AWS 글로벌 인프라



# AWS 글로벌 인프라



**32 AZ**  
Availability Zones

단일 리전 내에 공간적으로 분리된 전원, 냉각 및 물리적 보안 시설을 가지고 운영되는 데이터센터로서 개별 AZ 사이에는 낮은 지연 속도를 가진 서버 클러스터로 연결되어 있습니다.

# AWS 글로벌 인프라



콘텐츠 혹은 정적 파일을 더 빠르게 전달하기 위한 배포 서비스 인프라로서 개별 지역에 사용자에게 더 빠르게 파일을 배포할 수 있어 서비스 속도를 높혀줍니다.

# API를 이용한 신속하고 효율적 자원 생성 및 운영

```
ec2-run-instances ami-b232d0db  
--instance-count 3  
--availability-zone  
ap-northeast-2a  
--instance-type m1.small
```

```
ec2-run-instances ami-b232d0db  
--instance-count 3  
--availability-zone  
ap-northeast-2b  
--instance-type m1.small
```

서울 리전 내 2개의 가용영역  
(독립 데이터센터)에 각각  
3개의 m1.small  
가상 서버를 띄워라!

```
--instance-filters  
--filter-name=  
image-  
owner-  
aws-marketplace  
--  
--  
--  
--
```



# AWS 클라우드 서비스 확대



# AWS의 빠른 기술 혁신 속도

2006년 이후 **1,900** 여회가 넘는 신규 기능 출시

2014년 대비 **40%** 성장

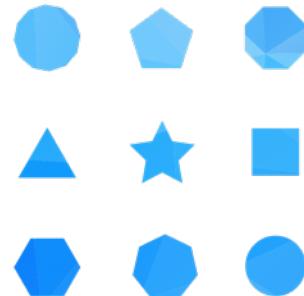


\*2015년 12월 31일 현재

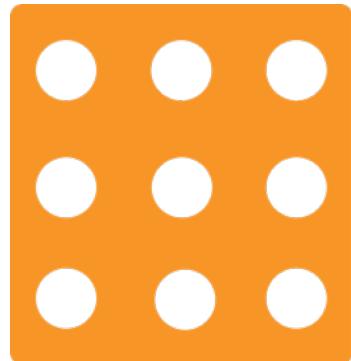


## 현업에서 필요한 다양한 업무 요구에 맞는 유연성 제공

다양한 고객의  
요구사항과  
워크로드가 존재

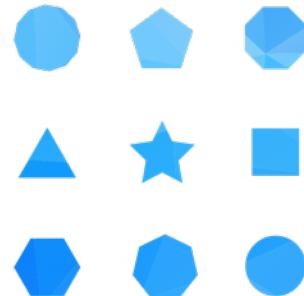


낮은 수준의 표준에  
맞출 필요가 없음



## 현업에서 필요한 다양한 업무 요구에 맞는 유연성 제공

다양한 고객의  
요구사항과  
워크로드가 존재

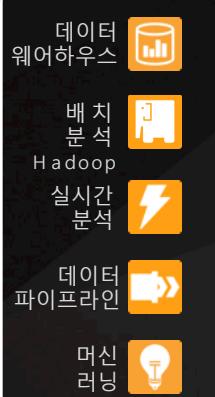


클라우드에서 유연한  
기능 및 서비스 선택



# AWS의 다양하고 폭넓은 서비스 분야

## 데이터 분석



## 모바일



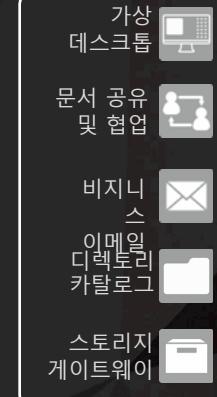
## 개발 배포



## 애플리케이션 서비스



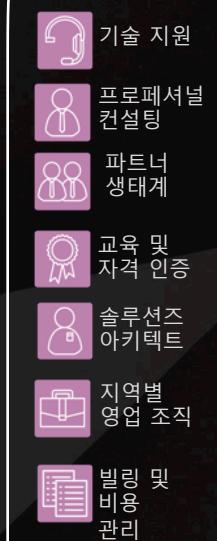
## 엔터프라이즈



## 하이브리드



## 기술/영업 지원



사설 네트워크(VPC)



인증 및 접근 제어



암호화 및 키 관리



자동화 인프라 관리



모니터링 로그/감사



데디케이트 서비스



보안/관리

컴퓨팅



스토리지  
(캐시/볼륨/백업)



데이터베이스  
(RDS/NoSQL/캐싱)



네트워킹



핵심 서비스

리전



가용영역



콘텐츠 배포



글로벌 인프라

# Amazon Web Services

## Compute

- EC2 Virtual Servers in the Cloud
- EC2 Container Service Run and Manage Docker Containers
- Elastic Beanstalk Run and Manage Web Apps
- Lambda Run Code in Response to Events

## Storage & Content Delivery

- S3 Scalable Storage in the Cloud
- CloudFront Global Content Delivery Network
- Elastic File System PREVIEW Fully Managed File System for EC2
- Glacier Archive Storage in the Cloud
- Import/Export Snowball Large Scale Data Transport
- Storage Gateway Integrates On-Premises IT Environments with Cloud Storage

## Database

- RDS Managed Relational Database Service
- DynamoDB Predictable and Scalable NoSQL Data Store
- ElastiCache In-Memory Cache
- Redshift Managed Petabyte-Scale Data Warehouse Service

## Networking

- VPC Isolated Cloud Resources
- Direct Connect Dedicated Network Connection to AWS
- Route 53 Scalable DNS and Domain Name Registration

## Developer Tools

- CodeCommit Store Code in Private Git Repositories
- CodeDeploy Automate Code Deployments
- CodePipeline Release Software using Continuous Delivery

## Management Tools

- CloudWatch Monitor Resources and Applications
- CloudFormation Create and Manage Resources with Templates
- CloudTrail Track User Activity and API Usage
- Config Track Resource Inventory and Changes
- OpsWorks Automate Operations with Chef
- Service Catalog Create and Use Standardized Products
- Trusted Advisor Optimize Performance and Security

## Security & Identity

- Identity & Access Management Manage User Access and Encryption Keys
- Directory Service Host and Manage Active Directory
- Inspector PREVIEW Analyze Application Security
- WAF Filter Malicious Web Traffic

## Analytics

- EMR Managed Hadoop Framework
- Data Pipeline Orchestration for Data-Driven Workflows
- Elasticsearch Service Run and Scale Elasticsearch Clusters
- Kinesis Work with Real-time Streaming data

## Internet of Things

- AWS IoT BETA Connect Devices to the cloud

## Mobile Services

- Mobile Hub BETA Build, Test, and Monitor Mobile apps
- Cognito User Identity and App Data Synchronization
- Device Farm Test Android, Fire OS, and iOS apps on real devices in the Cloud
- Mobile Analytics Collect, View and Export App Analytics
- SNS Push Notification Service

## Application Services

- API Gateway Build, Deploy and Manage APIs
- AppStream Low Latency Application Streaming
- CloudSearch Managed Search Service
- Elastic Transcoder Easy-to-use Scalable Media Transcoding
- SES Email Sending Service
- SQS Message Queue Service
- SWF Workflow Service for Coordinating Application Components

## Enterprise Applications

- WorkSpaces Desktops in the Cloud
- WorkDocs Secure Enterprise Storage and Sharing Service
- WorkMail PREVIEW Secure Email and Calendaring Service

US East (N. Virginia)

US West (Oregon)

US West (N. California)

EU (Ireland)

EU (Frankfurt)

Asia Pacific (Singapore)

Asia Pacific (Tokyo)

Asia Pacific (Sydney)

**Asia Pacific (Seoul)**

South America (São Paulo)

Explore the next generation of AWS cloud capabilities. See what's new

## Service Health

All services operating normally.

Updated: Nov 10 2015 14:17:01 GMT+0900

[Service Health Dashboard](#)

arn more

of  
re tags.  
,

**Editor**

N our  
S.

# AWS 활용 = Building Block 조립



비지니스 요구 사항에 맞는  
50여개 이상의 서비스 조립을 통해 유연한 활용 가능



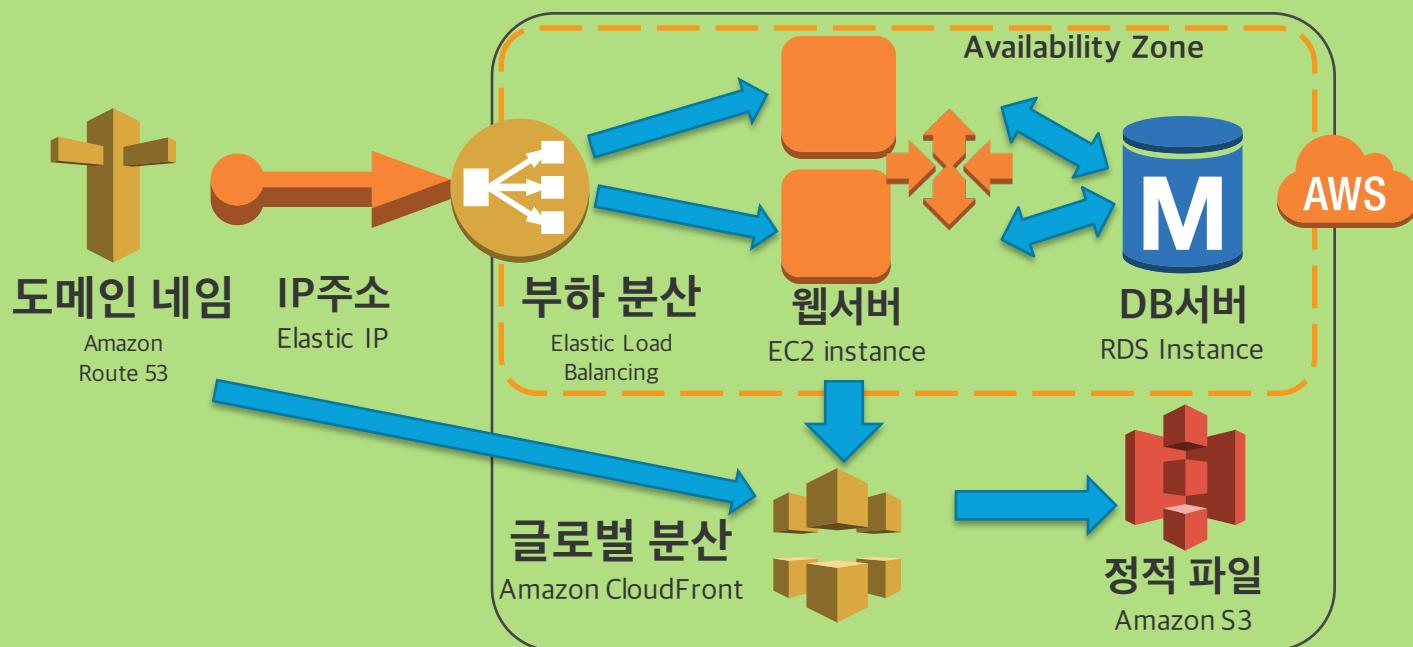
# AWS 클라우드 아키텍쳐

사용자 요청

도메인

애플리케이션

데이터



# 폭넓으면서도… 깊이 있는 서비스 기능

## 컴퓨팅

General Purpose (M4)  
Compute Optimized (C4)  
Memory Optimized (R3)  
GPU Optimized (G2)  
Storage Optimized (D2)  
IO Optimized (I2)  
Low Cost, Burst-able Performance (T2)  
Dedicated Instances

## 데이터베이스

RDS For MySQL  
RDS For SQL Server  
RDS For Oracle  
RDS For PostgreSQL  
RDS For Amazon Aurora  
Multi-AZ Synchronous Replication  
Read Replica Support

## 암호화

Server-side Object Encryption  
Customer Controlled Keys  
Dedicated HSMs  
Integrated Key Management  
Key Usage Auditing

## 접근 제어

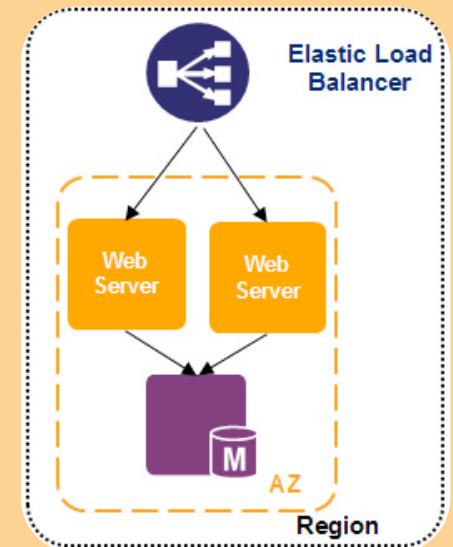
Identity Policies  
Location Policies  
Time-based Policies  
Individual API Calls  
Key Rotation  
Temporary Credentials  
Policy Simulator





# Amazon Relational Database Service (RDS)

- 관계형 데이터베이스(RDB) 관리형 서비스
- 제공 가능 데이터베이스
  - Aurora, MySQL, Microsoft SQL, Oracle, PostgreSQL, MariaDB
- 업데이트, 보안 패치, 백업 지원  
Multi-AZ 배포 옵션 등



ORACLE®

Amazon Aurora

PostgreSQL



Microsoft®  
SQL Server



INN

MariaDB

amazon  
web services

# Amazon Database 서비스 성장세

Increased 113% YoY

(Q3 2014 vs Q3 2015)

A \$1B Annual Revenue Run-rate Business



2012

2013

2014

2015



# AWS 보안 우선 순위 정책



FISMA

ISO

MPAA



“우리 경험에 따르면 AWS 클라우드 환경이 자체 데이터 센터 보다 보안이 더 강화되었다고 생각합니다.”

*“Based on our experience, I believe that we can be even more secure in the AWS cloud than in our own data centers.”*

-Tom Soderstrom, CTO, NASA JPL

<http://aws.amazon.com/ko/security> 참고

클라우드 보안 요구 사항에 적합한 아키텍쳐  
및 업무에 대한 보안 감사 및 인증 획득

# AWS 클라우드 플랫폼 생태계 확장



Partner  
Network

수 만개의 SI & Consultants 및  
ISV 파트너



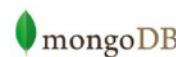
accenture  
High performance. Delivered.



Infosys® Cloud



35개의 제품 분류에 2,500개가  
넘는 파트너 제품이 있으며,  
고객이 원클릭으로 설치 가능



MicroStrategy



전 세계 190여개국

# 1백만 이상의 고객

Over 1 Million Active Customers



"Active customer" is defined as a non-Amazon customer with AWS account usage activity in the past month, including the free tier



# Gartner Magic Quadrant for Cloud Infrastructure as a Service, Worldwide

Figure 1. Magic Quadrant for Cloud Infrastructure as a Service, Worldwide



Source: Gartner (May 2015)

Gartner "Magic Quadrant for Cloud Infrastructure as a Service, Worldwide." Lydia Leong, Douglas Toombs, Bob Gill, May 18, 2015. This Magic Quadrant graphic was published by Gartner, Inc. as part of a larger research note and should be evaluated in the context of the entire report. The Gartner report is available at <http://aws.amazon.com/resources/analyst-reports/>. Gartner does not endorse any vendor, product or service depicted in its research publications, and does not advise technology users to select only those vendors with the highest ratings or other designation. Gartner research publications consist of the opinions of Gartner's research organization and should not be construed as statements of fact. Gartner disclaims all warranties, expressed or implied, with respect to this research, including any warranties of merchantability or fitness for a particular purpose.

# AWS 국내 고객 현황

스타트업



게임 개발사



엔터프라이즈



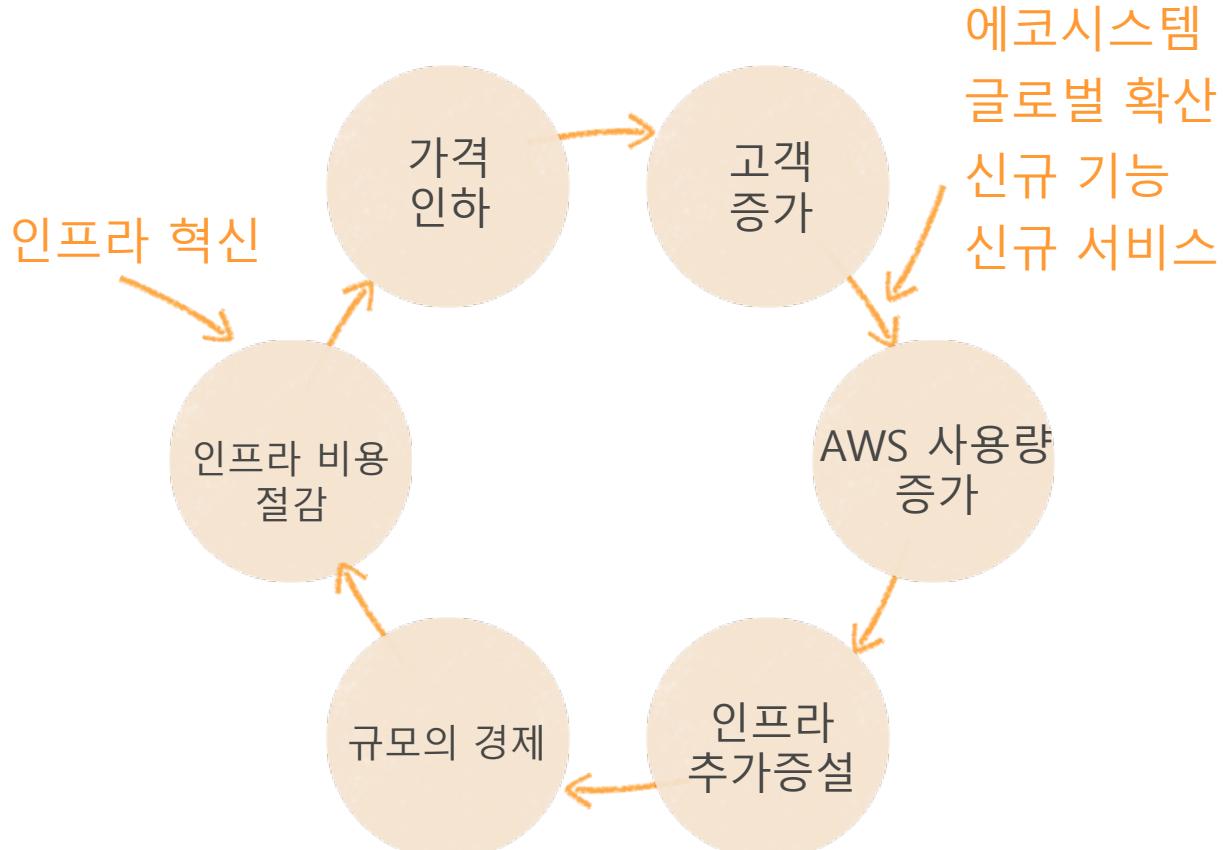
### 3. 클라우드 비용 최적화 전략

비용 최적화 전략 #1

아무것도 안 하기



# AWS 가격 정책 철학



51

## PRICE REDUCTIONS

규모의 경제와 대용량 저이익  
구조에 최적화하여 비용  
절감에 따른 이익을 통해  
2006년 이후 자체적으로  
가격인하



비용 최적화 전략 #2

거의 아무것도 안 하기

# AWS Trusted Advisor 활용하기

The screenshot shows the AWS Trusted Advisor interface under the 'Cost Optimization' section. It displays a summary icon with 2 green checkmarks, 5 orange triangles, and 0 blue exclamation marks, indicating potential monthly savings of \$337.01. Below this, a list of six cost optimization checks is shown:

- Amazon EC2 Reserved Instances Optimization**: Checks your Amazon Elastic Compute Cloud (Amazon EC2) computing consumption history and calculates an optimal number of Heavy Utilization Reserved Instances. Monthly savings of up to \$212.64 (82.6%) might be available with optimal Reserved Instance use.
- Low Utilization Amazon EC2 Instances**: Checks the Amazon Elastic Compute Cloud (Amazon EC2) instances that were running at any time during the last 14 days and alerts you if the daily GPU utilization was 10% or less and network I/O was 5 MB or less on 4 or more days. 4 of 8 Amazon EC2 instances have low average daily utilization. Monthly savings of up to \$59.25 might be available by minimizing underutilized instances.
- Idle Load Balancers**: Checks your Elastic Load Balancing configuration for load balancers that are not actively used. 1 of 1 load balancers appear to be idle. Monthly savings of up to \$20.16 are available by minimizing unused load balancers.
- Underutilized Amazon EBS Volumes**: Checks Amazon Elastic Block Store (Amazon EBS) volume configurations and warns when volumes appear to be underused. 5 of 13 EBS volumes appear to be underutilized. Monthly savings of up to \$44.96 are available by minimizing underused EBS volumes.
- Unassociated Elastic IP Addresses**: Checks for Elastic IP addresses (EIPs) that are not associated with a running Amazon Elastic Compute Cloud (Amazon EC2) instance. 4 of 6 Elastic IP addresses are not associated with a running instance.



In 2014:  
2,613,585 recommendations  
\$350+ in cost reductions

비즈니스, 엔터프라이즈 서포트 고객에게 33 가지 이상 체크리스트 제공  
(네 가지 체크리스트는 모든 고객이 이용가능)

<http://aws.amazon.com/blogs/aws/trusted-advisor-console-basic/>



# 모니터링을 통한 최적의 인스턴스 탑입 선택

## 1. 시작

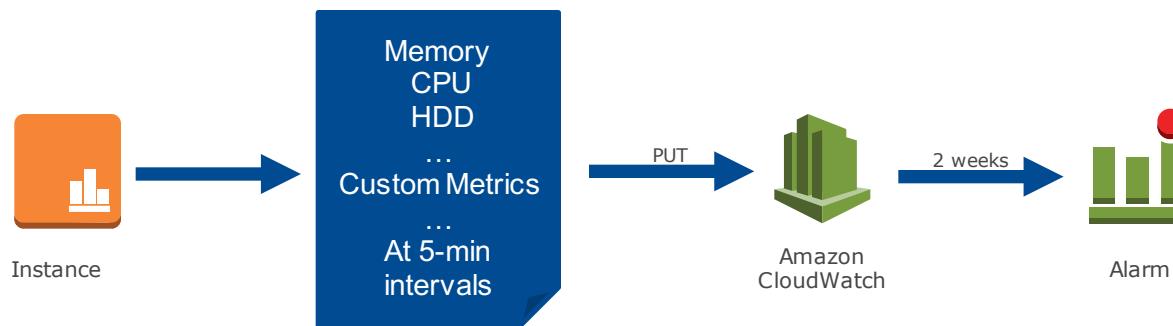
여러분의 애플리케이션의  
요구 사항에 맞는 메모리와  
CPU 등을 선택

## 2. 모니터링

모니터링을 통해 인스턴스  
타입을 변경 (CloudWatch  
활용)

## 3. Roll-Out

증설 시 인스턴스를  
추가로 넣어 구성





# Amazon CloudWatch



The screenshot shows the AWS EC2 Dashboard. On the left, there's a sidebar with links like EC2 Dashboard, Events, Tags, Reports, Limits, Instances, Spot Requests, Reserved Instances, AMIs, Bundle Tasks, Volumes, Snapshots, Security Groups, Elastic IPs, Placement Groups, Load Balancers, Key Pairs, and Network Interfaces. The main area has tabs for Launch Instance, Connect, Actions, and Monitoring. Under Monitoring, it shows CloudWatch alarms (1 of 1 in OK) and CloudWatch metrics (Detailed monitoring). Below that, two line charts are displayed: CPU Utilization (Percent) and Network In (Bytes). The CPU Utilization chart shows a fluctuating line between 0% and 50% over the dates 6/26 to 7/5. The Network In chart shows a single sharp peak reaching nearly 10,000,000 bytes around June 29.

## • 기본 모니터링

- EC2 7가지 통계
  - CPU utilization
  - Data transfer
  - Disk usage and more
- 5분 간격
- Amazon EBS, Amazon DynamoDB, Amazon RDS 도 지원

## • 세부 모니터링

- 1분 간격
- 인스턴스 태입 및 AMI 별로 통합 가능



# 다양한 EC2 인스턴스 선택 사항

범용



컴퓨팅  
최적화



메모리  
최적화



대용량  
스토리지



스토리지/  
IO 최적화



- 사이즈 선택: small/medium/large/xlarge/2xlarge
- 인스턴스 타입: c4.medium

# 인스턴스 타입별 요금 체계 차이

## ■ 비슷한 성능 다른 요금 체계

| Instance   | vCPU | Mem (GiB) | Monthly price (3-yr heavy RI ) | Ideal use case                 |
|------------|------|-----------|--------------------------------|--------------------------------|
| c3.2xlarge | 8    | 15        | \$121.22                       | Best price-compute performance |
| m3.2xlarge | 8    | 30        | \$161.15                       | Balanced                       |
| r3.2xlarge | 8    | 61        | \$189.66                       | Lowest cost per GiB RAM        |
| r3.xlarge  | 4    | 30.5      | \$94.83                        | Lowest cost per GiB RAM        |

## ■ T2 인스턴스 타입 - 버스팅에 알맞는 인스턴스 타입

| Instance  | vCPU | Mem (GiB) | Monthly price (3-yr heavy RI ) | Ideal use case             |
|-----------|------|-----------|--------------------------------|----------------------------|
| m3.medium | 1    | 3.75      | \$20.16                        | Always available, balanced |
| c3.large  | 2    | 3.75      | \$29.95                        | Always available, compute  |
| t2.medium | 2    | 4         | \$17.87                        | Burst workloads            |



# 인스턴스 타입 변경을 통한 비용 절감

- T2 인스턴스 타입이 잘 맞다면?
  - 11% 절감: m3.medium를 t2.medium으로 변경
  - 40% 절감: c3.large를 t2.medium으로 변경
- 워크로드에 맞는 인스턴스 타입을 선택한다면?
  - 25% 절감: CPU를 많이 쓰는 서비스 (m3.2xlarge -> c3.2xlarge)
  - 41% 절감: 메모리를 많이 쓰는 서비스 (m3.2xlarge -> r3.xlarge)

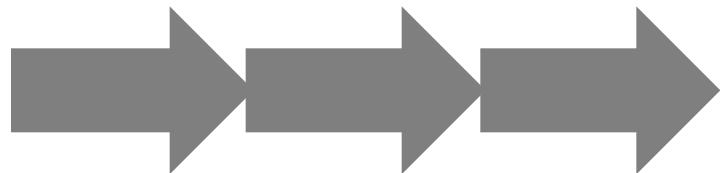
비용 최적화 전략 #3

아키텍쳐 최적화



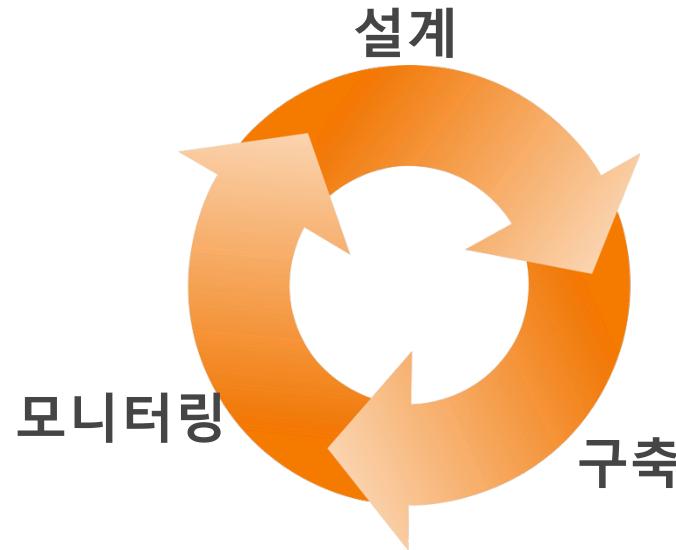
# 아키텍쳐 운영 프로세스의 변화

과거 - 온프레미스



Months

현재 -  amazon  
web services



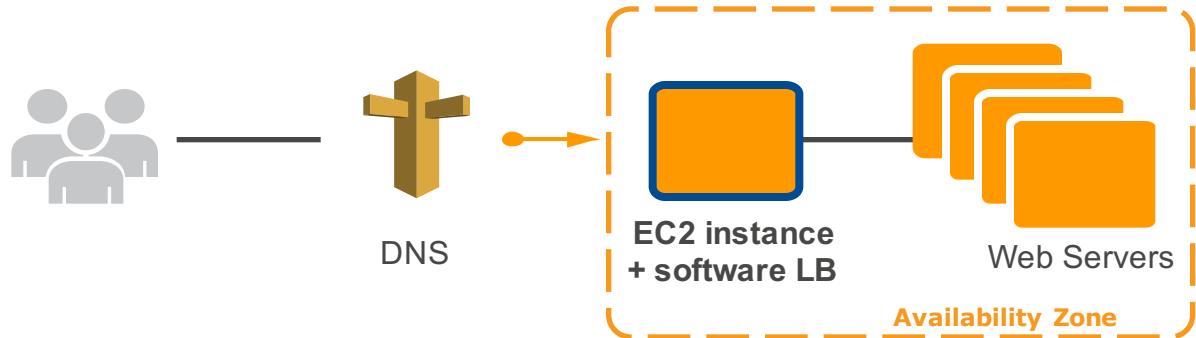
Minutes



# 로드밸런서 하나 바꿨을 뿐인데...

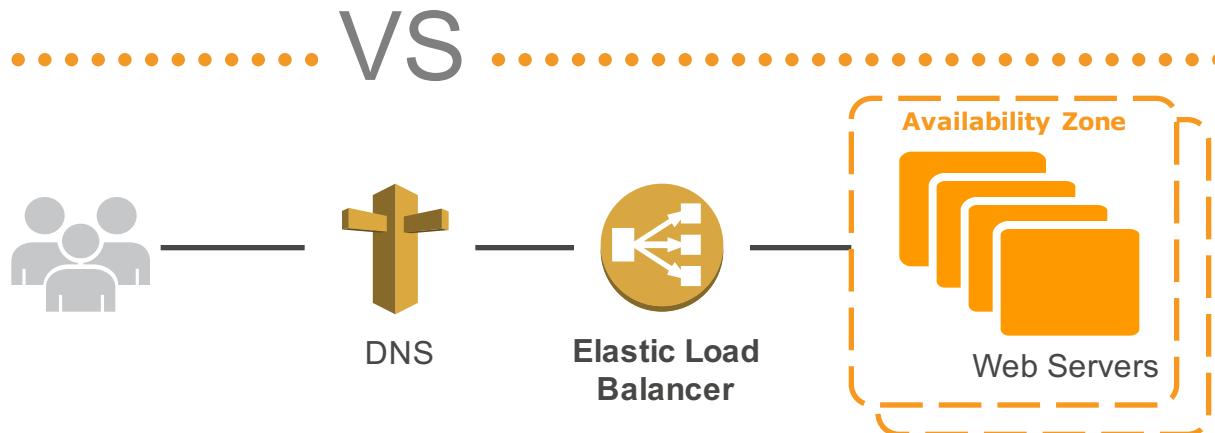
**\$0.080**

per hour  
(small instance)



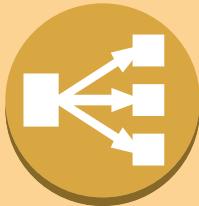
**\$0.028**

per hour

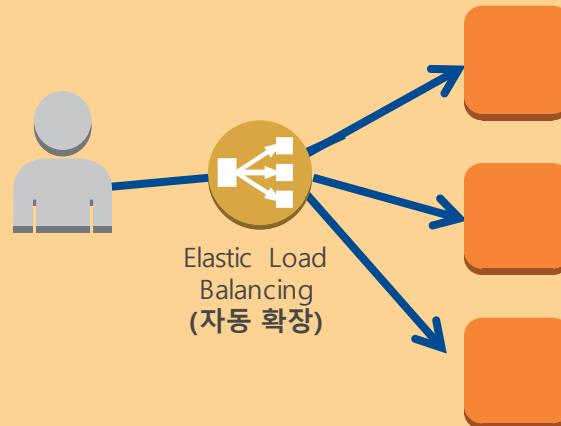




# Elastic Load Balancing(ELB)



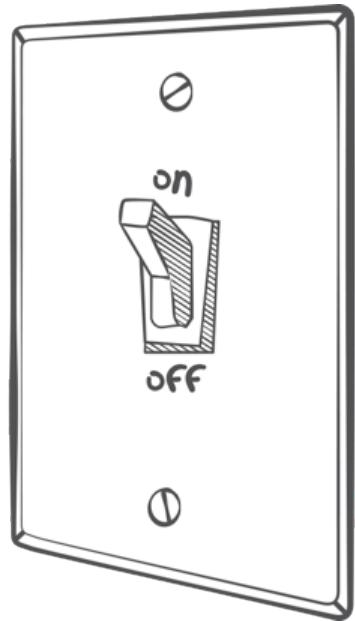
- 사용자 트래픽에 따라 가상 서버 사이의 부하를 분산해 주는 서비스
- 오토 스케일링 지원 및 스스로 자동으로 확장 가능
- 멀티 AZ 지원을 통한 가용성 확보



# 비용 최적화 전략 #3

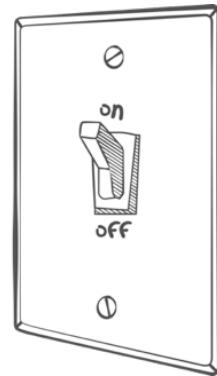
아키텍처 최적화를 위한  
7가지 단계

# 1. 사용 안 하는 인스턴스는 끄세요

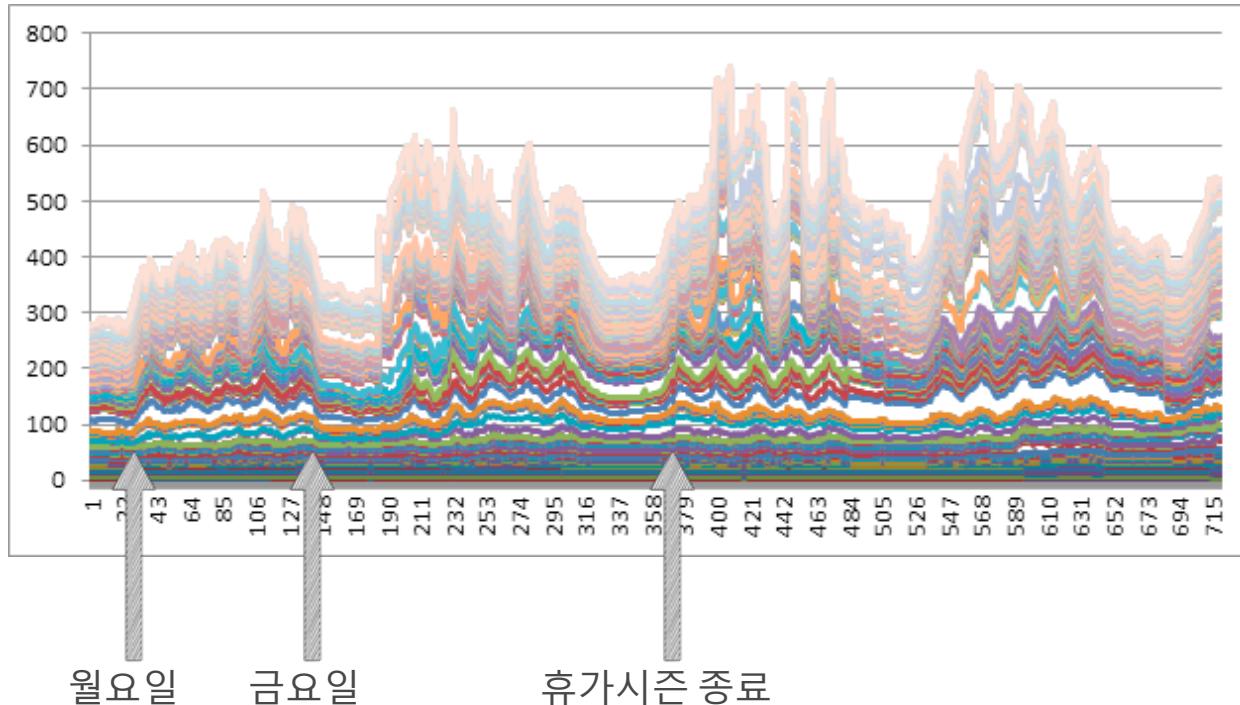


# 1. 사용 안 하는 인스턴스는 끄세요

- 개발자, 테스트, 교육용 인스턴스
  - 인스턴스 Start, Stop 기능 사용
- 자동화 및 스크립트 활용
  - 주말 / 퇴근 스크립트
- 인스턴스는 쓰고 버리는 물건이라는 인식

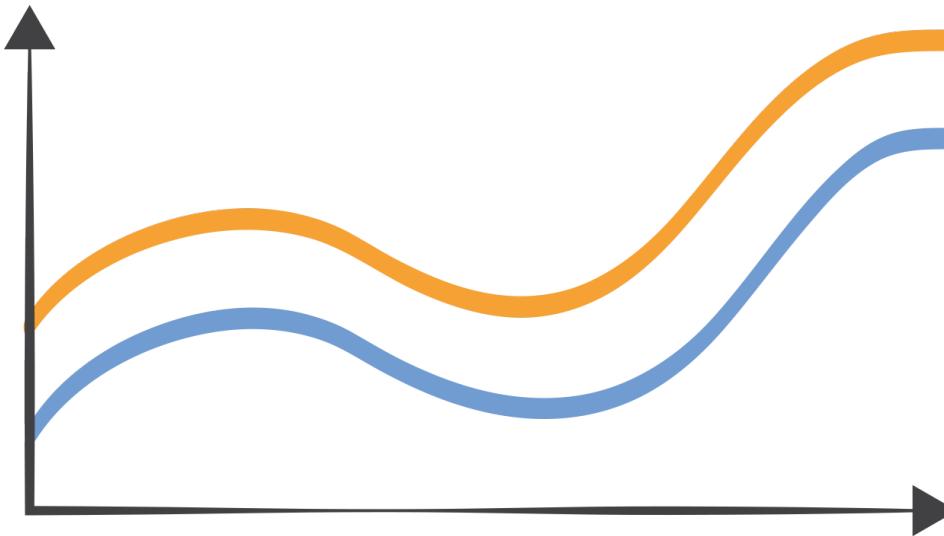


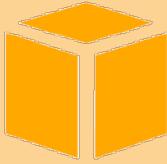
# 사례. 어떤 고객의 개발/테스트 서버



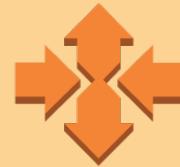
35% 절감

## 2. Auto Scaling을 사용하세요!

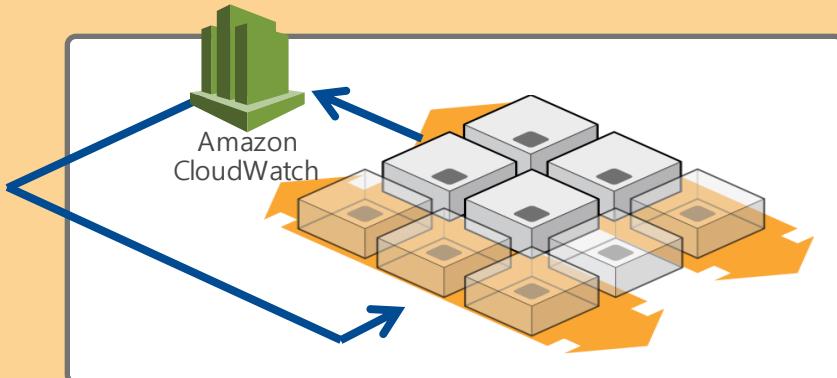




# 오토 스케일링 - Auto Scaling



- 사용자 트래픽(CPU 용량 혹은 데이터 전송량)에 따라 EC2 인스턴스 숫자를 자동으로 증가 혹은 감소
- 트래픽에 따라 사용된 인스턴스에 따라 시간단위로 과금하므로 비용 효율적



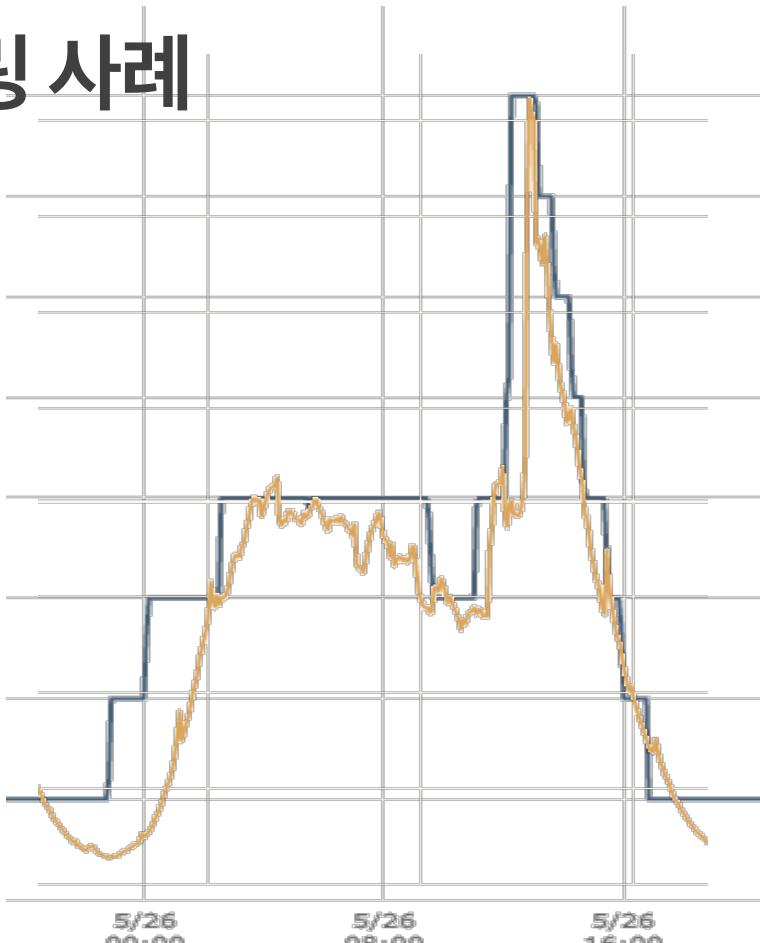
## AWS CLI

```
$ aws autoscaling create-auto-scaling-group  
--auto-scaling-group-name MyGroup  
--launch-configuration-name MyConfig  
--min-size 4  
--max-size 200  
--availability-zones ap-northeast-2a, ap-northeast-west-2b
```

# 데브시스터즈 오토 스케일링 사례



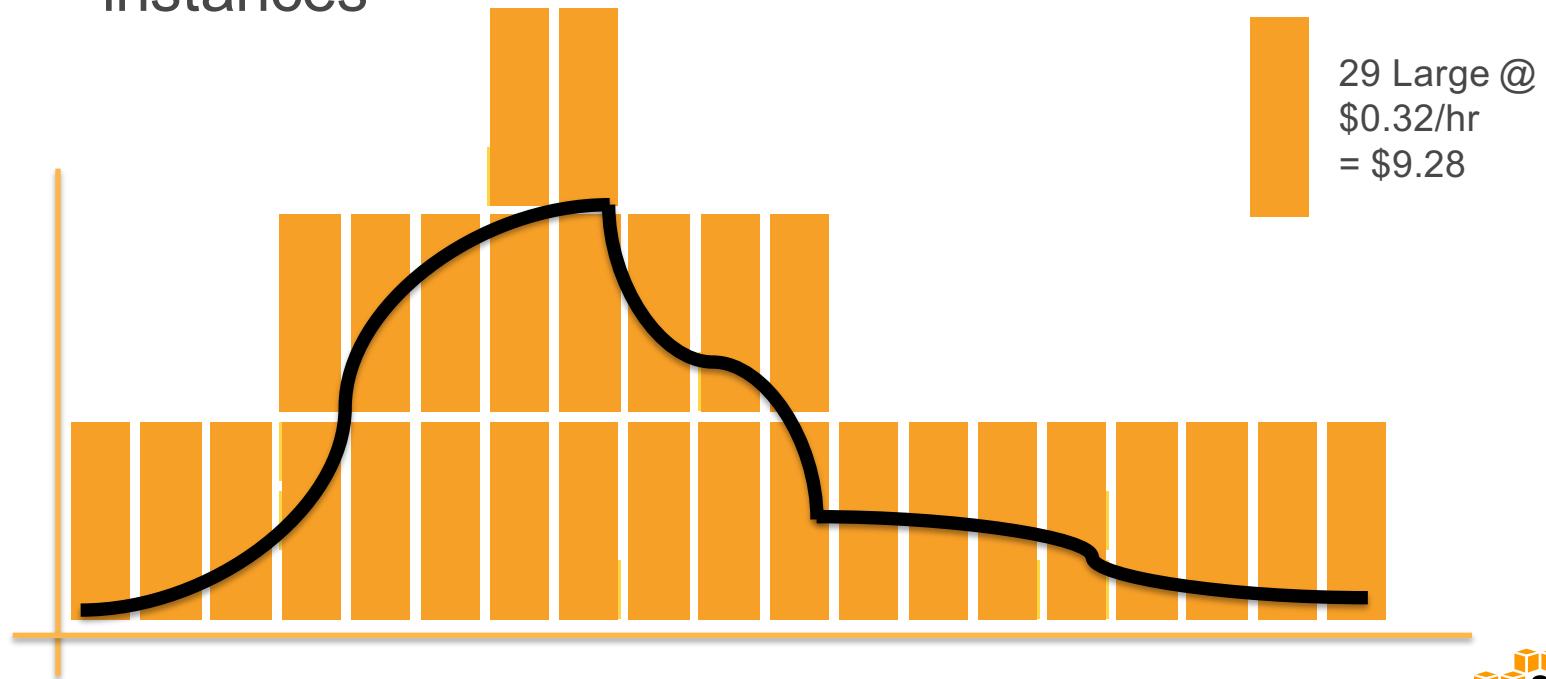
5월 어느 날..!



# instances vs. CPU utilization

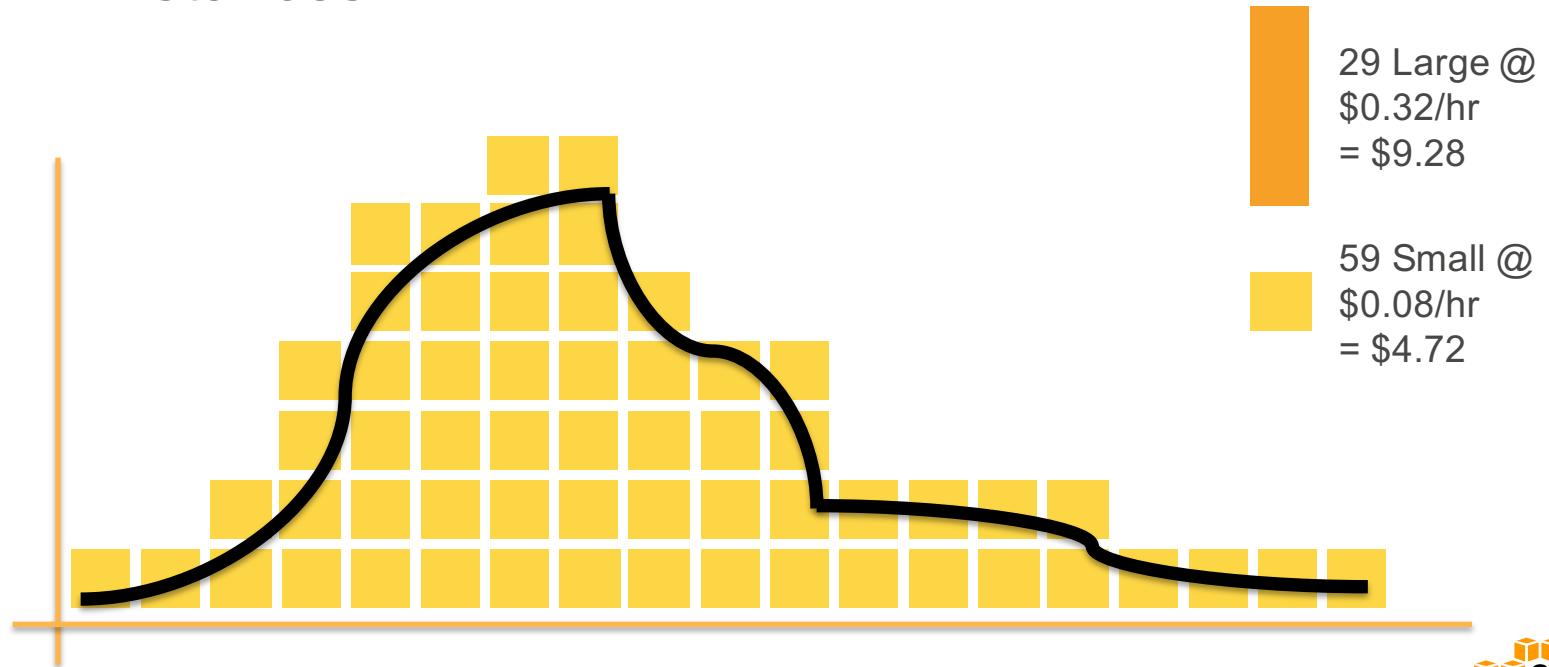
# 오토스케일링: 작은 인스턴스를 활용!

more small instances vs. less large instances



# 오토스케일링: 작은 인스턴스를 활용!

more small instances vs. less large instances



# 디스패치 오토 스케일링 사례

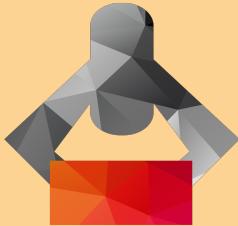


오토 스케일링을 쓰기에  
작은 워크로드가 나눠져 있다면?

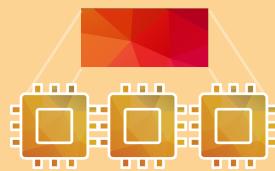


# Amazon EC2 Container Service

“Docker기반 확장성과 고성능 콘테이너 관리 서비스”



손쉬운 Docker  
콘테이너  
제작 및 배포



EC2 인스턴스를  
클러스터 구성 가능

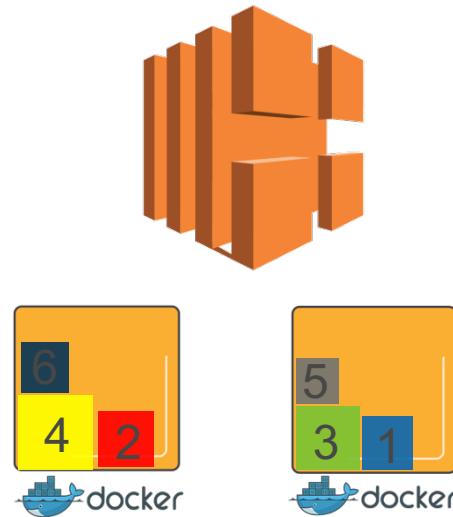
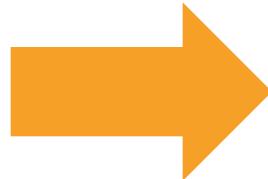
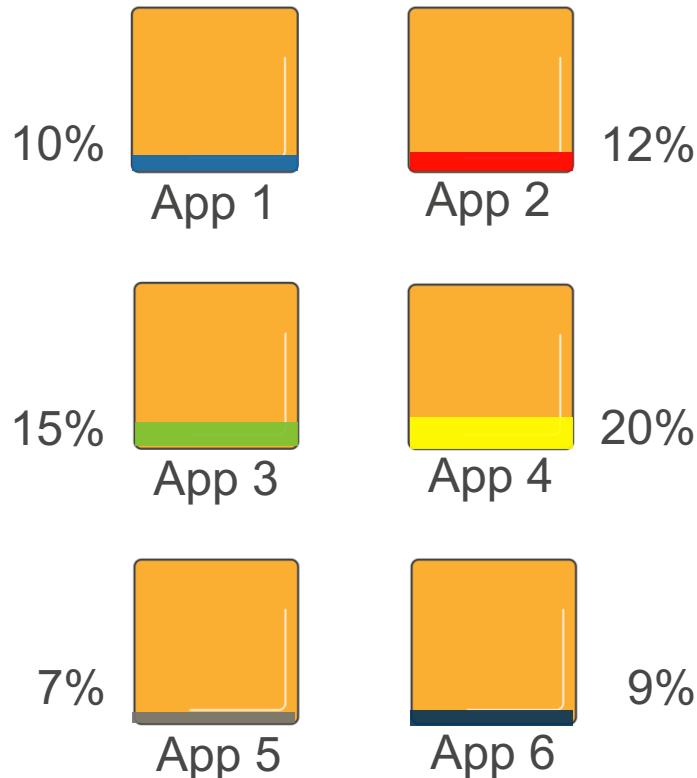


AWS 서비스와  
연계를 통한 확장성  
및 높은 성능



AWS 기반 높은  
보안성 제공

# 콘테이너를 통한 기존 EC2 인스턴스 효율화

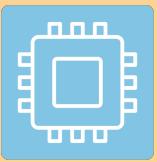


굳이 서버를 쓰기가 불필요한  
더 작은 워크로드가 나눠져 있다면?

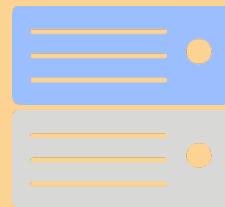


# AWS Lambda

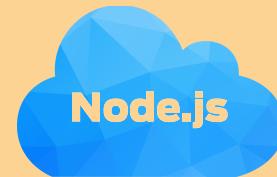
“이벤트 기반 확장성 높은 클라우드 코드 실행 서비스”



AWS 자원의 변경이 발생 시  
이벤트에 대한 변경 인식  
(API Gateway로 REST 실행도 가능)



서버 없이도  
고 확장성 기반 실행



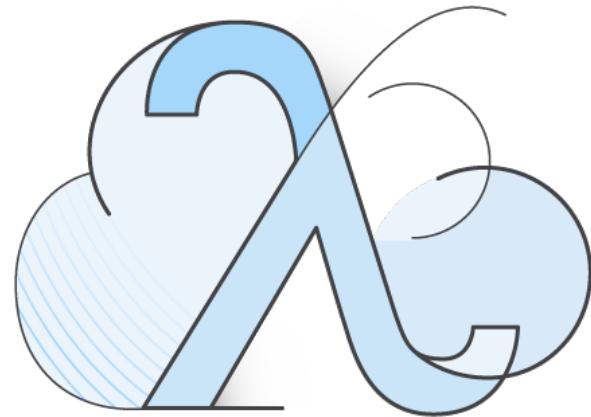
Node.js 및 자바, Python을  
지원하는 클라우드 함수

# AWS Lambda를 활용한 썸네일 생성하기

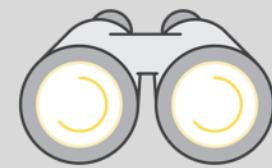


# AWS Lambda를 통한 서버 Idle time 없애기

- 자동 확장 및 자동 코드 배포 및 수행
- 인프라 관리 비용 불필요
- 백만 요청 당 \$0.20 (매월 백만 무료)
- 100 ms 단위 과금
- 수행한 시간 만큼만 과금

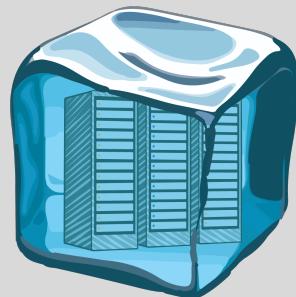


40% 이상 최적화가 필요하다면? Lambda를 고려해 보세요!



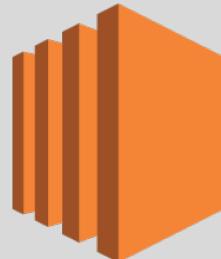
# 고객 요구에 따른 컴퓨팅 플랫폼 변화

Weeks



On-Premises

Minutes



Amazon EC2

Seconds



Amazon EC2 Containers  
Services

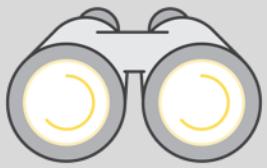


Milliseconds



AWS Lambda





# 고객 요구에 따른 클라우드 구매 옵션

## Free Tier

초기 비용없이 주요 서비스를 일정 기간 활용할 수 있음

첫 테스트에 용이



## On-Demand

약정없이 쓴 만큼만 지불

갑작스런 트래픽 예측하기 어려운 경우 신규 서비스



## Reserved

1년 혹은 3년 약정 40~70% 할인

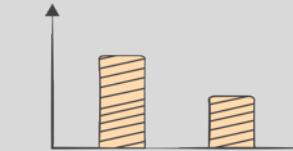
항상 사용 중인 안정화된 서버 자원을 위한 요금제



## Spot

남은 자원에 대한 경매 방식 80~90% 저렴

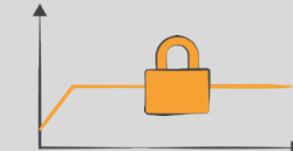
단기적으로 수요가 많을 때 유리



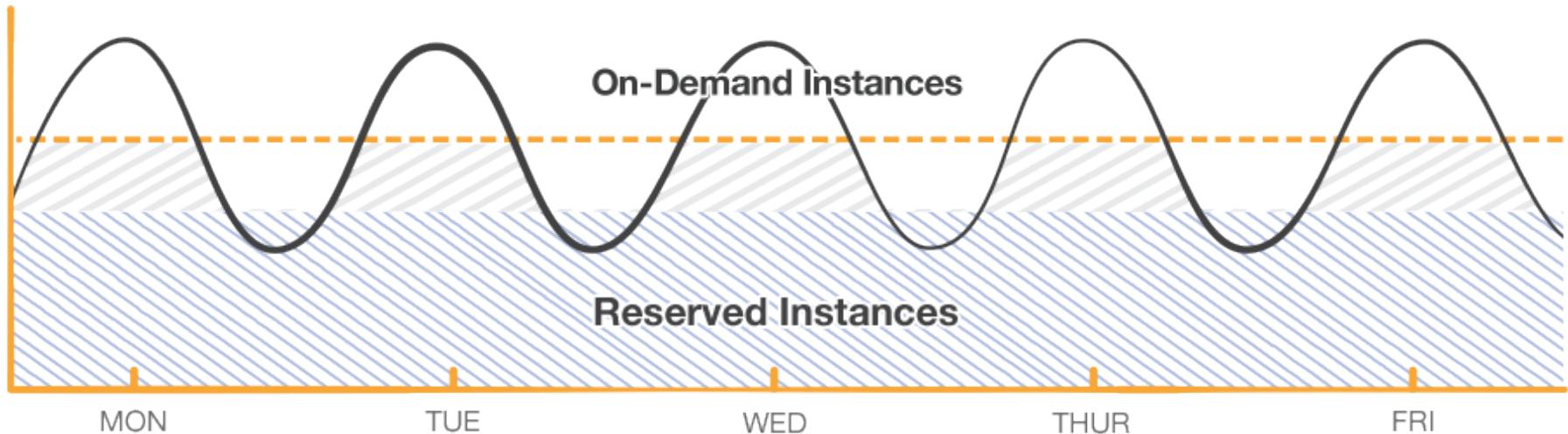
## Dedicated

VPC내에 한 고객만을 위한 물리적 서버에 런칭

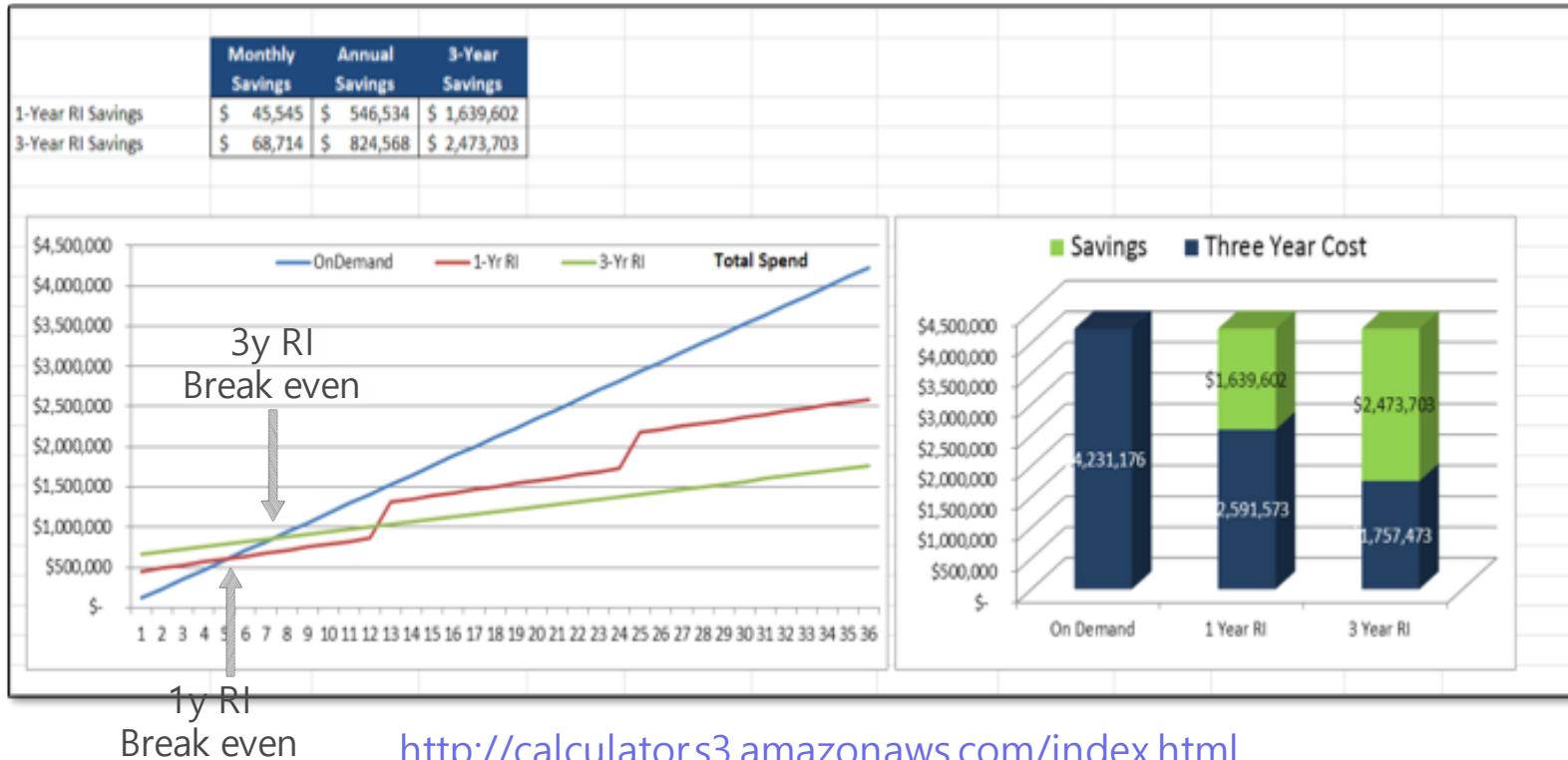
민감한 정보, 회사내규, 법적인 문제 가 있을경우



### 3. 예약 인스턴스(RI)를 쓰세요!



# 예약 인스턴스(RI) 가격 절감 계산



# 예약 인스턴스(RI) 가격 모델

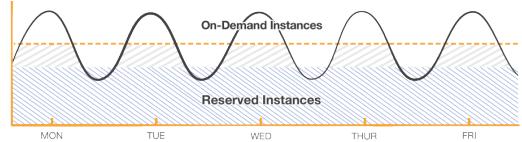
- **All Upfront(모두 선불)**- 3년 또는 1년 RI의 기간 이용 금액을 모두 일괄로 사전에 지불합니다. 가장 저렴합니다.
- **Partial Upfront(일부 선불)**- 이용액의 일부를 사전에 지불하고 3년 또는 1년 RI의 기간 동안 매월 지불하는 옵션입니다. All Upfront와 No Upfront의 중간 할인율입니다.
- **No Upfront(선불 없음)**- 사전 요금은 없습니다. 다만, RI의 기간 동안 지불 계약을 하며 할인율은 온 디멘드에 비해 30%가량 싸집니다.

<https://aws.amazon.com/ko/blogs/korea/simplified-reserved-instances/>



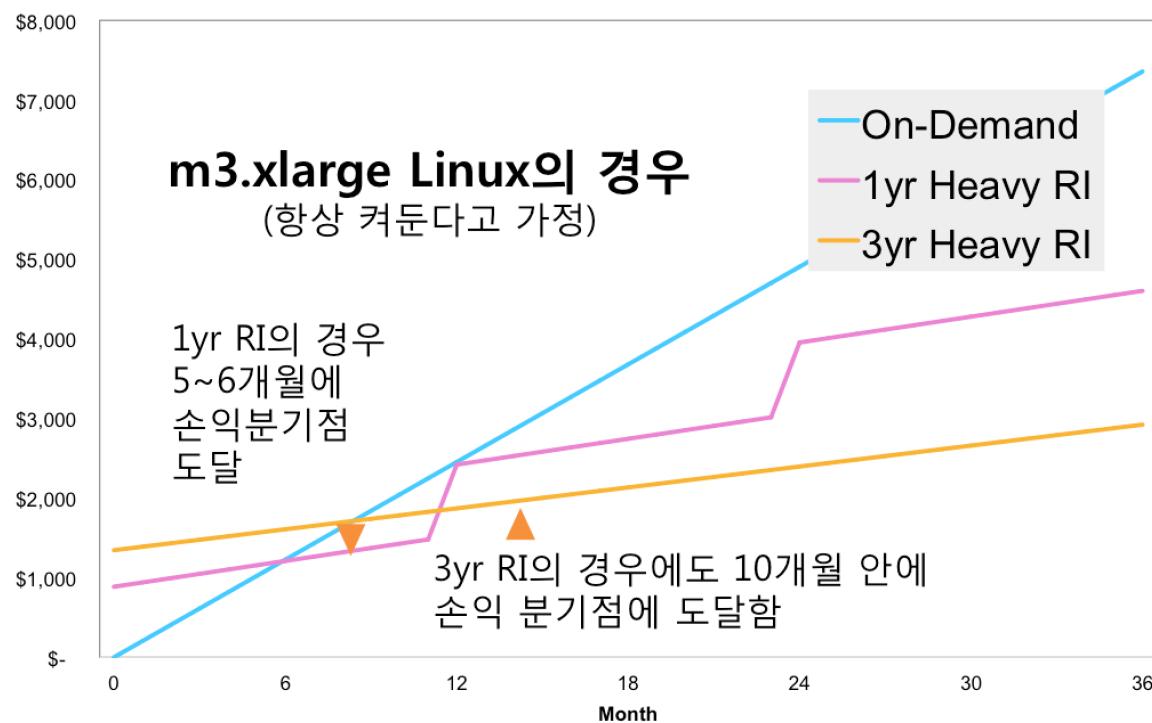
# 유연한 RI 활용 방법

- EC2 예약 인스턴스 RI 특징
  - 가용 영역(AZ) 사이에 이동 가능
  - EC2-Classic / EC2-VPC 이동 가능
  - 동일 인스턴스 패밀리 안에서 크기 변경 가능
- 다양한 서비스에 RI 모델 존재
  - Amazon RDS - 데이터베이스
  - Amazon Redshift - 데이터웨어 하우스
  - Amazon EMR - Hadoop 클러스터



| Instance Size | Normalization Factor |
|---------------|----------------------|
| small         | 1                    |
| medium        | 2                    |
| large         | 4                    |
| xlarge        | 8                    |
| 2xlarge       | 16                   |
| 4xlarge       | 32                   |
| 8xlarge       | 64                   |

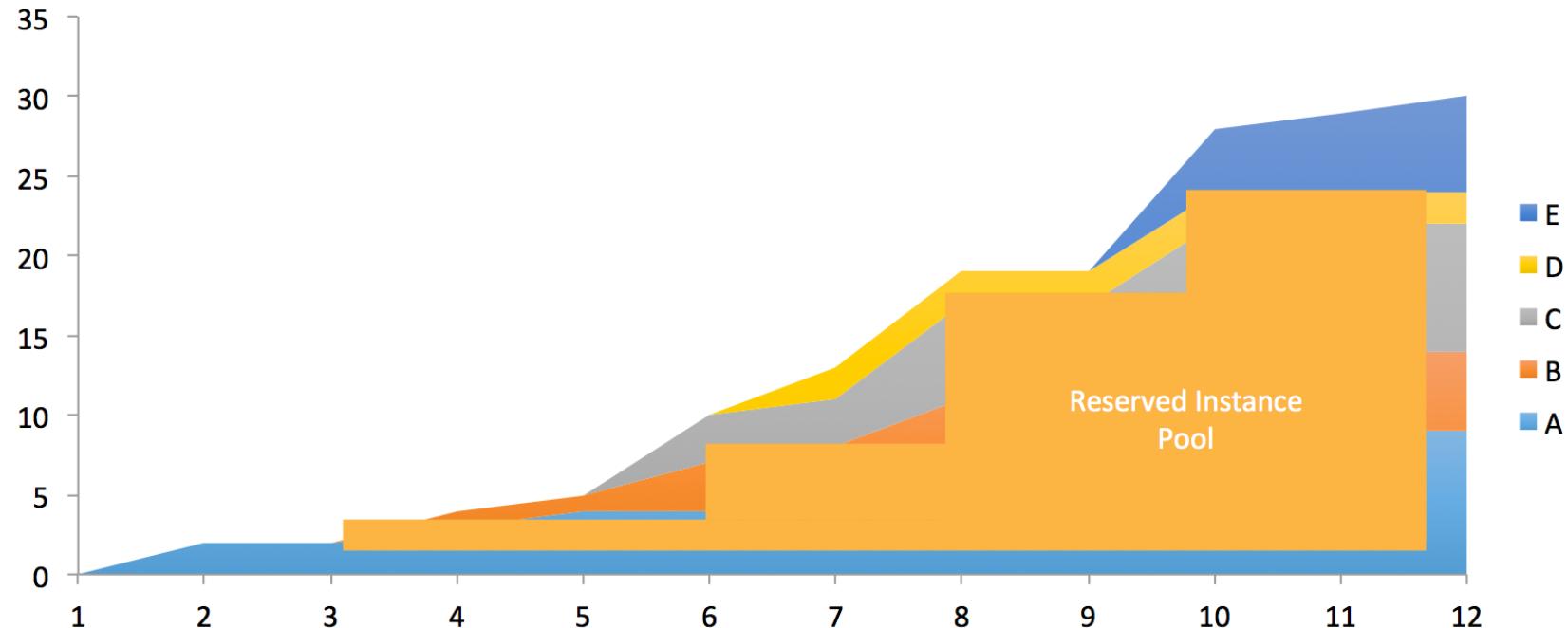
# 게임빌 RI 활용 사례



200여개국 서비스 중  
App Annie 2014 글로벌  
퍼블리셔 11위  
(국내 게임사 중 1위)  
30개 AWS 계정 관리 중  
다양한 RI 옵션을 통해 가격  
절감 중



# RI 구매 전략- 리소스 성장에 따라 단계적



# 4. 스팟 인스턴스 쓰세요!



# 스팟 인스턴스 규칙(SI) 및 활용 사례

- 가격 규칙

- ✓ 수요/공급에 따라 가격 결정
- ✓ 시간당 가격의 최대치는 고객이 선택
- ✓ 스팟 가격이 더 낮으면 인스턴스 시작
- ✓ 스팟 가격이 더 높으면 인스턴스 종료 (terminated)
- ✓ 대비책: Fault Tolerant 아키텍처로 설계!



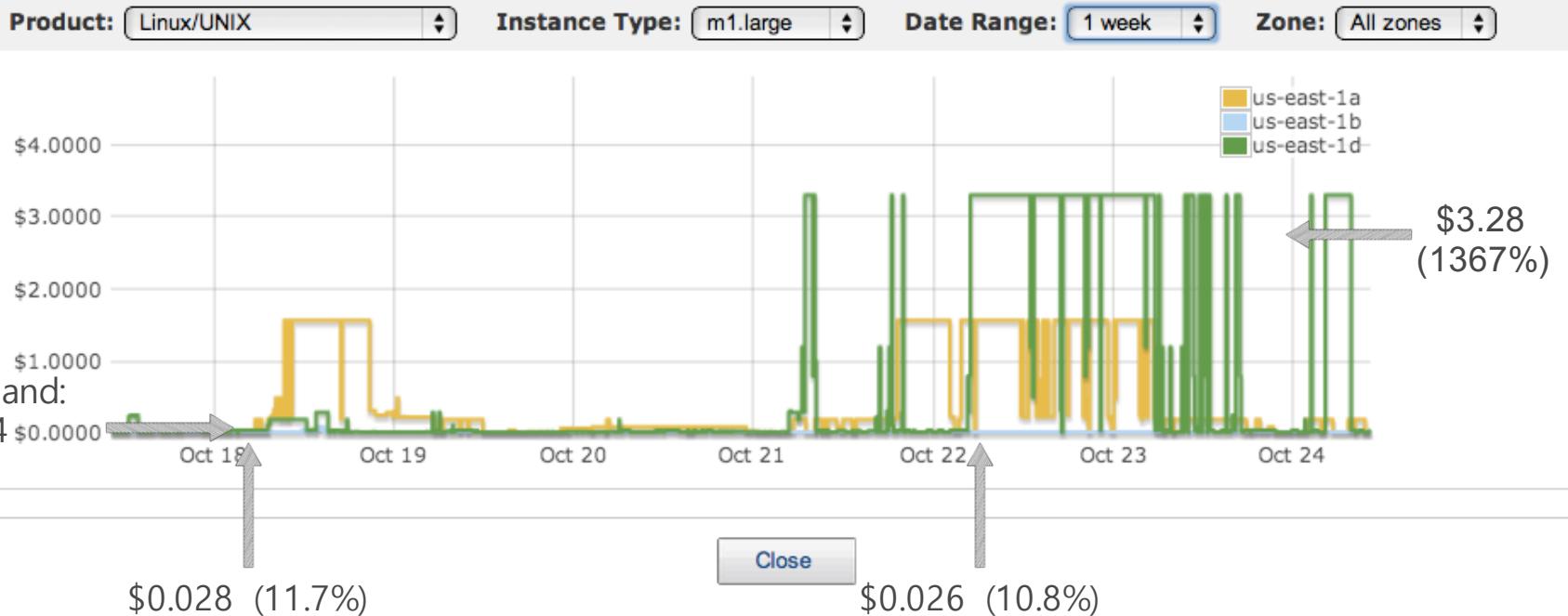
- 활용 사례

- ✓ 배치 작업, 짧은 시간 데이터 분석(Amazon Elastic MapReduce)
- ✓ 웹 크롤링, 프로그램 컴파일, 이미지 변환, 동영상 렌더링

# 스팟 인스턴스(SI) 가격 변동

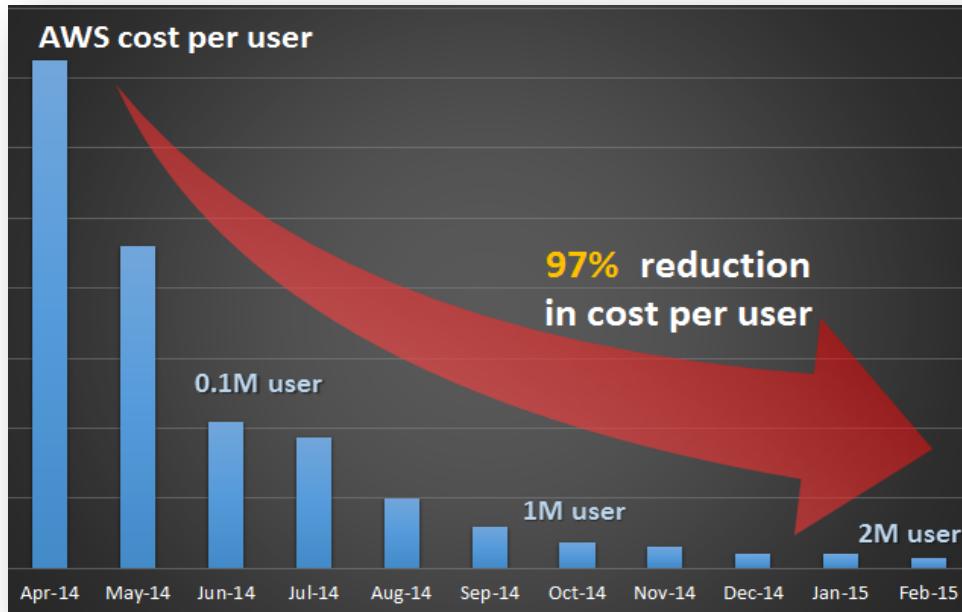
## Spot Instance Pricing History

Cancel X



# 비트패킹의 스팟 인스턴스 활용 사례

- 비트에서 사용하는 서버의 80%가 스팟 인스턴스
- 온디멘드/예약/스팟을 적절히 조합하여 비용 절감



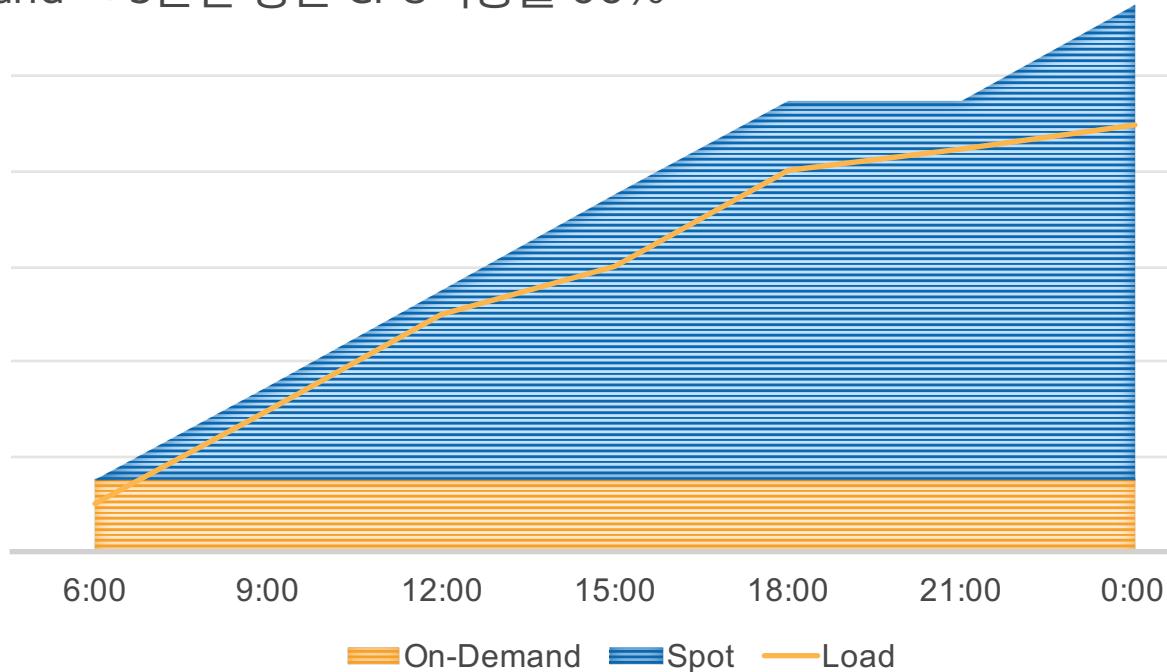
"저희는 AWS를 통해 끊임없이 비용을 최적화하고 있으며, 청취자 한 명 당 소요되는 AWS 비용은 근 1년 전 보다 97% 가까이 줄었습니다."

-비트패킹컴퍼니  
박수만 대표-

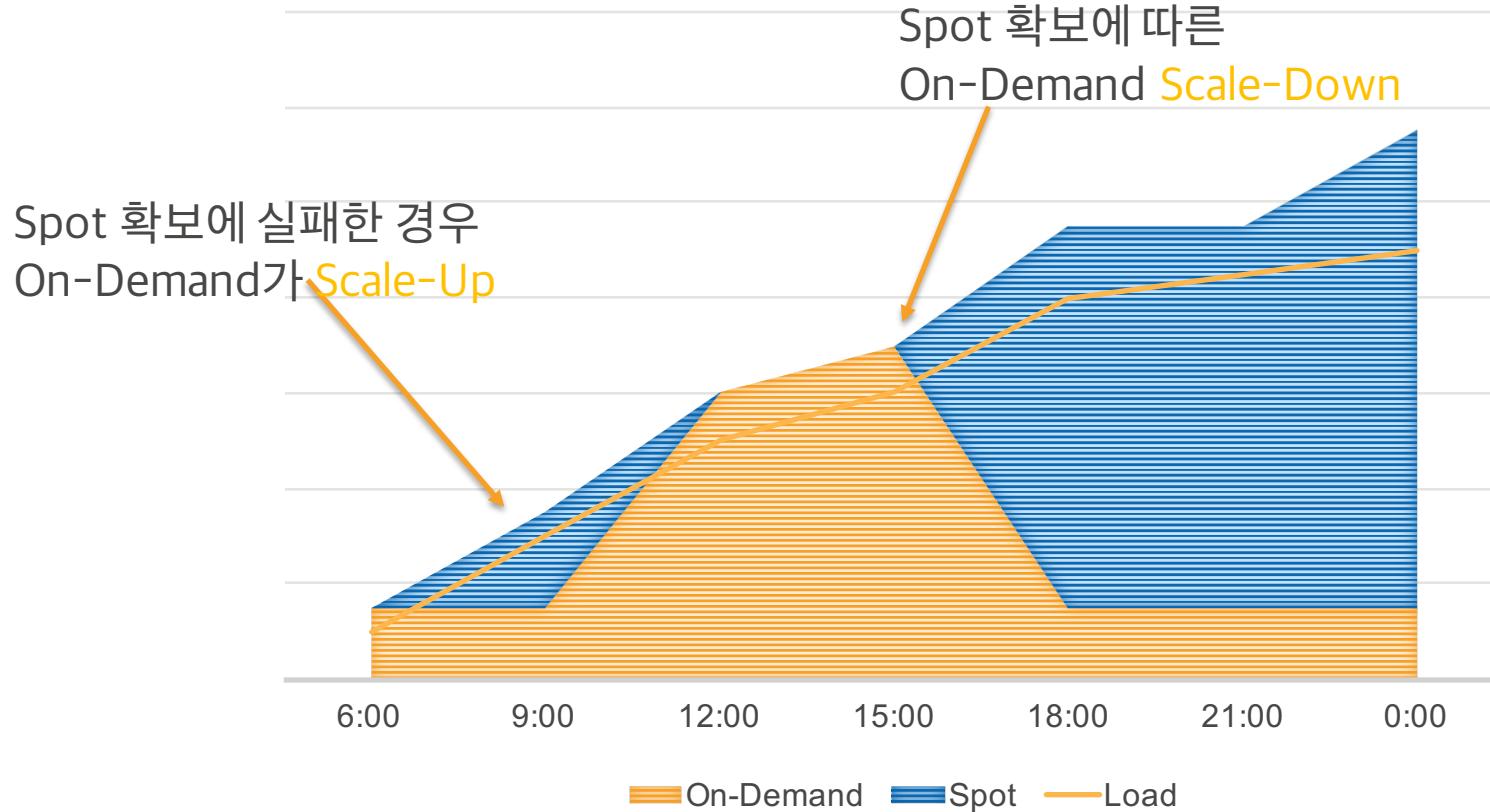
# 비트패킹의 스팟 인스턴스 활용 사례

Spot : 5분간 평균 CPU이용률 40%

On-Demand : 5분간 평균 CPU이용률 60%



# 비트패킹의 스팟 인스턴스 활용 사례



# 스팟인스턴스 요점정리

- AZ별 매우 동적으로 가격변동
- 비용을 80~90%까지 절약가능
- Auto Scaling 적극 이용!
  - 스팟 그룹 + 온디멘드 그룹
  - 두 가지 장점 동시 획득
- 최근 신규 기능
  - Spot Bid Advisor 콘솔
  - Spot Fleet API 기능

Spot Bid Advisor

Region: US East (N. Virginia) OS: Linux/UNIX Bid Price: 25% On-Demand

Instance type filter: vCPU (min): 1 Memory GiB (min): 0 Instance types supported by EMR

| Instance Type | vCPU | Memory GiB | Savings over On-Demand* | Frequency of being outbid (month) | Frequency of being outbid (week) |
|---------------|------|------------|-------------------------|-----------------------------------|----------------------------------|
| t1.micro      | 1    | 0.613      | 88%                     | Low                               | Low                              |
| hi1.4xlarge   | 16   | 60.5       | 94%                     | Low                               | Low                              |
| cr1.8xlarge   | 32   | 244        | 92%                     | Low                               | Low                              |
| c1.xlarge     | 8    | 7          | 88%                     | Low                               | Low                              |
| c1.medium     | 2    | 1.7        | 88%                     | Low                               | Low                              |
| m4.large      | 2    | 8          | 88%                     | Low                               | Low                              |
| m4.xlarge     | 4    | 16         | 88%                     | Low                               | Low                              |
| m1.large      | 2    | 7.5        | 90%                     | Low                               | Low                              |
| m1.medium     | 1    | 3.75       | 89%                     | Low                               | Low                              |
| m1.small      | 1    | 1.7        | 84%                     | Low                               | Low                              |

Display all 44 instance types

# 5. 다양한 스토리지 옵션 활용

## Amazon S3 - 무제한 스토리지

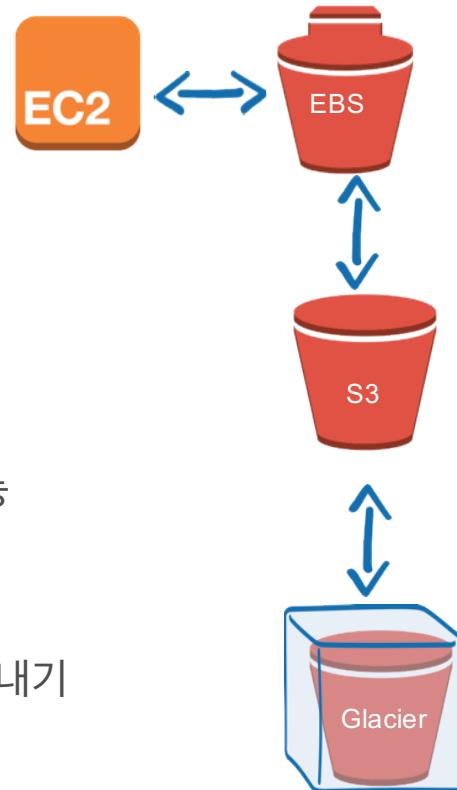
- 99.99% 가용성 / 99.999999999% 내구성
- EBS에 대비 20% 까지 가격이 저렴

## Amazon S3 - Infrequent Access Storage

- S3와 같은 내구성 및 성능 / 99.9% 가용성
- 기존 S3 대비 58% 가격 절감 가능 / 최근 백업에 사용 가능

## Amazon Glacier - 데이터 백업

- S3와 같은 내구성, 성능 및 가용성 / 3~5시간 내 데이터 꺼내기
- S3 표준 대비 최고 77% 가격이 저렴
- 아케이빙, 장기간 백업 및 오래된 로그 데이터





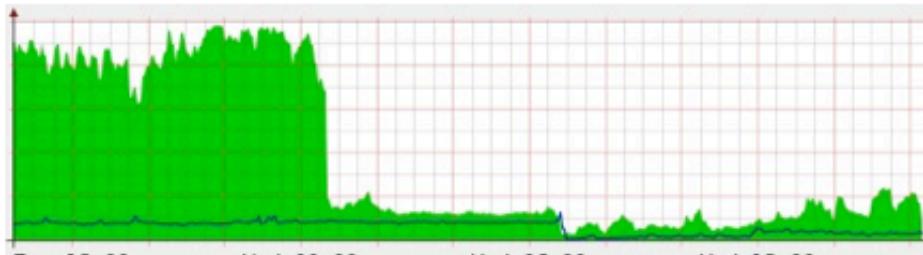
Arzumy MD

@arzumy

Follow

After serving our precompiled assets through S3, we cut down our servers' bandwidth usage by 80%  
[pic.twitter.com/105N1Ugs](http://pic.twitter.com/105N1Ugs)

Reply Retweet Favorite More



1  
RETWEET



1:30 AM - 1 Aug 12

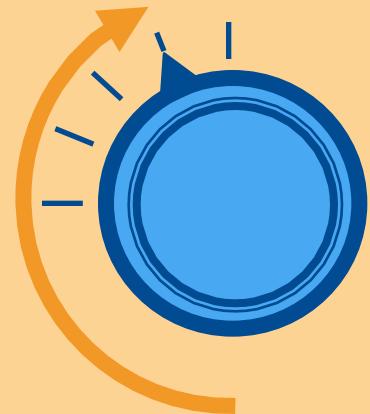
Flag media





# Amazon DynamoDB

- NoSQL 매니지드 서비스
- Read/Write 처리용량(CapacityUnit)을 조절
- 용량 및 확장 제한 없음
- SSD 내구성 및 10ms 이하 빠른 응답



- 주의점:
  - ✓ Capacity Error 발생하지 않을 만큼까지는 넉넉히 프로비저닝 필요
  - ✓ 피크 (Peaks) 대비 지속적으로 모니터링/조정 필요

# 6. Amazon DynamoDB 비용 최적화 방안

## Read 용량을 줄이기 위해 캐싱 활용

- Amazon ElastiCache 서비스 활용



## 용량 단위(CapacityUnit) 최적화 전략

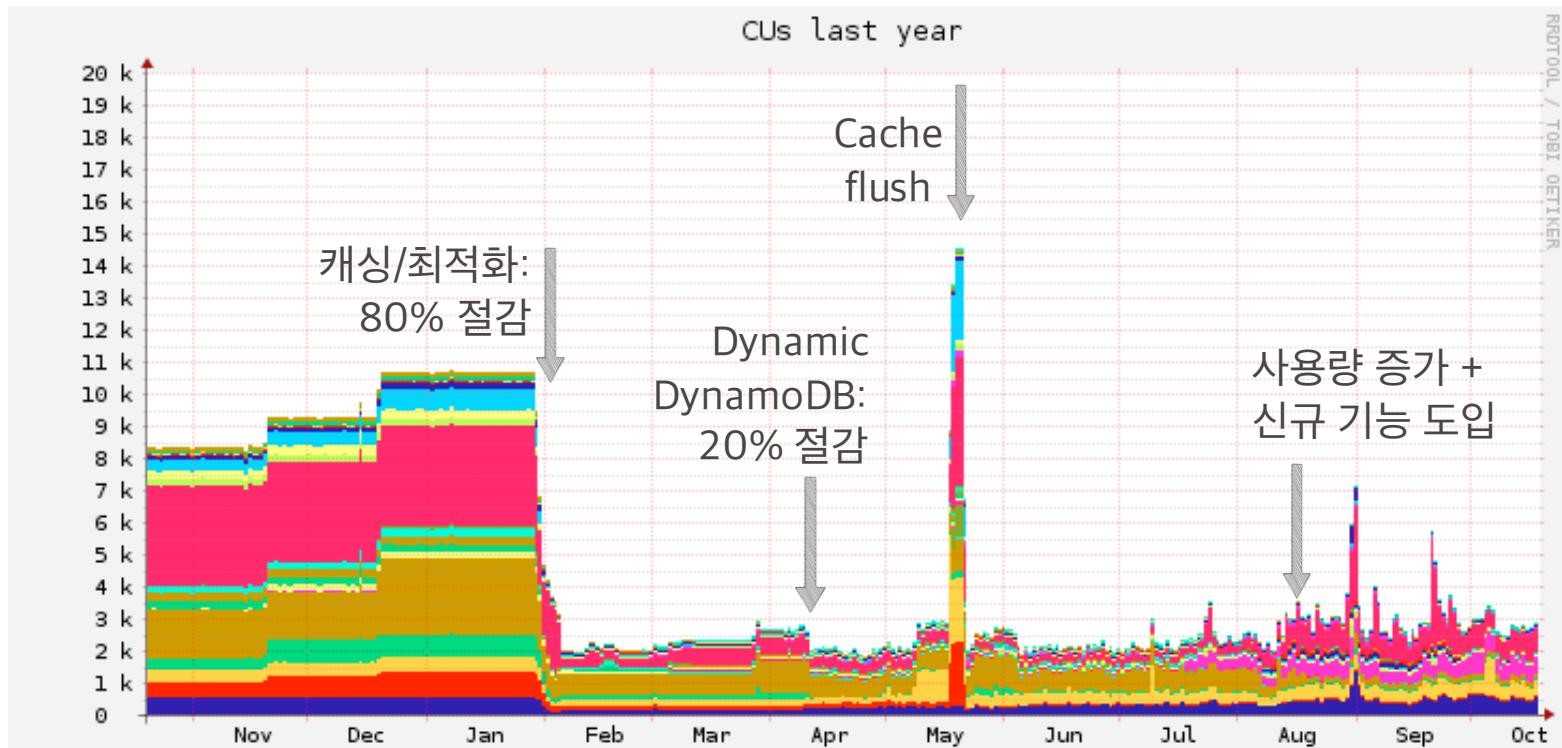
- 테이블 나누기를 통한 최소화 및 데이터 압축하기
- 시간 단위의 데이터 활용 패턴 이해하기
- Amazon SQS를 통해 쓰기 용량에 대한 버퍼 마련하기



## Dynamic DynamoDB 활용하기

- 자동으로 Read/Write CU 최적화 및 자동 설정
- <https://github.com/sebdah/dynamic-dynamodb>

# DynamoDB 최적화 사례

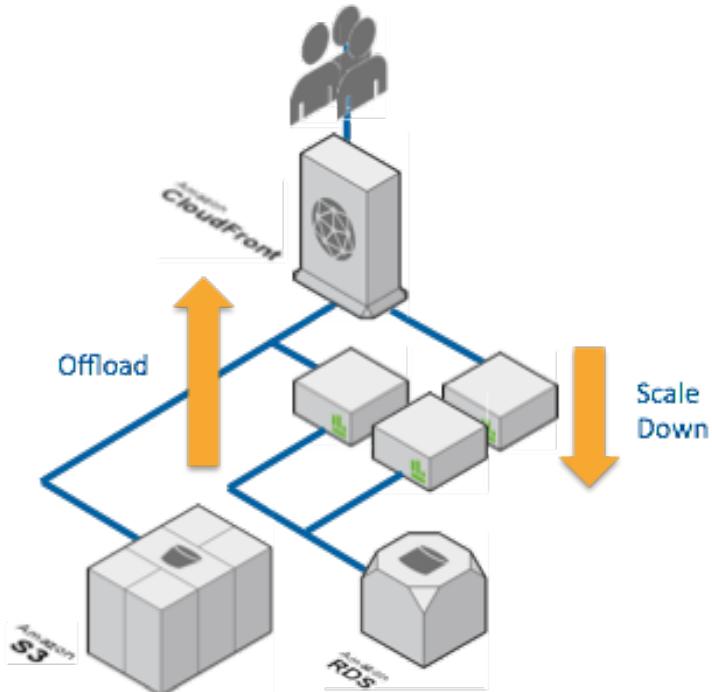


# 7. 아키텍처 부하를 떼어내세요

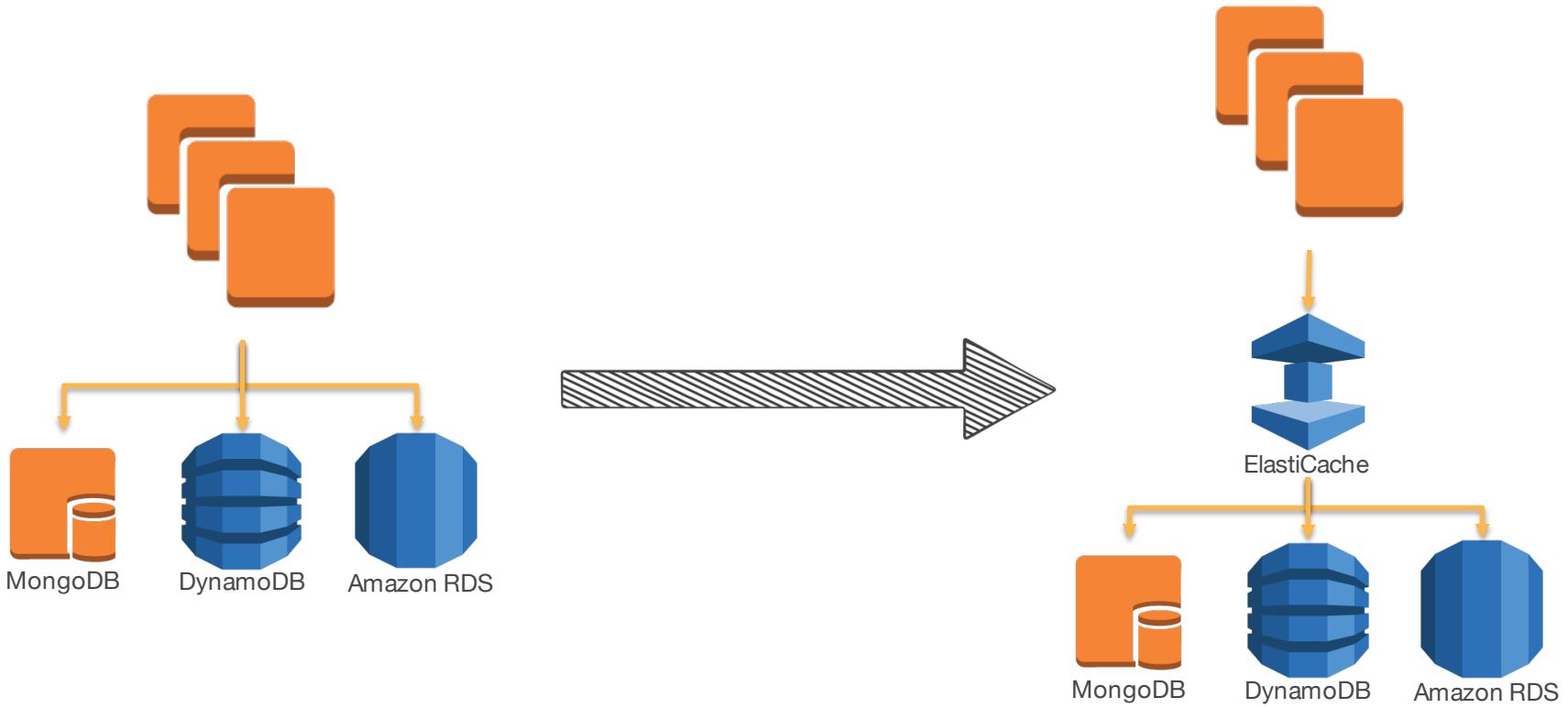
부하를 떼어낼 수록,  
유지보수하고 확장하고 비용 내야 하는  
인프라도 작아짐

## 부하를 떼어내는 세 가지 방법:

- 데이터 캐싱은 항상 옳다!
- 컨텐츠 배포 네트워크 활용하라!
- 다양한 AWS 관리 서비스를 사용하라!



# 1) 데이터 캐싱(Caching)은 항상 옳다!





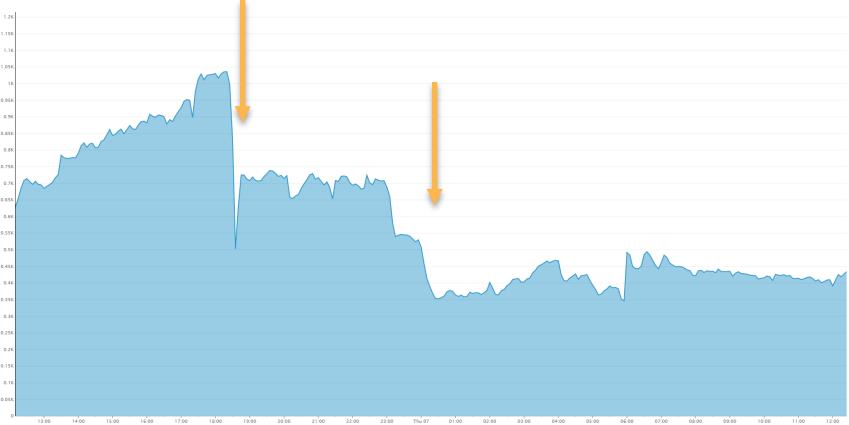
# Amazon ElastiCache

- 인메모리 기반 관리형 서비스
- Memcached 또는 Redis 클러스터 구성 가능
- 멀티 AZ 기반 고가용성
- Read replica 구성 가능
- 모니터링 및 알람 기능 제공

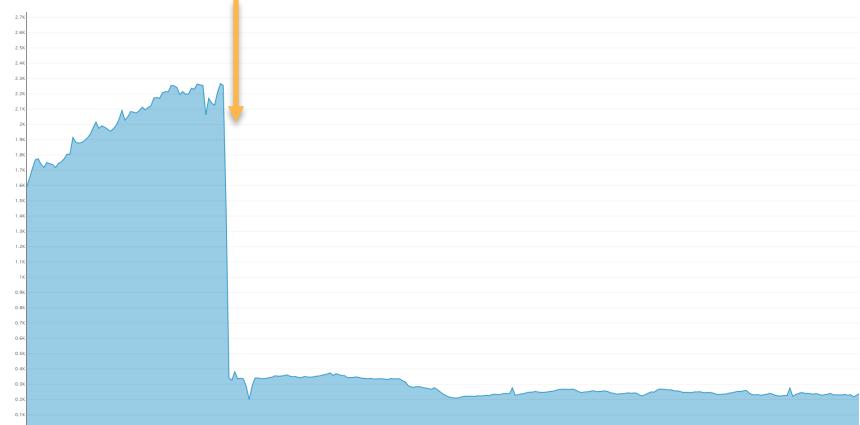


# 1) 데이터 캐싱(Caching)은 항상 옳다!

MongoDB Queries



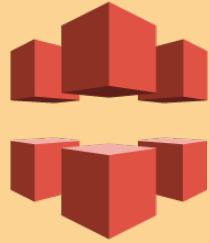
DynamoDB Reads



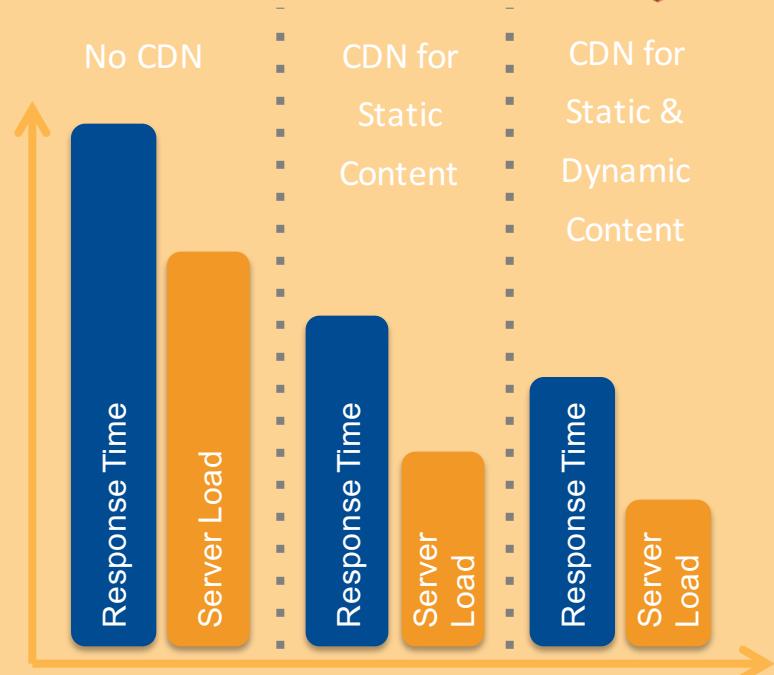
- Shut down 8 machines
- Saved 3k reads per second  
(>20k reads per second in total)



# Amazon CloudFront



- 글로벌 콘텐츠 배포 네트워크
- 이미지, 동영상, 웹 페이지, 사용자 파일 전송
- 지리적 위치에 따라 더 빠른 접속 속도 제공
- 정적/동적 콘텐츠 캐싱 및 가속



## 2) 콘텐츠배포네트워크(CDN)을 활용하라!

### EC2 인스턴스로 설정 시

- 4 x Medium Instances \$485
- AWS Data Transfer 1 TB \$194
- **Total = \$679**

### CloudFront로 최적화 시

- 1 x Medium Instance \$121
- CloudFront Data 1 TB \$168
- CloudFront Requests \$1.89
- **Total = \$291**

**57% Lower Cost + 6X Faster**

# 3) 다양한 AWS 관리 서비스를 활용하라

- DB/빅 데이터 분석 관리형 서비스
  - RDB: Amazon RDS NoSQL: Amazon DynamoDB Cache: Amazon ElastiCache
  - Hadoop: Amazon Elastic MapReduce DW: Amazon Redshift
  - 실시간 분석: Amazon Kinesis
- 검색 엔진 관리형 서비스
  - Amazon CloudSearch or Amazon Elasticsearch Service
- 애플리케이션 관리형 서비스
  - 동영상 인코딩: Amazon Elastic Transcoder
  - 이메일 전송: Amazon Simple Email Service
  - 큐 서비스: Amazon Simple Queue Service
  - 푸시 알림: Amazon Simple Notification Service

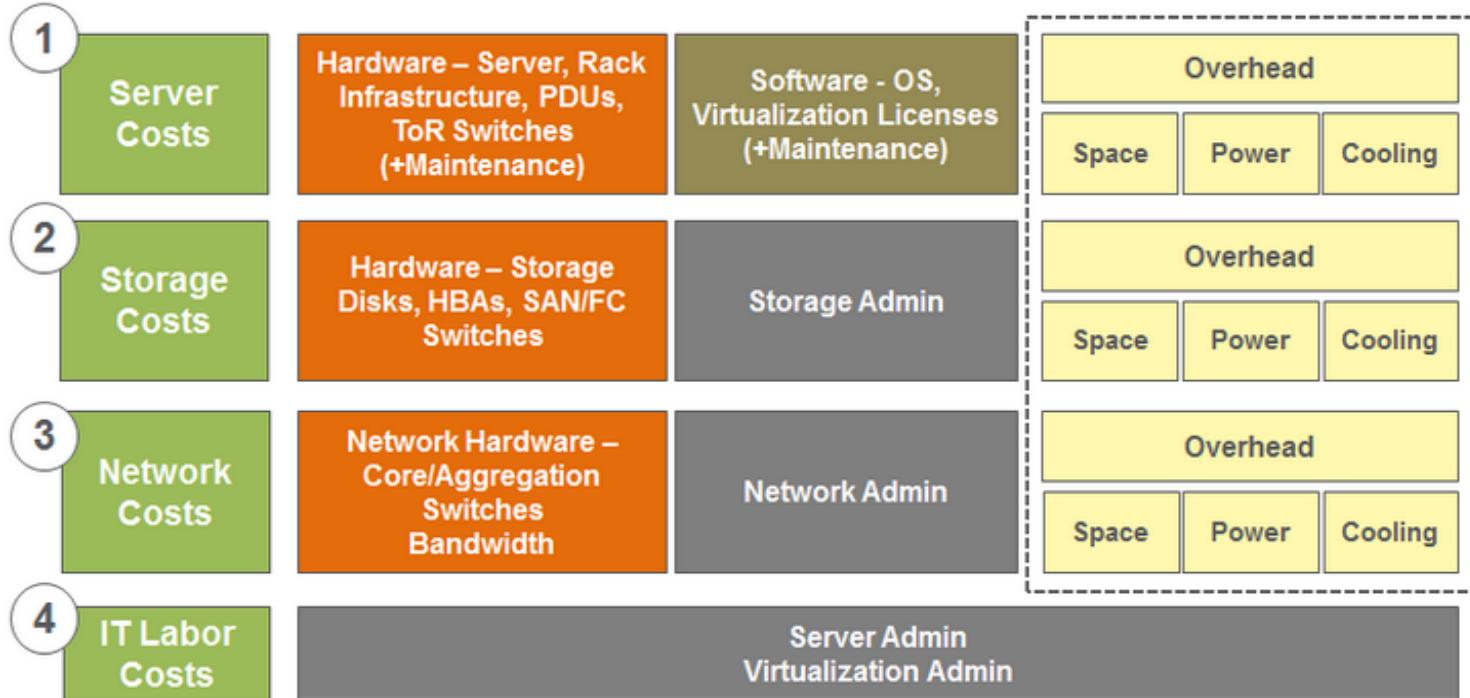
직접 만드는 것 보다 더 저렴하고, 높은 성능 및 관리가 편하다!



# 4. 비용 모니터링 / 분석 도구

# AWS TCO 계산기

<https://awstcocalculator.com>



# AWS TCO 계산기

<https://awstcocalculator.com>

 Contact Sales [Download Report](#)

## AWS Total Cost of Ownership (TCO) Calculator

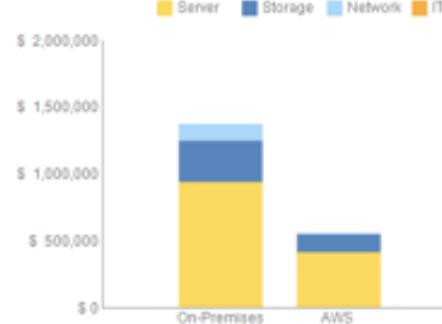
[« Change Input](#)

**On-Premises vs. AWS Summary**

You could save **60%** a year by moving your infrastructure to AWS.  
Your three year total savings would be **\$821,570**.

**3 Years Cost Breakdown**

Legend: Server (Yellow), Storage (Blue), Network (Light Blue), IT Labor (Orange)



|          | 3 Yr. Total Cost of Ownership |           |
|----------|-------------------------------|-----------|
|          | On-Premises                   | AWS       |
| Server   | \$947,412                     | \$420,659 |
| Storage  | \$306,160                     | \$130,420 |
| Network  | \$119,077                     | \$ -      |
| IT-Labor | \$ -                          | \$ -      |
| Total    | \$1,372,649                   | \$661,079 |

AWS cost includes business level support

# AWS 요금 계산기

언어: Korean 

Need Help? [Watch the Videos](#) or [Read 'How AWS Pricing Works' Whitepaper](#)

AWS pricing helps you reduce costs in multiple ways. [Learn more about AWS's pricing philosophy »](#)

프리 티어: 신규 고객은 처음 12개월 동안 프리 티어를 사용할 수 있습니다.

모두 초기화

서비스 월별 청구서 견적 (\$ 1328.44)

지역 선택: 아시아 태평양(일본)

인바운드 데이터 전송은 무료이며 아웃바운드 데이터 전송은 매월 지역당 1GB가 무료로 제공됩니다.

Amazon Elastic Compute Cloud(Amazon EC2)는 클라우드에서 규모를 자유자재로 변경할 수 있는 컴퓨팅 파워를 제공하는 웹 서비스입니다. 이 서비스는 개발자가 웹 스케일 처리 능력을 쉽게 사용할 수 있도록 설계되었습니다. Amazon Elastic Block Store(EBS)는 Amazon EC2 인스턴스에 영구 스토리지를 제공합니다.

영식 지원하기

컴퓨팅: Amazon EC2 인스턴스:

| 설명          | 인스턴스 | 사용량       | 유형              | 결제 옵션        | 월별 비용    |
|-------------|------|-----------|-----------------|--------------|----------|
| App Servers | 2    | 100 사용률/월 | Linux(m1.small) | 온 디맨드(계약 없음) | \$ 89.32 |
| Web Servers | 2    | 100 사용률/월 | Linux(m1.small) | 온 디맨드(계약 없음) | \$ 89.32 |

+ 새 행 추가

스토리지: Amazon EBS 볼륨:

| 설명            | 볼륨 | 볼륨 유형    | 스토리지   | IOPS | 스냅샷 스토리지      |
|---------------|----|----------|--------|------|---------------|
| Customer data | 4  | Magnetic | 300 GB | 100  | 10 일별 스냅샷 변경률 |

+ 새 행 추가

엘라스틱 IP:

추가 엘라스틱 IP 개수: 1

일반 고객 샘플

- AWS 무료 웹 사이트
- AWS Elastic Beanstalk 기본 설정
- 마케팅 웹 사이트
- 대규모 웹 애플리케이션(모두 온 디맨드)
- 미디어 애플리케이션
- 유럽 웹 애플리케이션
- 재해 복구 및 백업

<http://calculator.s3.amazonaws.com/index.html>



# AWS 대금 및 비용 관리 보드

AWS 대금 및 비용 관리 대시보드

대시보드

Channy Yun Global Support

AWS Services Edit

대금 및 비용 관리 대시보드

소비 요약

AWS 계정 결제 콘솔입니다. 지난달 비용, 이번 달 현재까지 비용, 이번 달 말까지 예상 비용이 아래에 나타납니다.

현재 달 누적잔액 산정 대상: 1월 2016

\$37.47

서비스별 이번 달 현재까지 소비

아래 차트로는 각 서비스에 사용한 금액의 비율을 확인하실 수 있습니다.

청구서 세부 정보

?

EC2 Kinesis Route53 ES 기타 서비스

소비별 이번 달 현재까지 최상위 서비스

| 서비스     | 금액      |
|---------|---------|
| EC2     | \$18.08 |
| Kinesis | \$17.69 |
| Route53 | \$1.00  |
| ES      | \$0.68  |
| 기타 서비스  | \$0.02  |
| 세금      | \$0.00  |

이런 비용에 대한 중요한 정보

구독 요금 포함

# AWS 비용 탐색기 - Cost Explorer



대시보드  
청구서  
**비용 탐색기**  
결제 방법  
결제 내역  
통합 결제  
보고서  
기본 설정  
크레딧  
세금 설정  
DevPay

## 비용 탐색기



### 비용 탐색기 시작

소비를 그래프로 그려 시각화하

### 미리 구성된 보기

일반적인 소비 질의에 답하기 위해 이미 ·

#### 서비스별 월별 소비 보기

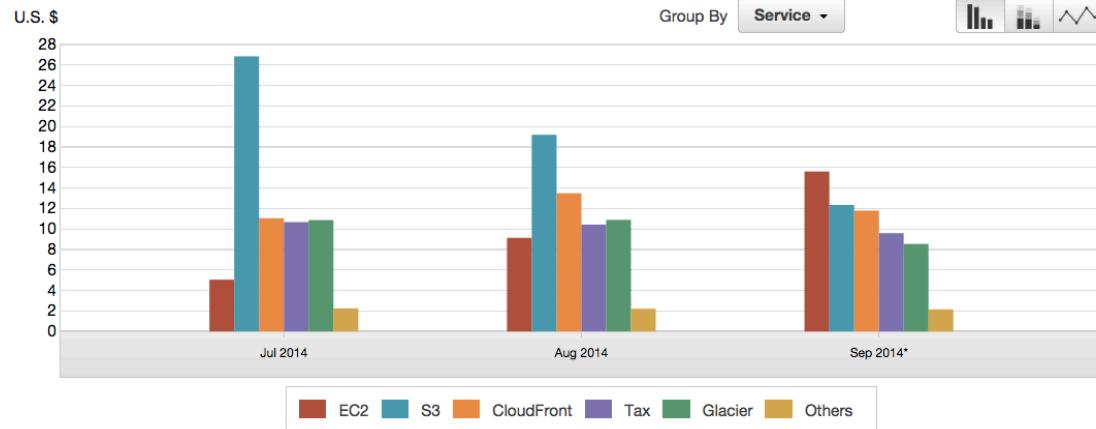
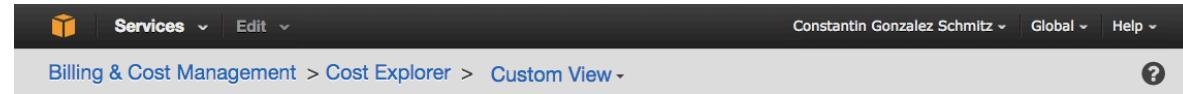
지난 3개월 동안의 월별 소비를

#### 연결된 계정별 월별 소비 보기

지난 3개월 동안의 월별 소비를

#### 일별 소비 보기

지난 60일간 일일 소비



# AWS 서비스별 예산 기능



대시보드

청구서

비용 탐색기

**예산**

보고서

비용 할당 태그

결제 방법

결제 내역

통합 결제

기본 설정

크레딧

세금 설정

DevPay

## 예산

예산 기능을 통해 AWS 비용에 대한 월간 예산을 세울 수 있습니다. 집계 비용("총 비용")에 대한 예산을 작성하거나, 계정, 서비스, 태그, 가용 영역("AZ"), 구매 옵션(예: 예약), API 작업을 비롯하여 특정 비용 차원 또는 비용 차원 그룹과 관련된 비용만 포함하도록 범위를 좁힐 수도 있습니다.

또한 예산에 이메일 알림을 연결하여 현재 비용이나 예상 비용이 예산 비용에 대해 정의한 한계 비율을 초과했을 때 알리도록 할 수 있습니다.

**예산 작성**

## 운영 방법



### 예산 작성

예산에는 AWS의 총 비용 또는 일부 비용을 포함시킬 수 있으며, 선택한 기간 동안 매월 반복하여 예산을 작성할 수 있습니다.

[자세히 알아보기 »](#)



### 예산에 알림 연결

알림을 사용하면 예산을 자동으로 모니터링하여 현재 비용 또는 예상 비용이 예산 비용을 초과할 경우 이메일로 알리도록 할 수 있습니다.

[자세히 알아보기 »](#)



### 예산 모니터링

예산 대시보드를 통해 언제든지 예산에 대한 현재 비용과 예상 비용을 모니터링할 수 있습니다. 예산 대시보드에는 자세한 예산 정보와 차액 데이터가 제공됩니다.

[자세히 알아보기 »](#)

# AWS 빌링 알림 - CloudWatch

Sales    Services    Edit    Constantin Gonzalez    N. Virginia    Help

Dashboard    Alarms    ALARM (2)    INSUFFICIENT (0)    OK (0)

Billing    Logs    Metrics (4)    Selected Metrics    Billing    DynamoDB    EBS    EC2    AWS/Kinesis    Redshift    Custom Metrics...

**Billing Metrics**

**Billing > Total Estimated Charge**

| Currency | Metric Name      |
|----------|------------------|
| USD      | EstimatedCharges |

**Billing > By Service**

| ServiceName        | Currency | Metric Name      |
|--------------------|----------|------------------|
| AWSDataTransfer    | USD      | EstimatedCharges |
| AWSSupportBusiness | USD      | EstimatedCharges |
| AmazonCloudFront   | USD      | EstimatedCharges |
| AmazonDynamoDB     | USD      | EstimatedCharges |
| AmazonEC2          | USD      | EstimatedCharges |
| AmazonKinesis      | USD      | EstimatedCharges |

EstimatedCharges (Maximum) 6 Hours

Time Range: Relative (From: 14 days ago, To: 0 minutes ago), Absolute, UTC (GMT)

Tools: Copy URL, Create Alarm

Feedback

**awsbilling-AWS-Service-Charges...**

EstimatedCharges > 2000

2,500  
2,000  
1,500  
1,000  
500  
0

9/19 00:00 9/21 00:00 9/23 00:00

EstimatedCharges (Maximum) 6 Hours

Time Range: Relative (From: 14 days ago, To: 0 minutes ago), Absolute, UTC (GMT)

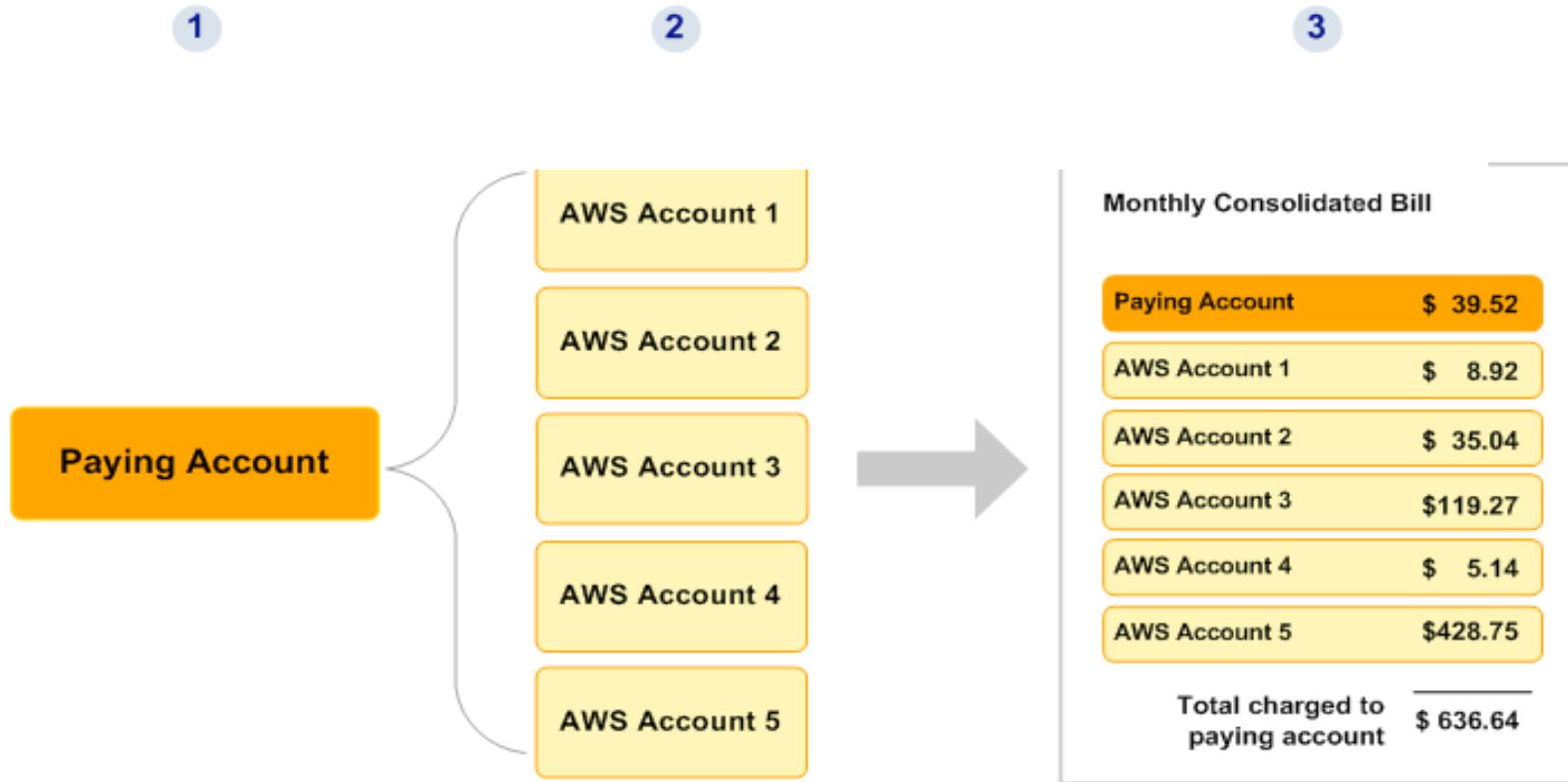
Tools: Copy URL, Create Alarm

Feedback

© 2008 - 2014, Amazon Web Services, Inc. or its affiliates. All rights reserved.    Privacy Policy    Terms of Use

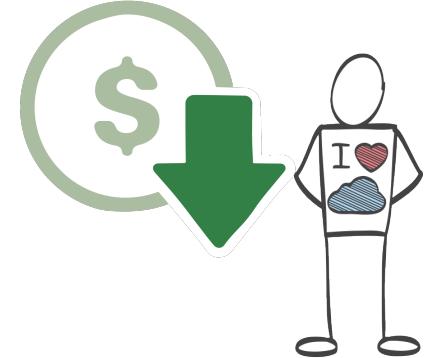
amazon web services

# AWS 통합 결제(Consolidated Billing)



# 정리 들어갑니다!

1. 사용 안 하는 인스턴스는 끄세요
2. 딱 맞는 인스턴스 타입을 고르세요!
3. Auto Scaling을 쓰세요!
4. 예약 인스턴스(RI) 쓰세요!
5. 스팟 인스턴스 잘 쓰세요!
6. Amazon S3/Glacier 스토리지 활용하세요
7. Amazon DynamoDB 최적화 하세요
8. 캐싱/CDN 등으로 아키텍처 부하를 떼어내세요!





# 한가지 더! 서울 리전 사용하세요!

- 1월 7일 AWS Asia-pacific (Seoul) Region 시작
- 네트워크 레이턴시: 도쿄 대비 1/3 수준 (15-20ms)
- S3 스토리지: 도쿄 대비 5% 가량 저렴
- 네트워크 전송 비용: 도쿄 대비 10% 가량 저렴
- EC2 인스턴스: 도쿄 대비 5-10% 저렴 (c4/m4 경우)
- EBS 볼륨: 도쿄 대비 5-10% 저렴 (GIOPS/PIOPS 경우)
- RDS 인스턴스: 도쿄 대비 5% 저렴 (m4 경우)

\* 본 수치는 2016년 1월 7일 현재로서 향후 변경 가능

마지막으로…  
계속 실험해보고 비용을 낮추세요!





# AWS 서비스의 차별점



## 축적된 경험

2006년 부터 클라우드 서비스 운영 경험



## 폭넓은 전문 서비스

다양한 클라우드 업무를 지원하는 50여개 이상



## 혁신의 속도

고객의 피드백을 기반한 빠른 혁신 속도



## 글로벌 인프라

12 리전, 32 가용영역 , 54 엣지 로케이션



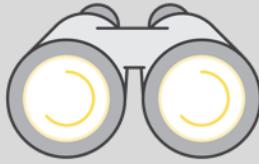
## 가격 철학

51번의 자발적인 가격 인하



## 파트너 생태계

수만개 파트너사 및 수천개 마켓플레이스 제품



# 여러분의 피드백을 기다립니다!

- 공식 블로그: <http://aws.amazon.com/ko/blogs/korea>
- 한국어 공식 소셜 미디어



@AWSKorea



AmazonWebServices



AWSKorea



AWSKorea

