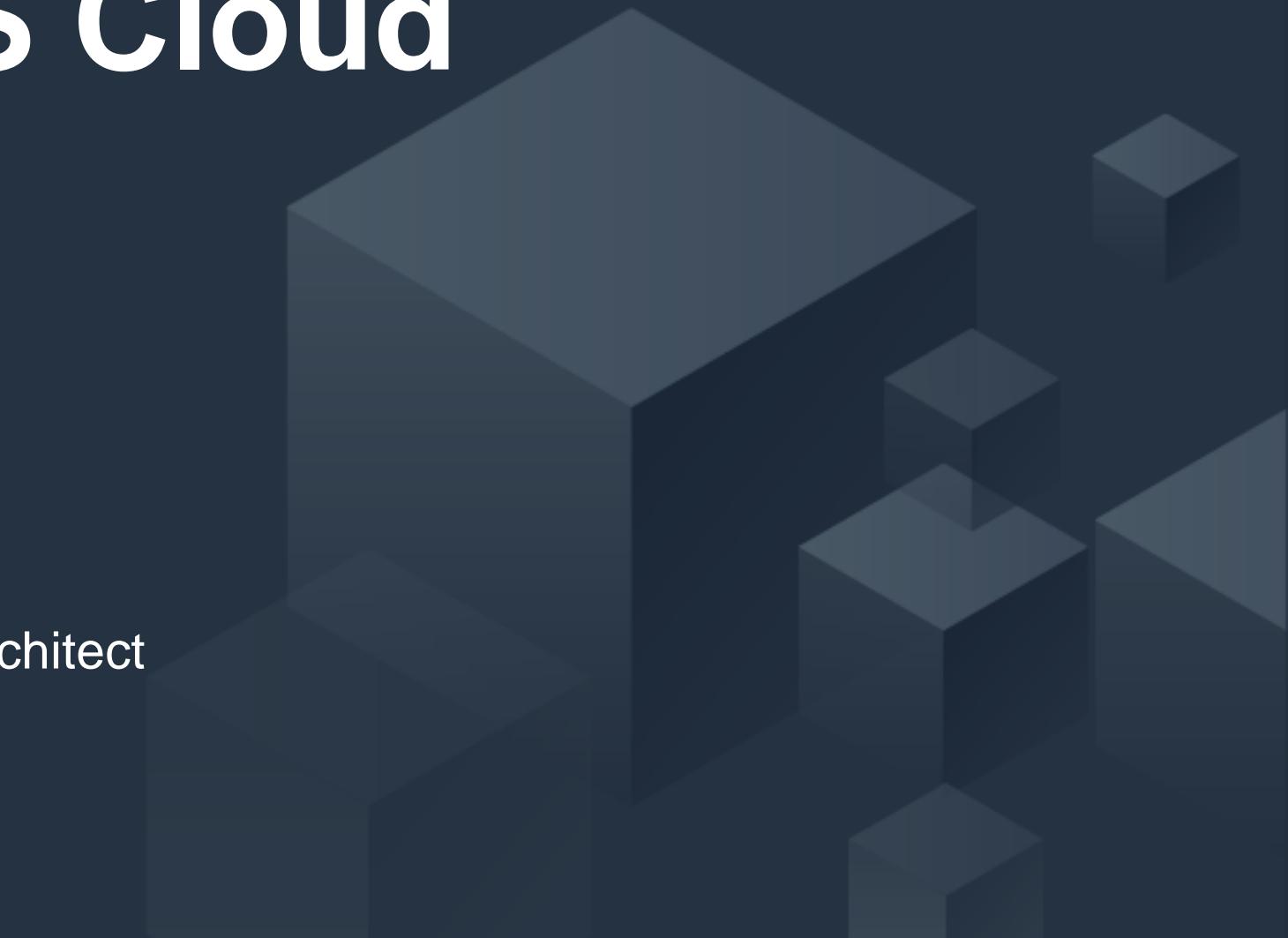




Introduction to AWS Cloud & EC2 Overview

Deven Suri (dsuri@amazon.com) Account Manager
Chetan Agrawal (agrcheta@amazon.com) Solutions Architect

17-Jul-2020



Agenda

- Introduction to AWS Cloud
- AWS Global Infrastructure
- AWS Compute services
- Introduction to EC2
- EC2 deep dive
- Q&A

Our mission:

To be Earth's most
customer-centric company

- 1994 o Founded
- 1995 o Amazon.com
- 1998 o CDs and DVDs
- 2006 o AWS
- 2007 o Kindle
- 2011 o Video
- 2012 o Groceries
- 2014 o Alexa/Echo
- 2015 o Bookstores
- 2017 o Go





Try "Rome"

Become a host Saved Trips Messages Help

Book places to stay and things to do, wherever you go



Homes



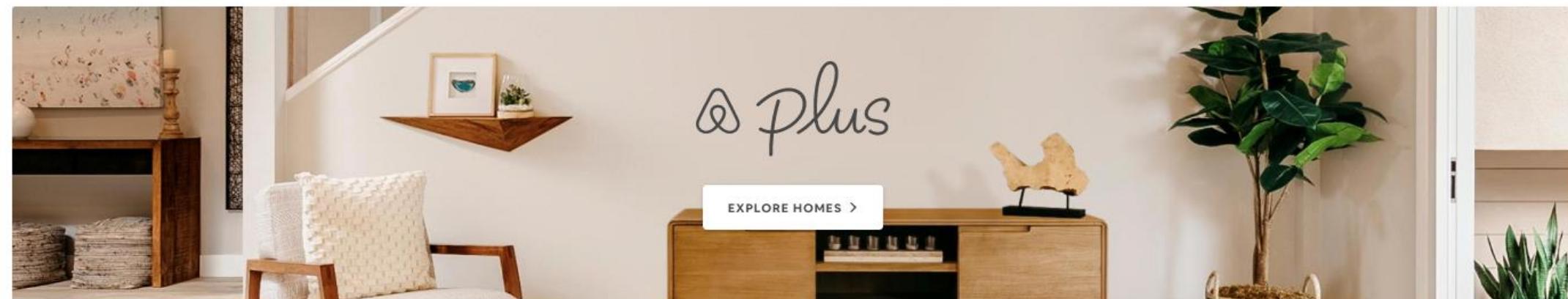
Experiences



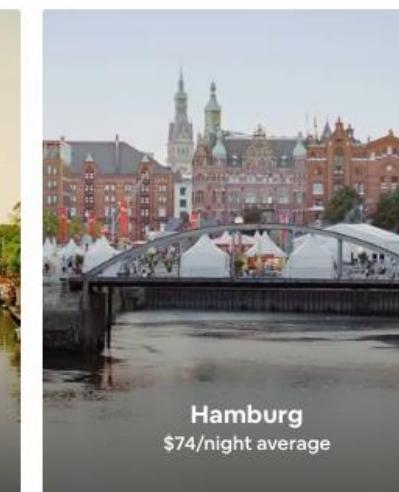
Restaurants

Introducing Airbnb Plus

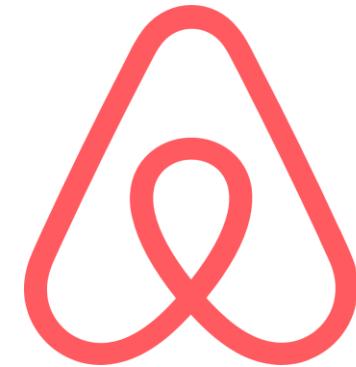
A selection of homes verified for quality and design



Recommended for you



Terms, Privacy, Currency & More



airbnb

500,000,000+

Total Guests arrivals

6,000,000+

Listings Worldwide

Total Number of Guests



15M

12M

9M

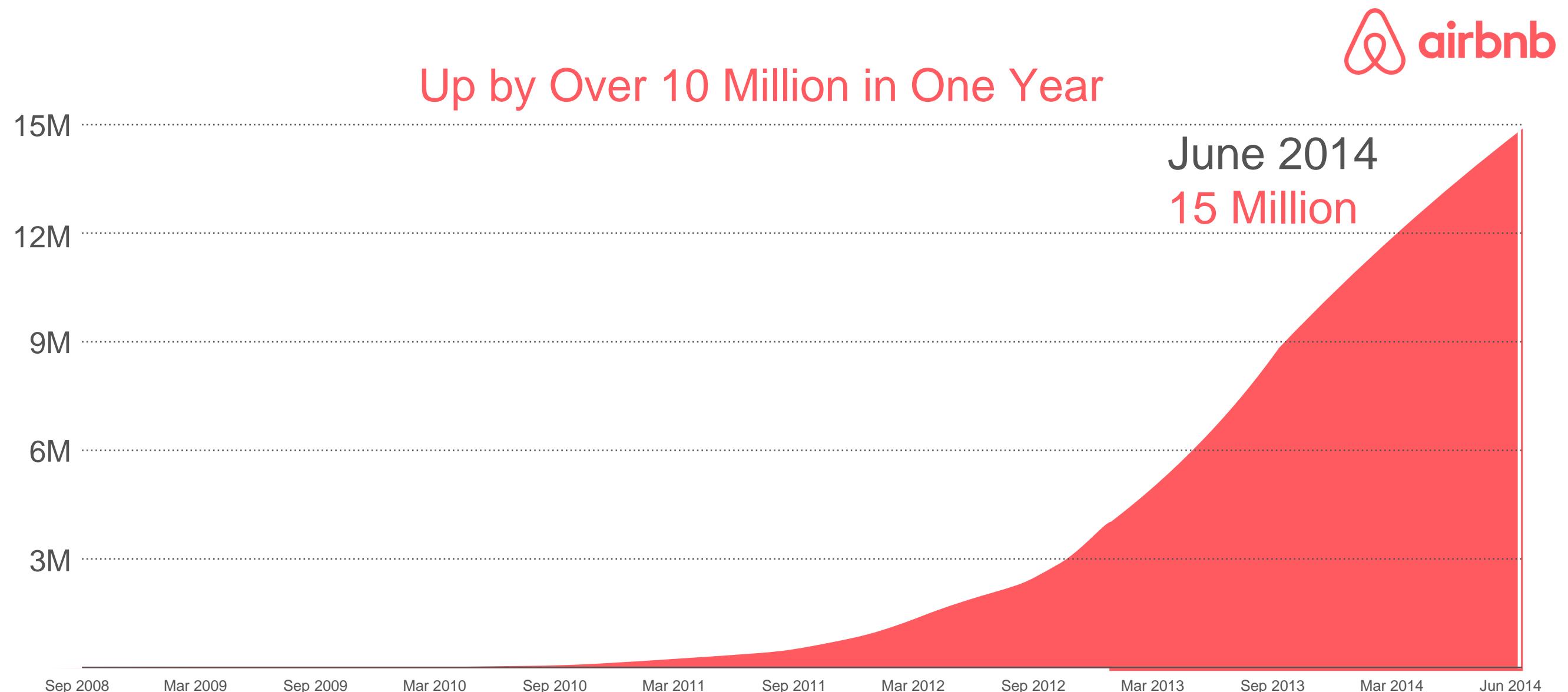
6M

3M

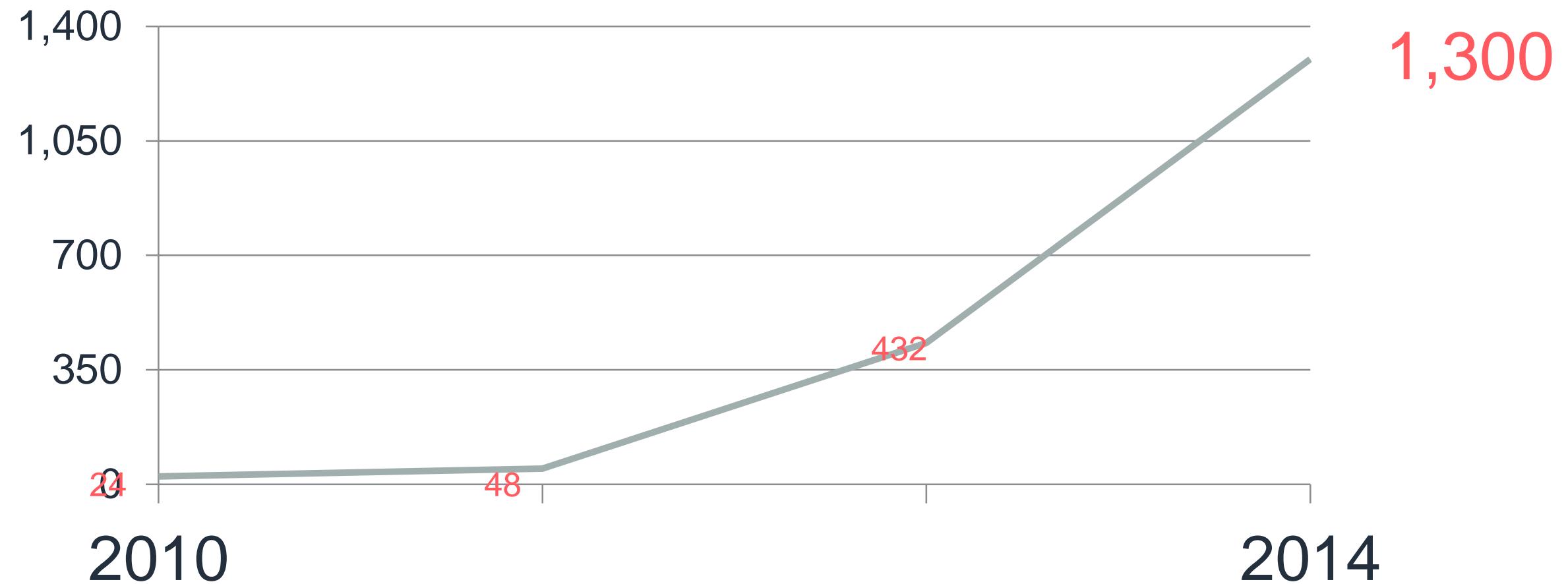
January 2013
4 Million

Sep 2008 Mar 2009 Sep 2009 Mar 2010 Sep 2010 Mar 2011 Sep 2011 Mar 2012 Sep 2012 Mar 2013 Sep 2013 Mar 2014 Jun 2014

Total Number of Guests



Total Number of EC2 instances





We“have a 5 person operations
team.”

AWS allows us to devote our resources
and mindshare to the core business.



Netflix



David Webster • 2nd

Cloud Engineer

1w

"At Netflix, we have...

- 100s of microservices
- 1,000s of daily prod changes
- 10,000s of AWS VMs
- 100,000s customer interactions/sec
- 1,000,000s of customers
- 1,000,000,000s of time series metrics

... and we do this with 10s of ops engineers and 0 data centers." **#aws #netflix #engineers**



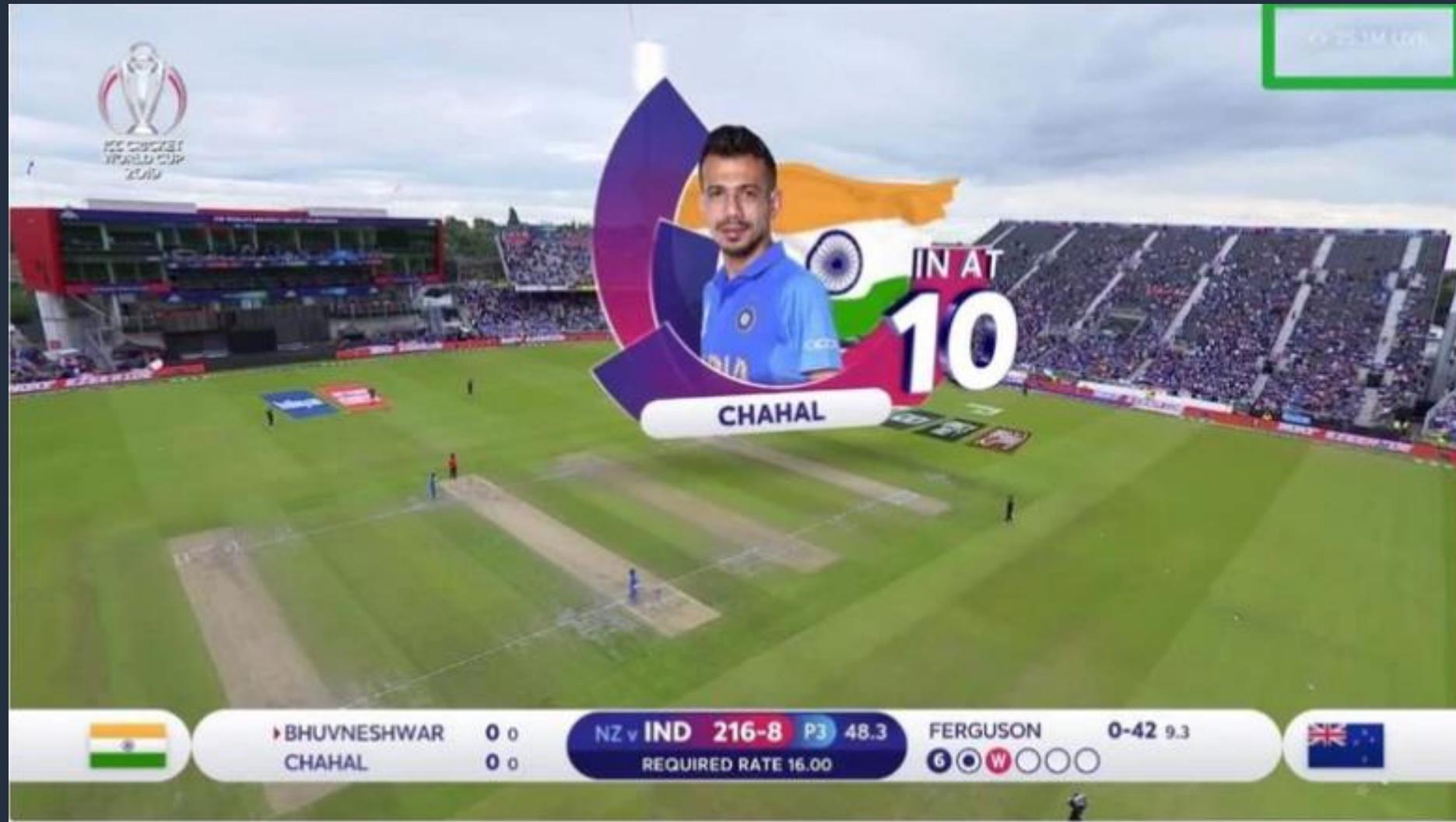
5,270

205 comments



Hotstar WC Coverage

Use Case



<https://yourstory.com/2019/07/hotstar-icc-world-cup-semis-india-record>

AWS Global Infrastructure

Regional expansion

First 5 years: 4 regions

Ireland

N. California

N. Virginia

Singapore

Regional expansion

- First 5 years: 4 regions
- Next 5 years: 7 regions



Regional expansion

- First 5 years: 4 regions
- Next 5 years: 7 regions
- 2016-2020: 13 regions and 1 local region
- Coming soon: 3 regions



AWS Global Infrastructure

24 geographical regions, 1 local region, 76 availability zones, 216 POPs

○ Region & Number of Availability Zones (AZs)

GovCloud (US)

US-East (3), US-West (3)

US West

Oregon (4)

Northern California (3)

US East

N. Virginia (6), Ohio (3)

Canada

Central (3)

South America

São Paulo (3)

Africa

Cape Town (3)

Europe

Frankfurt (3), Paris (3),
Ireland (3), Stockholm (3),
London (3), Milan (3)

Middle East

Bahrain (3)

Asia Pacific

Singapore (3), Sydney (3),
Tokyo (4), Osaka-Local (1)*
Seoul (3), Mumbai (3),
Hong Kong (3)

China

Beijing (2), Ningxia (3)

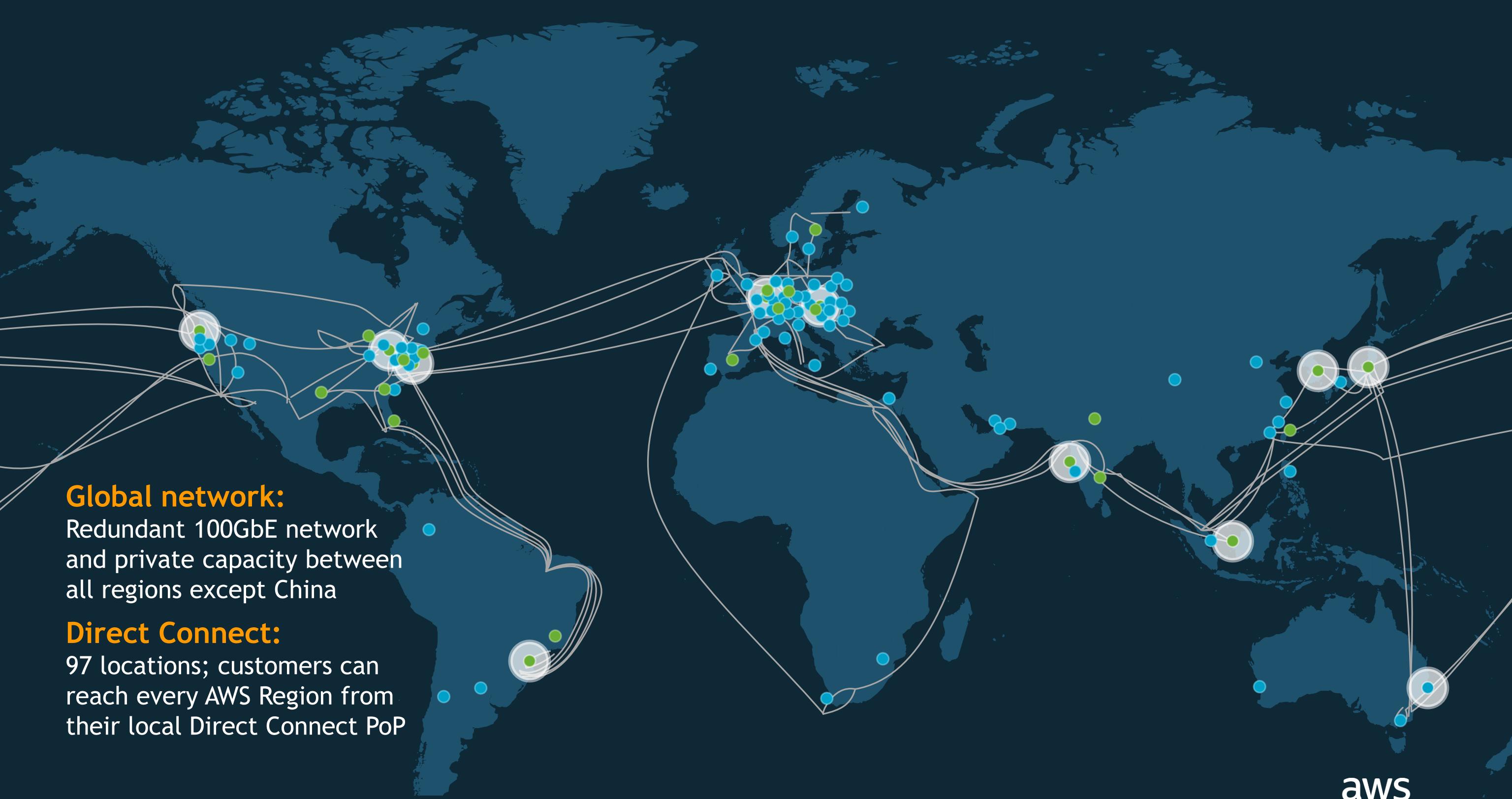
○ Announced Regions

* Available to AWS customers who request access. Customers wishing to use the Asia Pacific (Osaka) Local Region should speak with their sales representative.

© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved. Amazon Confidential

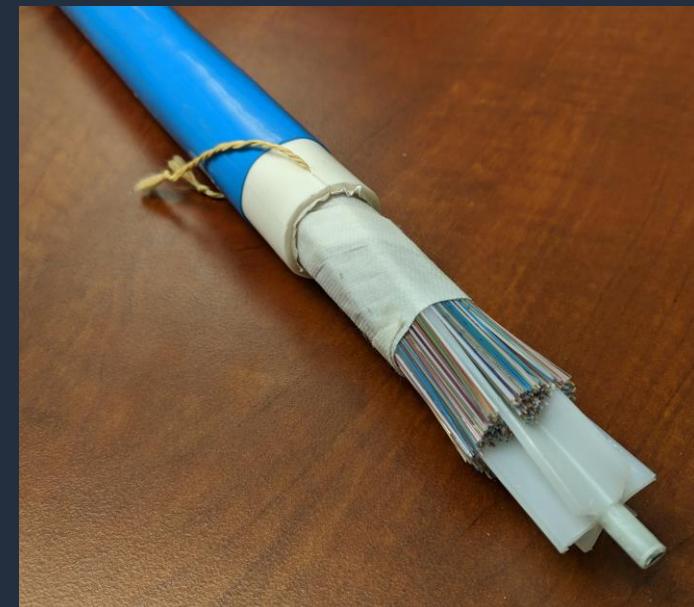
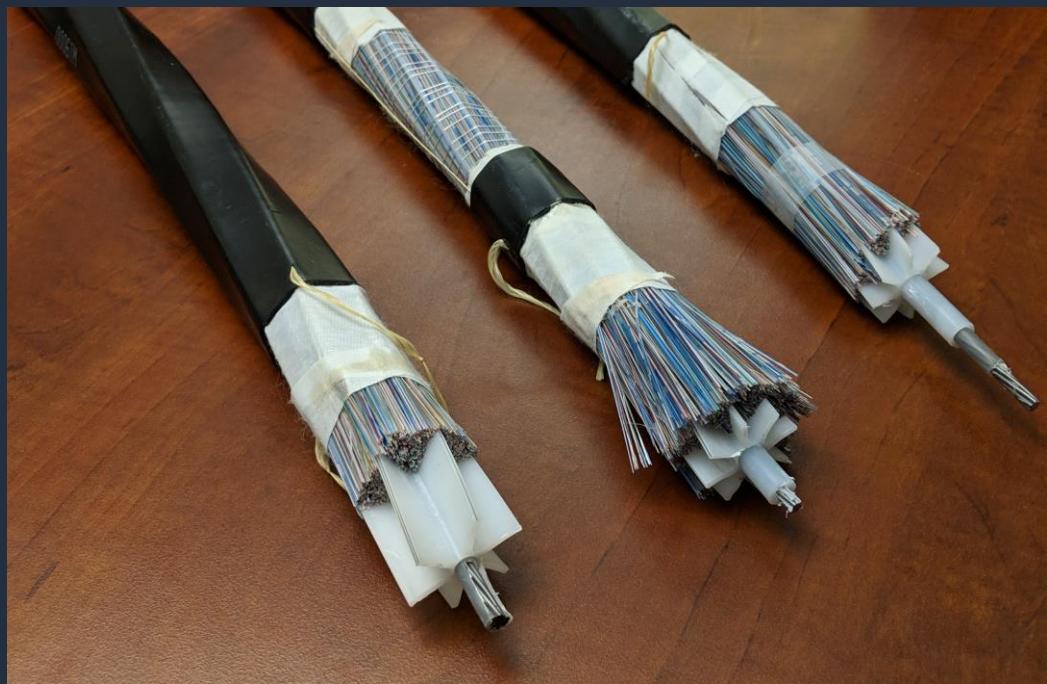


AWS Global Network



Intra & inter-AZ connectivity

- Dark fiber “spans”
 - Optimized for low-latency & physical diversity
- Amazon controlled infrastructure
- Geospatial coordinates
- Dense wavelength division multiplexing (DWDM)





Hawaiki

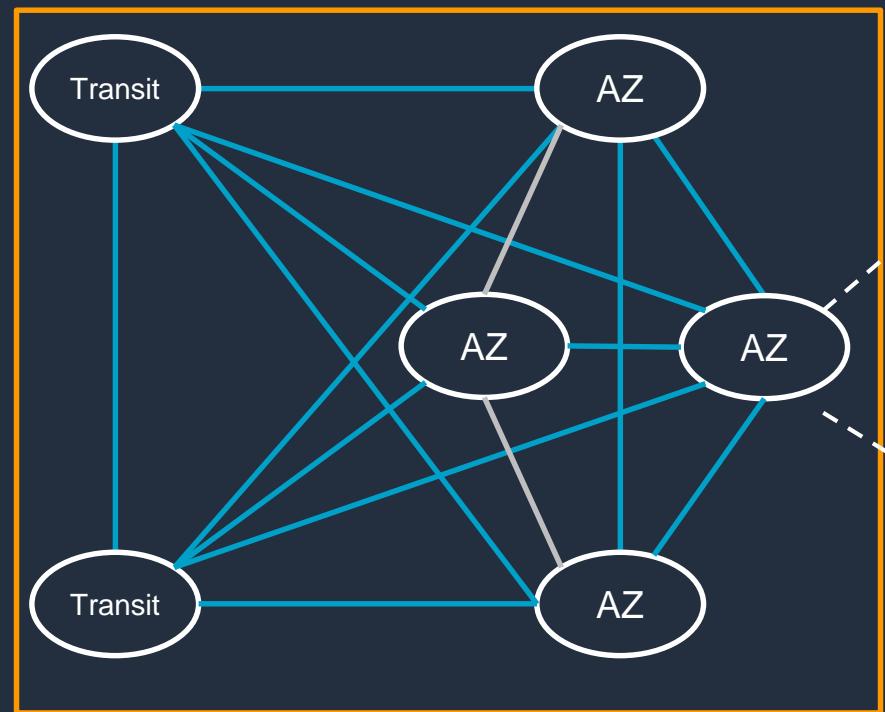




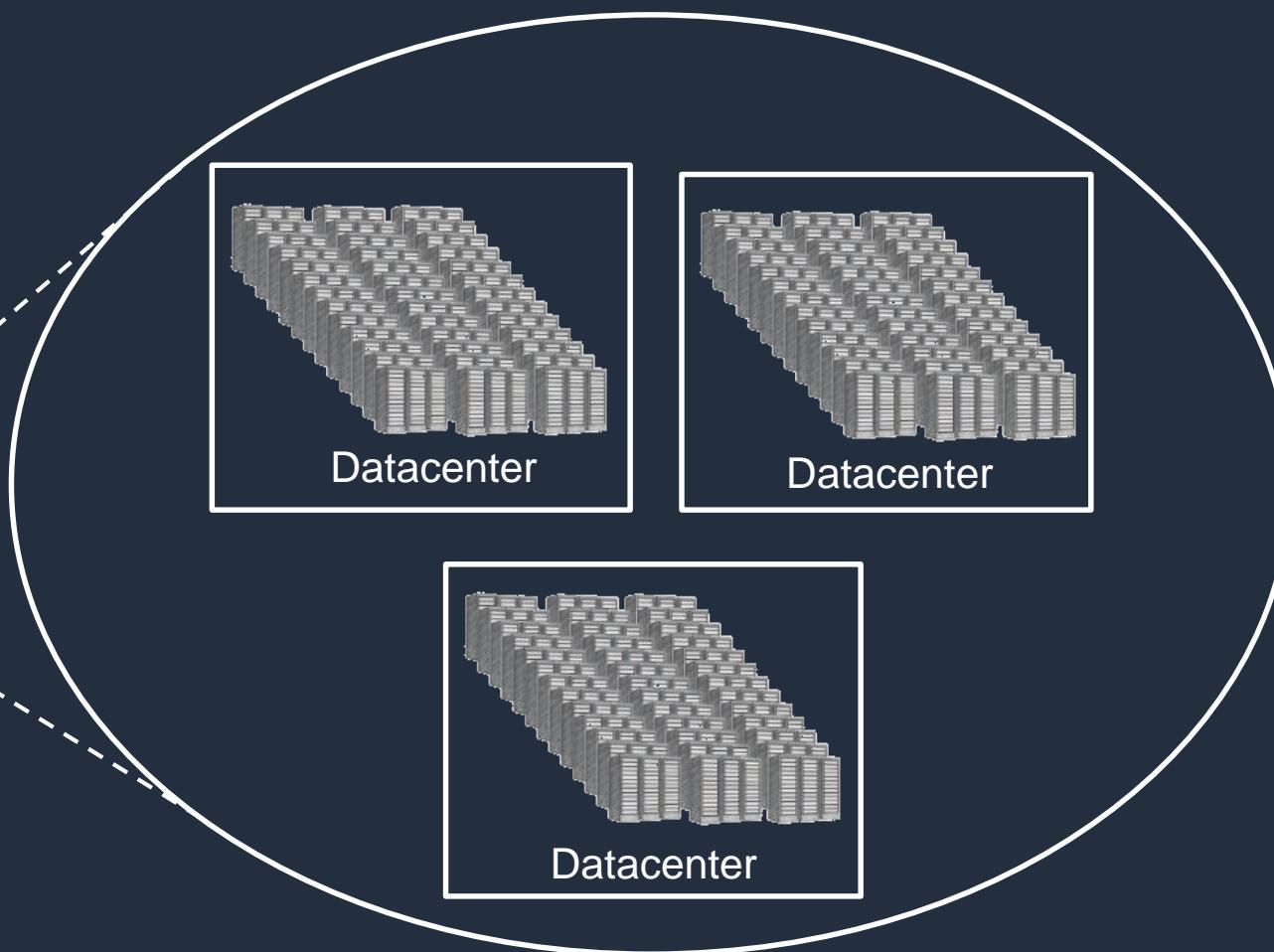
Bay to Bay Express

Availability Zones

AWS Region

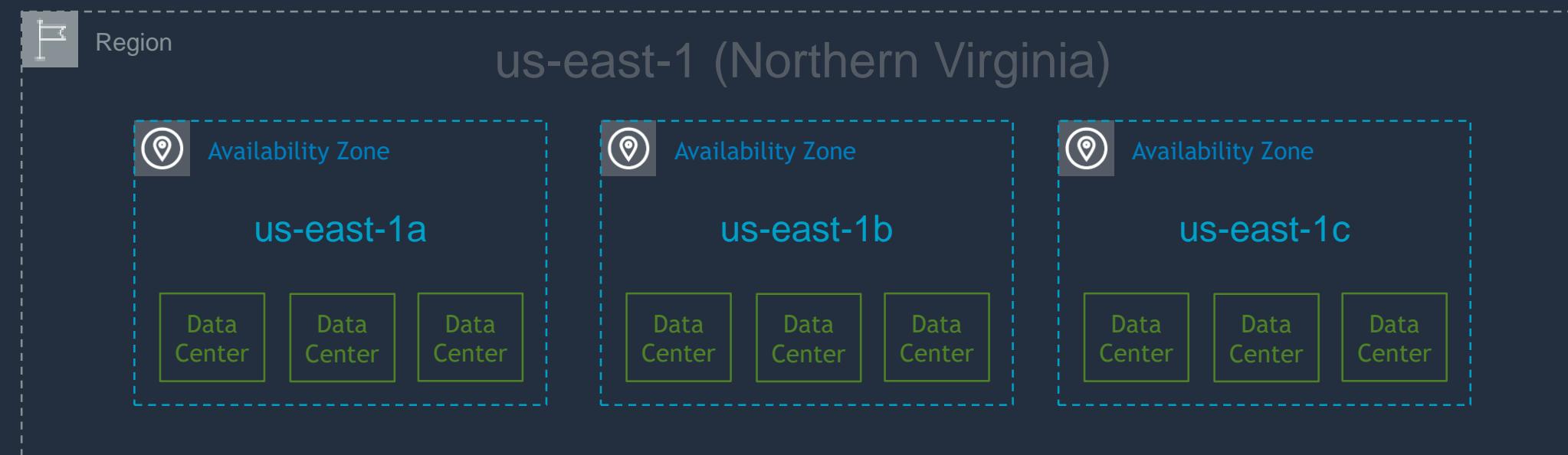


AWS Availability Zone (AZ)



Availability Zones

- A region is comprised of multiple Availability Zones (typically 3)
- Fully independent partitions on isolated fault lines, flood plains, and power grids
- Each AZ: redundant power and redundant dedicated network
- Each AZ: typically multiple data centers
- Between AZs: high throughput, low latency (<10ms) network
- Between AZs: physical separation < 100km (60mi)

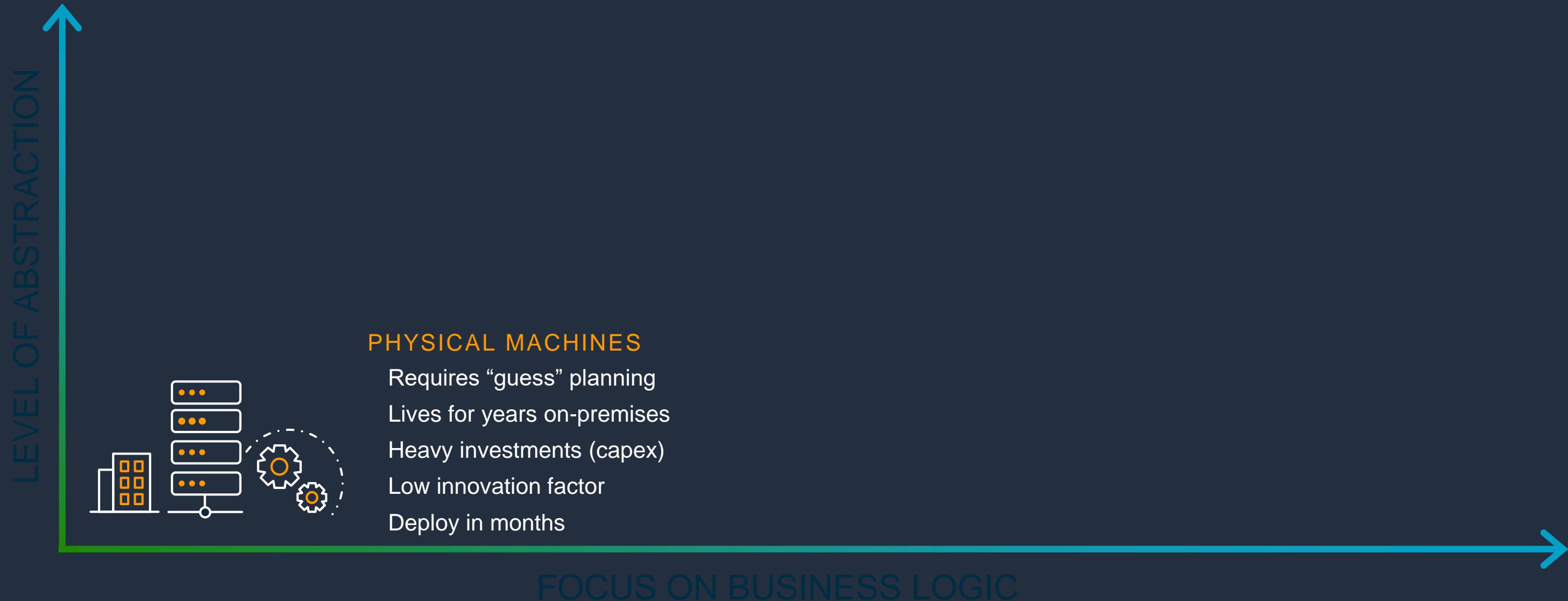


Quiz time ..

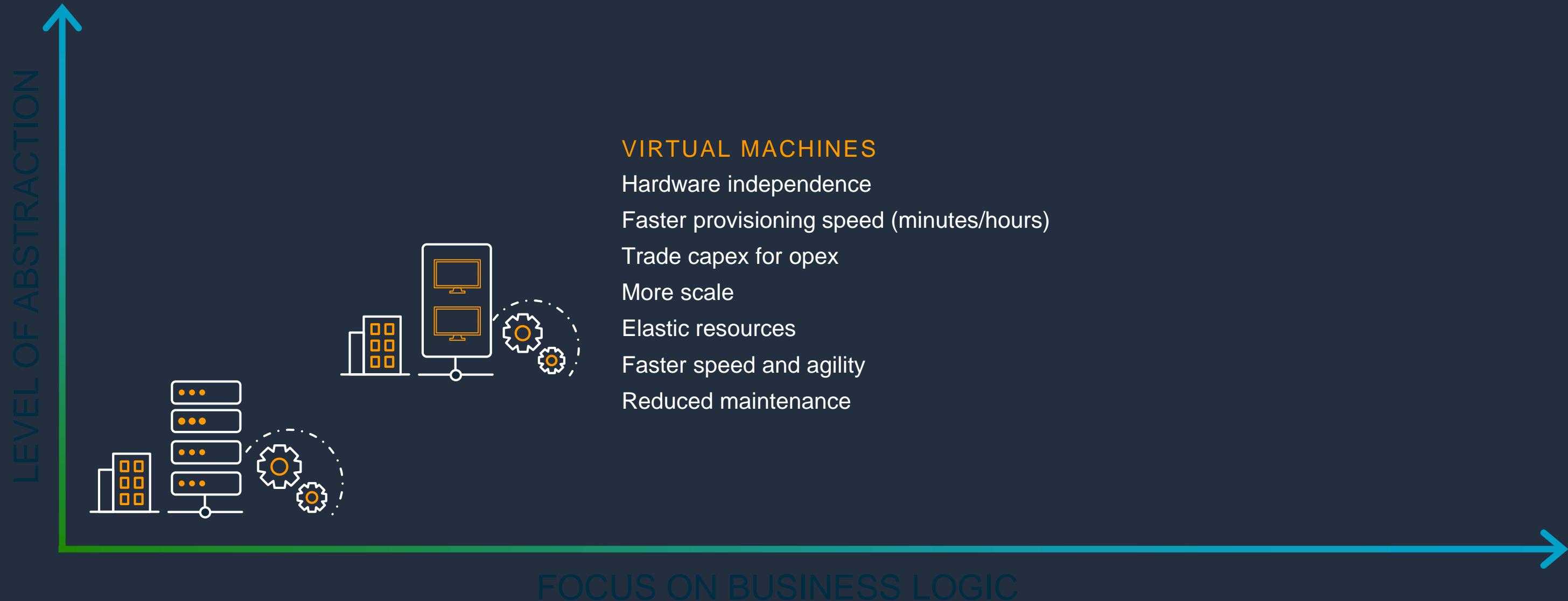
joinmyquiz.com

AWS Compute Services

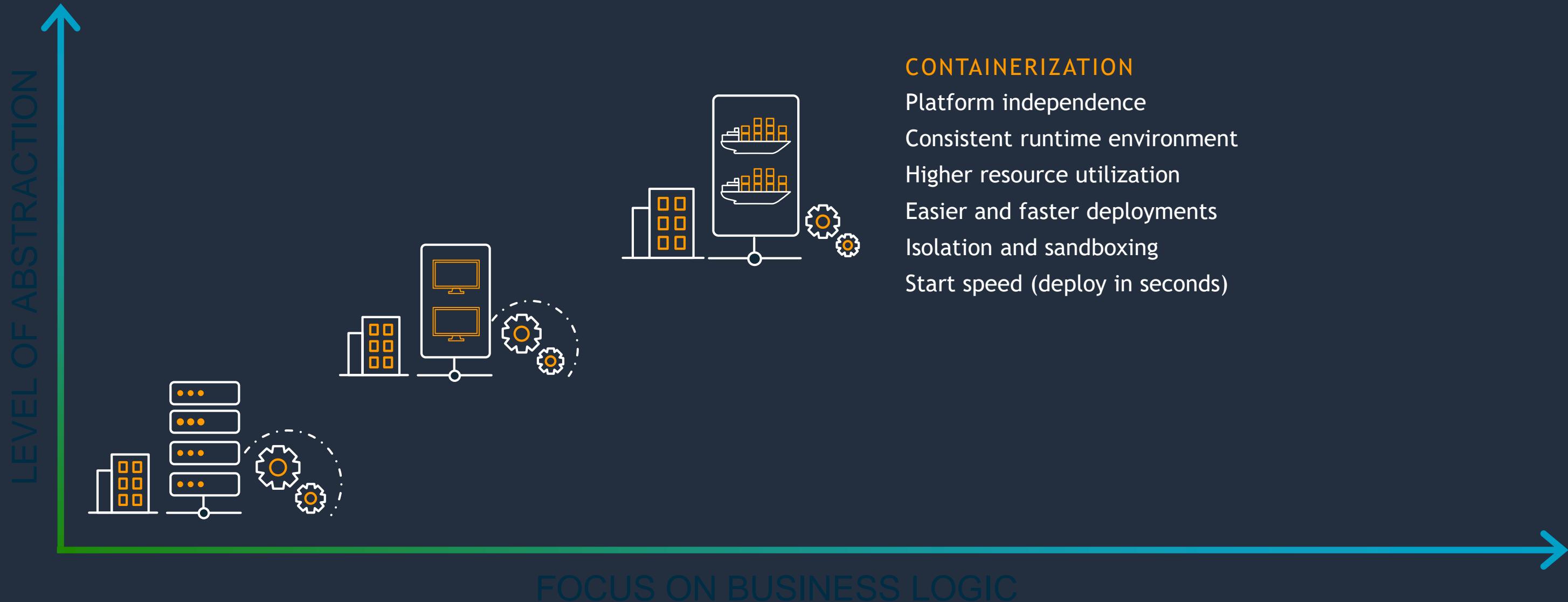
Computing evolution – A paradigm shift



Computing evolution – A paradigm shift



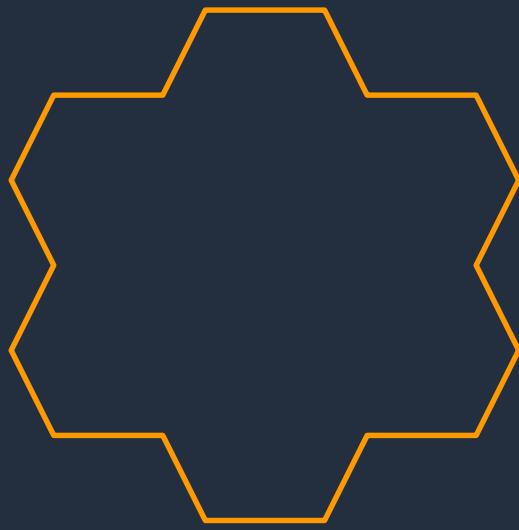
Computing evolution – A paradigm shift



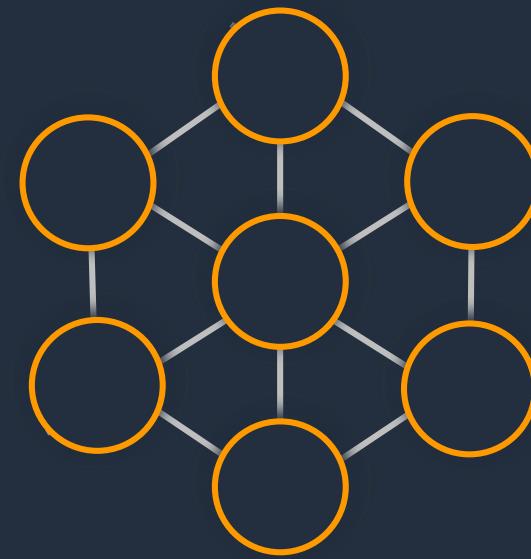
Computing evolution – A paradigm shift



Your application modernization starts with Micro services

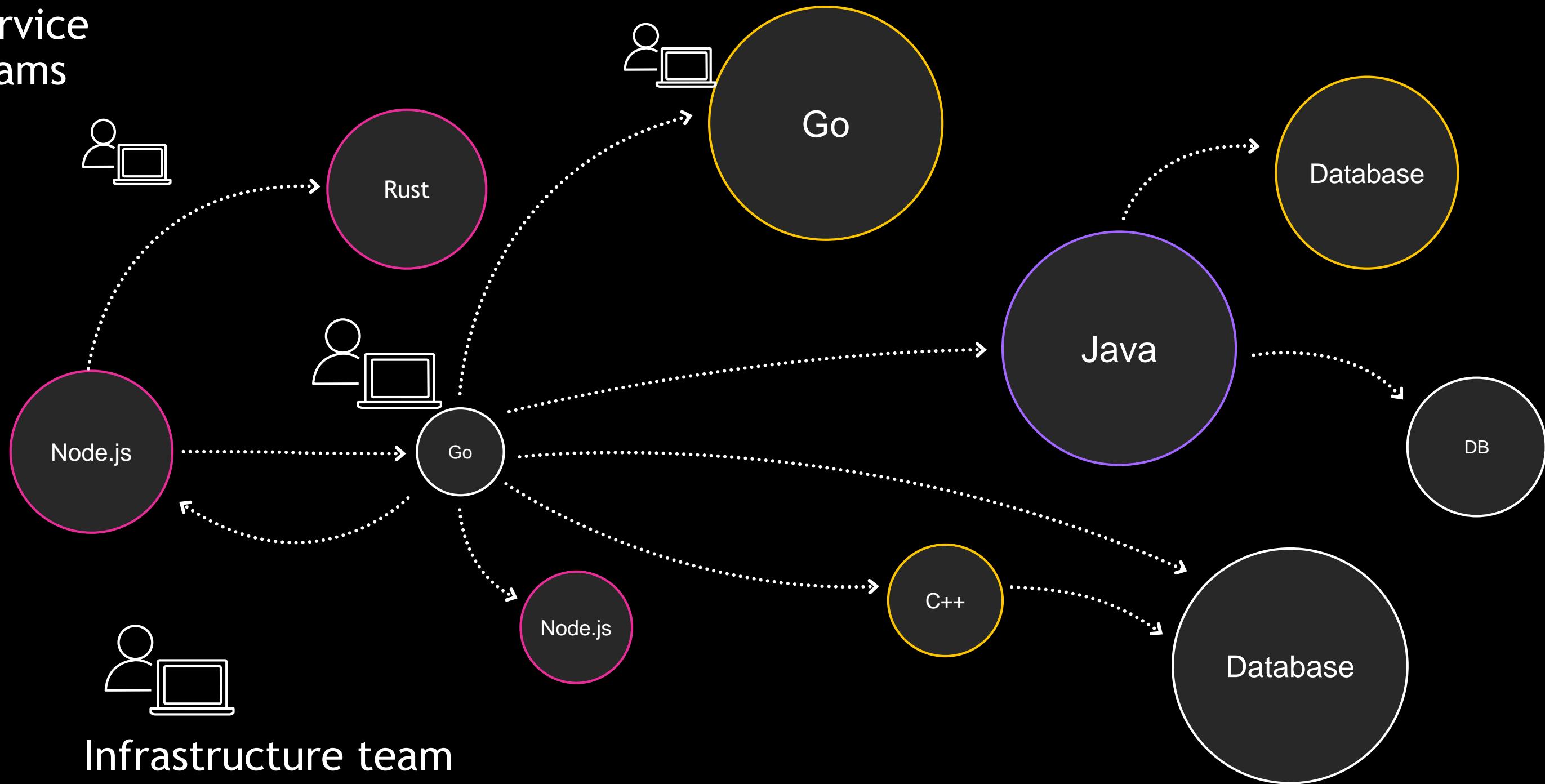


Monolith
Does everything

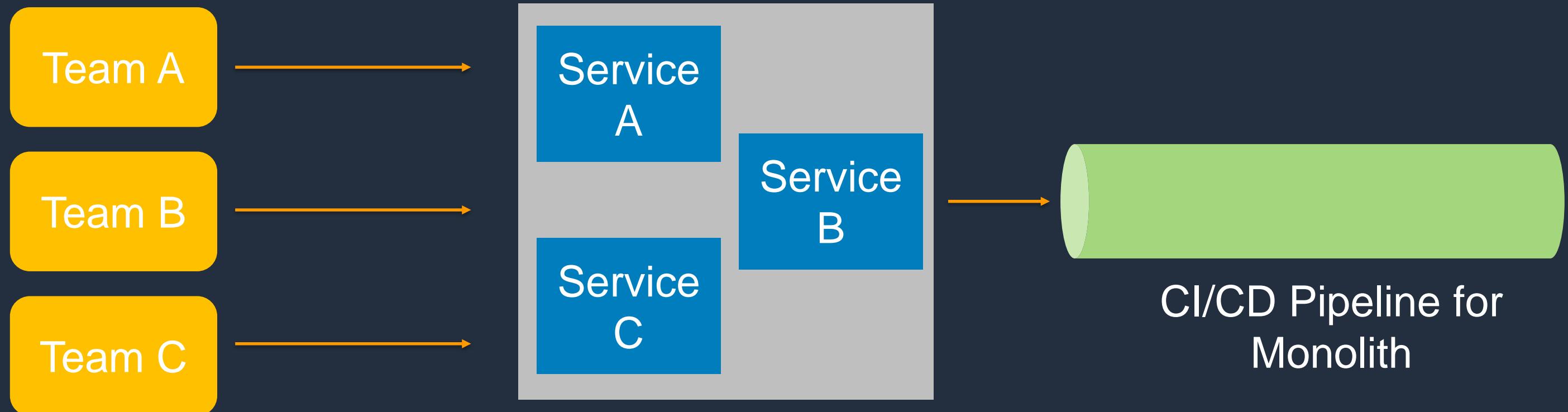


Microservices
Does one thing

Service teams

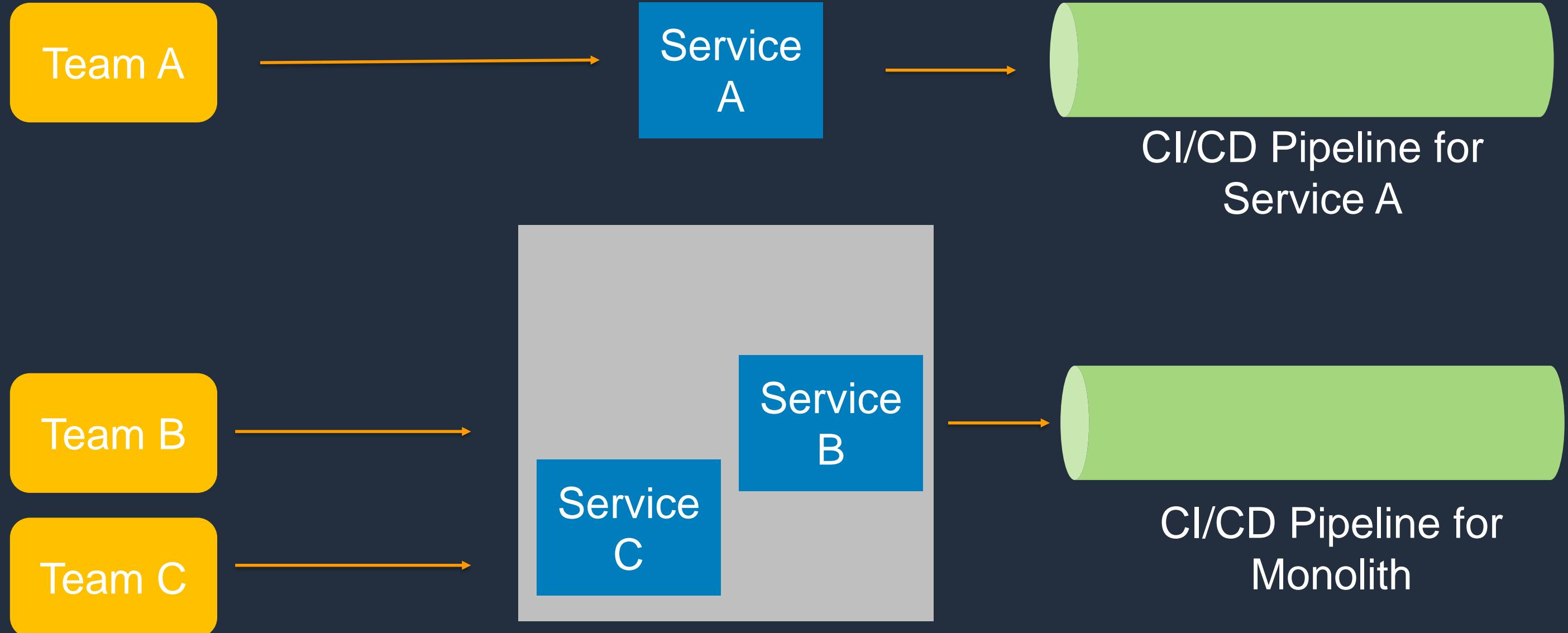


Mono to Micro



The Strangler Pattern

Mono to Micro



The Strangler Pattern

at Amazon.com

1000s of
teams



Micro-
services



CI/CD



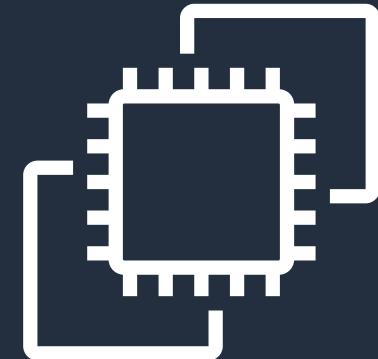
Serverless



>60 million deployments a year*



Choices for Compute



Amazon EC2

Virtual server instances
in the cloud



Amazon ECS, EKS, and Fargate

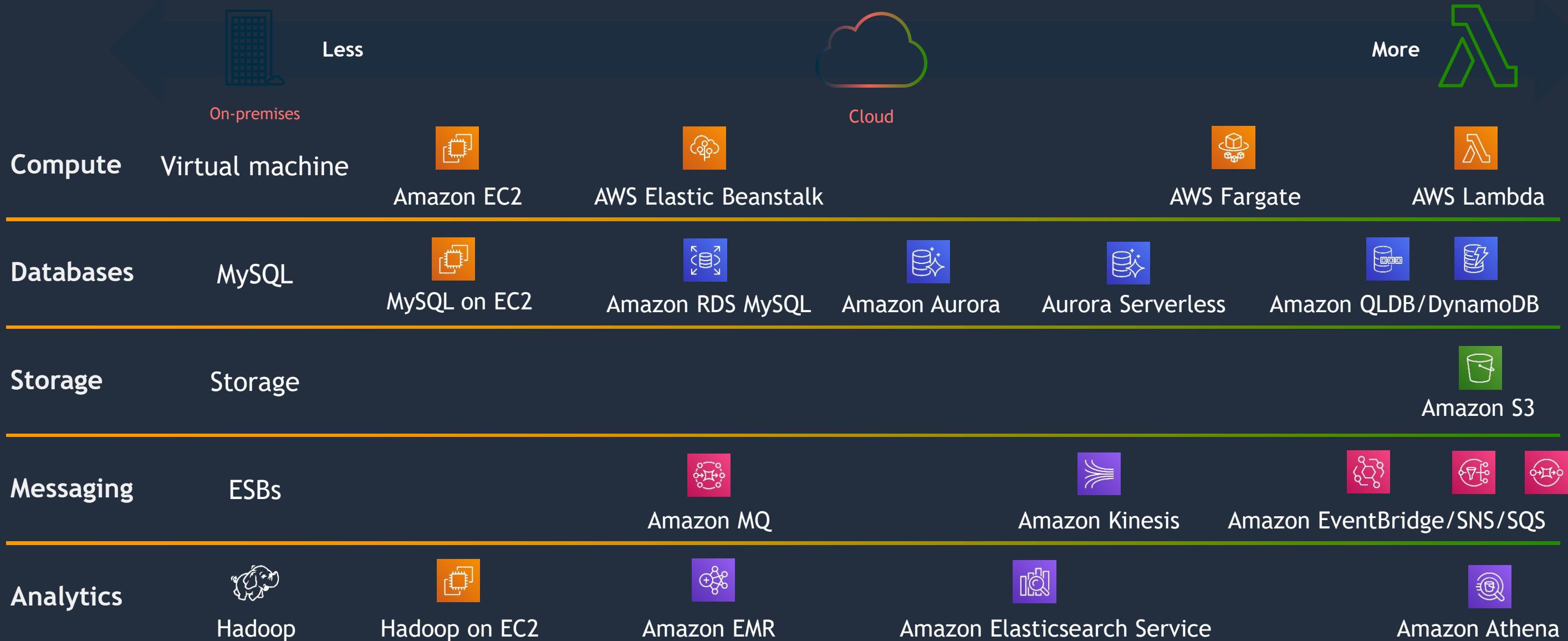
Container management service
for running
Docker on a managed
cluster of EC2



AWS Lambda

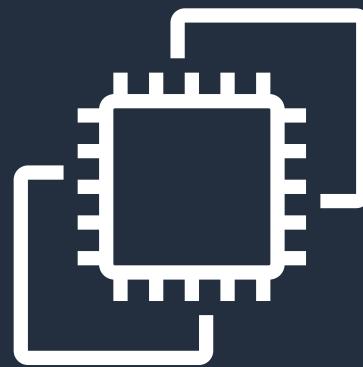
Serverless compute
for stateless code execution in
response to triggers

AWS operational responsibility models





Amazon EC2



Amazon EC2

Linux | Windows

Arm and x86 architectures

General purpose and workload optimized

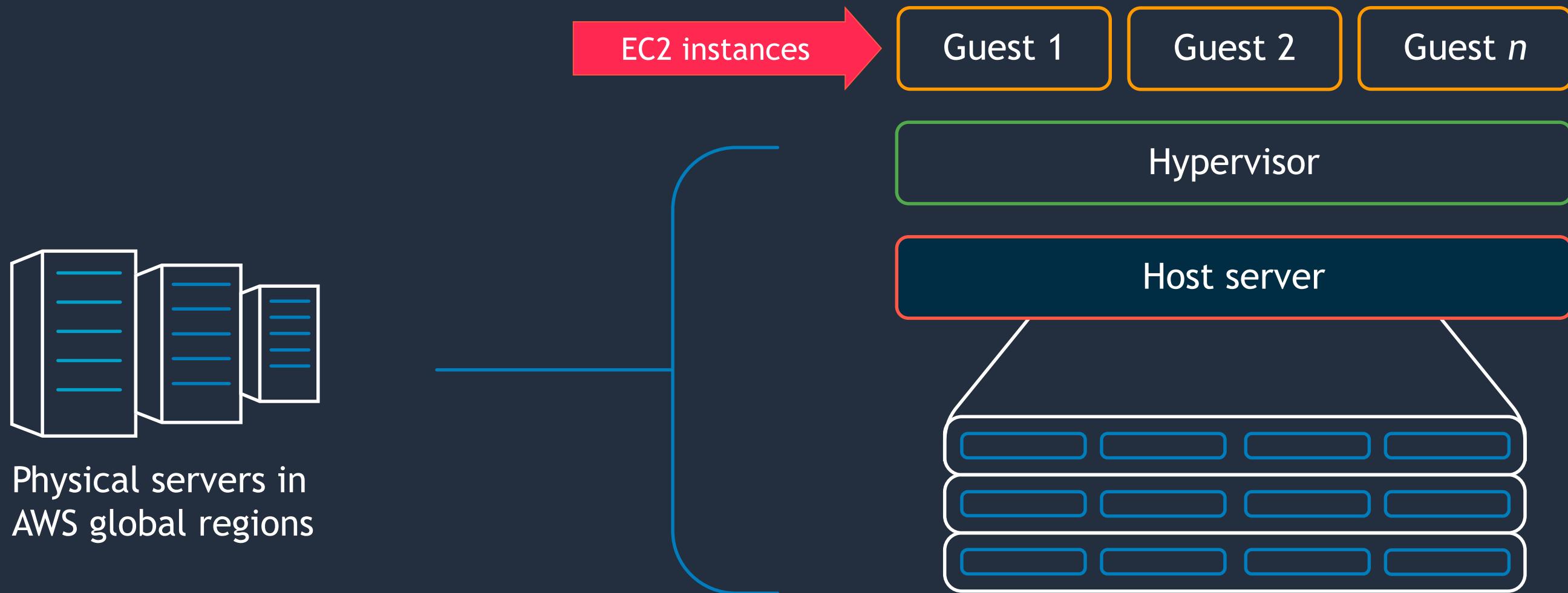
Bare metal, disk, networking capabilities

Packaged | Custom | Community AMIs

Multiple purchase options: On-demand, RI, Spot, Savings Plan

EC2 Design

EC2 Host Virtualization





Which hypervisor do we use?

Original host architecture: **Xen-based**

- Hypervisor consumed resources from the underlying host
- Limited optimization

AWS Nitro Hypervisor: **Custom KVM based hypervisor**

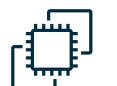
- AWS Nitro System (launched on Nov 2017)
- Less server resources used, more resources for the customer
- AWS optimized

Bare metal: **Direct access to processor and memory resources**

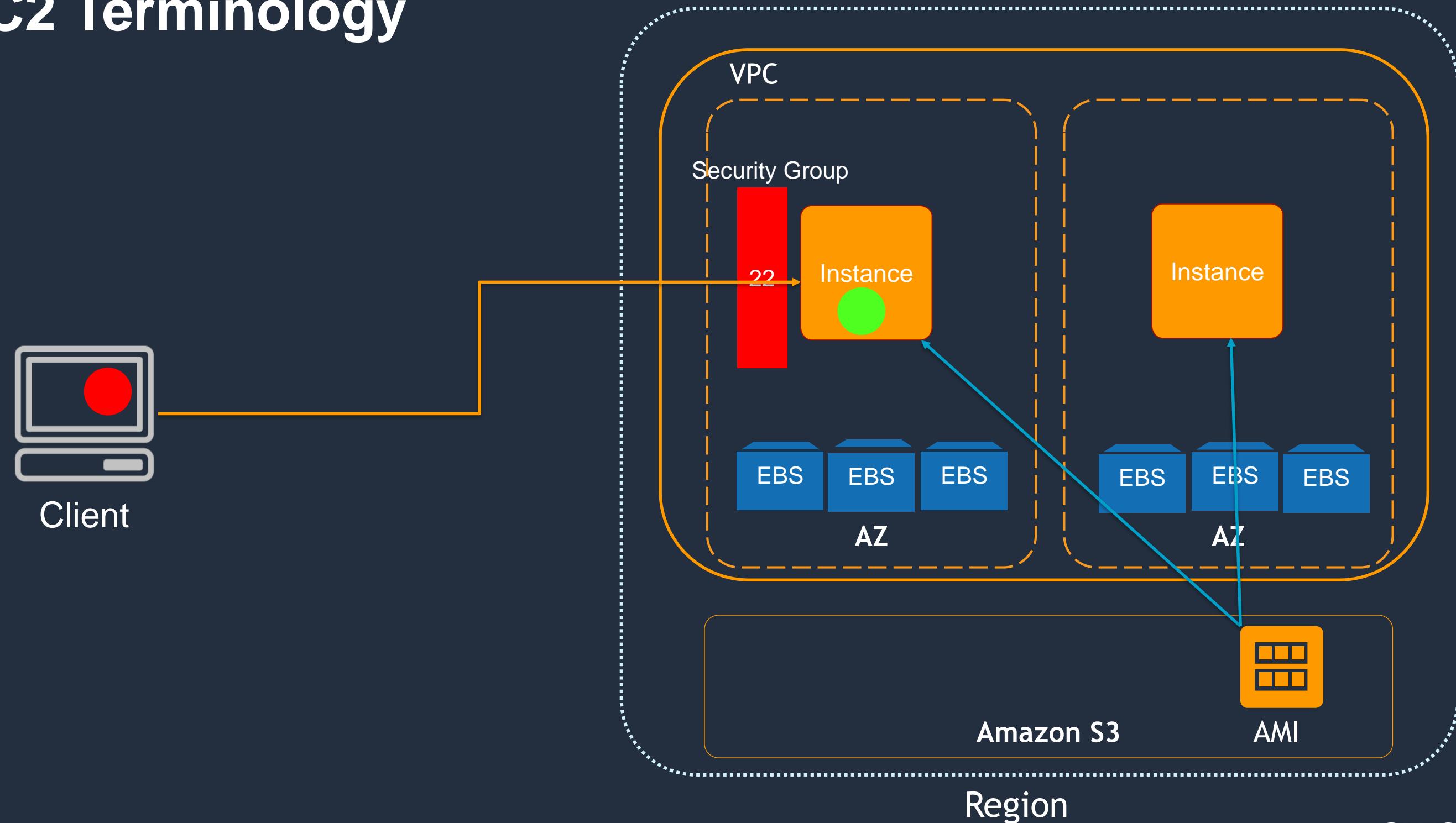
- Built on the AWS Nitro system
- Enables custom hypervisors and micro-VM runtimes

What do we need to create EC2 instance?

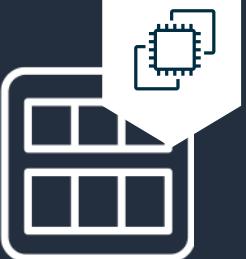
1. AWS account
2. AWS Datacenter - AWS Region (N. Virginia, Mumbai, Singapore etc)
3. Operating System - AMI (Amazon Machine Image)
4. CPU/RAM Configurations – Instance Types
5. Network – VPC/Subnets
6. Storage – EBS (Elastic Block Storage)
7. Firewall – Security Group
8. Login Credentials – SSH Key pair/Password



EC2 Terminology



What is an Amazon Machine Image (AMI)?



AMI

Provides the information required to launch an instance

Launch multiple instances from a single AMI

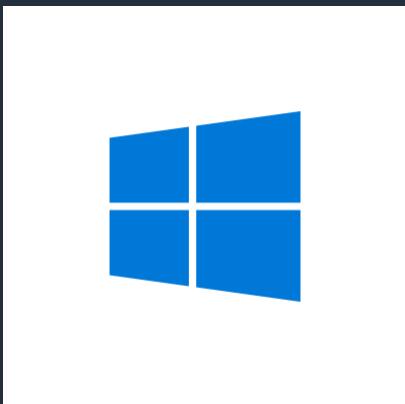
An AMI includes the following

- A template for the root volume (for example, operating system, applications)
- Launch permissions that control which AWS accounts can use the AMI
- Block device mapping that specifies volumes to attach to the instance



EC2 Operating Systems Supported

- Windows 2003R2/2008/2008R2/2012/2012R2/2016/2019
- Amazon Linux
- Debian
- Suse
- CentOS
- Red Hat Enterprise Linux
- Ubuntu



for more OSes see: <https://aws.amazon.com/marketplace/b/2649367011>



Choosing an AMI

AWS Console

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

Step 1: Choose an Amazon Machine Image (AMI)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs.

Quick Start

My AMIs AWS Marketplace Community AMIs

Free tier only

Image	Name	Type	Select	Architecture
Amazon Linux	Amazon Linux 2 AMI (HVM), SSD Volume Type - ami-04681a1dbd79675a5	Free tier eligible	Select	64-bit
Amazon Linux	Amazon Linux AMI 2018.03.0 (HVM), SSD Volume Type - ami-0ff8a9107f77f867	Free tier eligible	Select	64-bit
Red Hat	Red Hat Enterprise Linux 7.5 (HVM), SSD Volume Type - ami-6871a115	Free tier eligible	Select	64-bit

AWS Marketplace

aws marketplace

View Categories ▾ Migration Mapping Assistant Your Saved List Sell in AWS Marketplace Amazon Web Services Home Help

Operating Systems (336 results) showing 1 - 10

Categories

All Categories Infrastructure Software Operating Systems

Filters

Vendors

- clkwrk Ltd (84)
- Amazon Web Services (84)
- Center for Internet Security (20)
- Thinking Software, Inc. (13)
- CentOS.org (9)
- Technology Leadership Corporation (9)
- Plesk (9)
- Canonical Group Limited (8)
- SmartAMI (7)
- Cloud Linux (6)

Show more

Operating System

- + All Windows
- + All Linux/Unix

Software Pricing Plans

- Free (104)
- Hourly (212)
- Monthly (3)

CentOS 7 (x86_64) - with Updates HVM

★★★★★ (58) | Version 1805_01 | Sold by CentOS.org

This is the Official CentOS 7 x86_64 HVM image that has been built with a minimal profile, suitable for use in HVM instance types only. The image contains just enough packages...

Linux/Unix, CentOS 7 - 64-bit Amazon Machine Image (AMI)

CentOS 6 (x86_64) - with Updates HVM

★★★★★ (33) | Version 1805_01 | Sold by CentOS.org

This is the Official CentOS 6 x86_64 HVM image that has been built with a minimal profile. The image contains just enough packages to run within AWS, bring up an SSH Server...

Linux/Unix, CentOS 6 - 64-bit Amazon Machine Image (AMI)

Debian GNU/Linux 8 (Jessie)

★★★★★ (86) | Version 8.7 | Sold by Debian

Debian is a computer operating system composed of software packages released as free and open source software primarily under the GNU General Public License along with other...

Linux/Unix, Debian 8.6+1 - 64-bit Amazon Machine Image (AMI)

CentOS 6.5 (x86_64) - Release Media

★★★★★ (55) | Version 6.5 - 2013-12-01 | Sold by CentOS.org

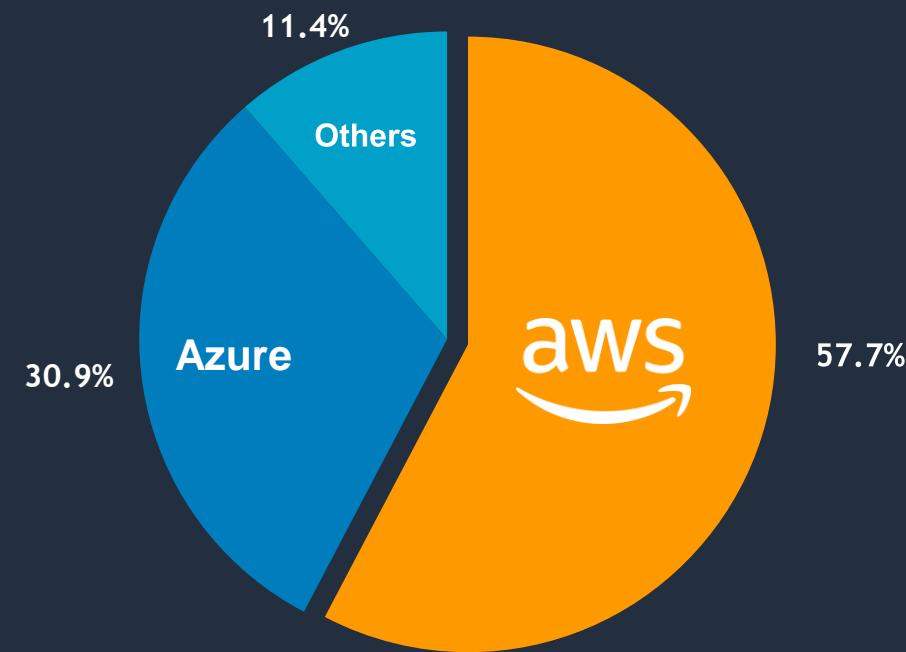
This is the Official CentOS 6.5 x86_64 image that has been built with a minimal profile. The image contains just enough packages to run within AWS, bring up an SSH Server...

Use the AMI ID to launch through the API or AWS Command Line Interface (AWS CLI)

```
aws ec2 run-instances --image-id ami-04681a1dbd79675a5 --instance-type c4.8xlarge --count 10 --key-name MyKey
```



Windows Licenses by Cloud Provider



Note: Includes Windows instances deployed in the public cloud IaaS market during 2017 Source: IDC estimates, 2018

IDC, Windows Server Operating Environment Market Update, Doc # US44217118, Aug 2018

https://d1.awsstatic.com/analyst-reports/IDC_Slide_WindowsonAWS_JM181015.pdf



Choose your processor and architecture



Intel® Xeon® Scalable
(Skylake) processor



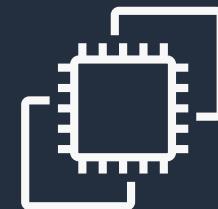
NVIDIA V100
Tensor Core GPUs



AMD EPYC processor



Amazon ARM based
Cloud Processor



FPGAs for custom
hardware acceleration

Right compute for the right application and workload



EC2 Instance Types

Instance generation

c5n.xlarge

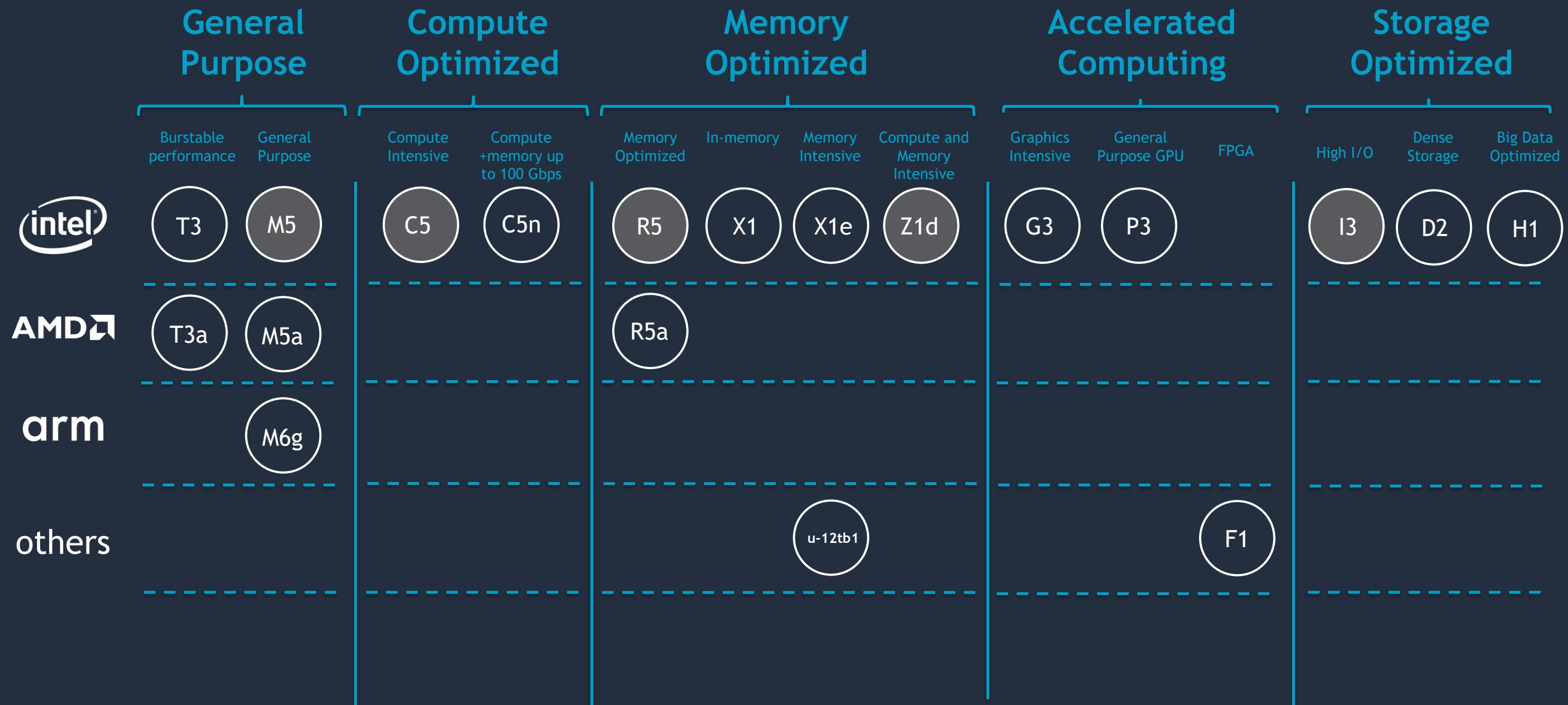
Instance
family

Attribute

Instance size



Instance Types





EC2 Options

Categories

- General purpose
- Burstable
- Compute intensive
- Memory intensive
- Storage (High I/O)
- Dense storage
- GPU compute
- Graphics intensive

Capabilities

- Choice of processor (AWS, Intel, AMD)

- Fast processors (up to 4.0 GHz)
- High memory footprint (up to 12 TiB)
- Instance storage (HDD and NVMe)
- Accelerated computing (GPUs and FPGA)


- Networking (up to 100 Gbps)
- Bare Metal
- Size (Nano to 32xlarge)

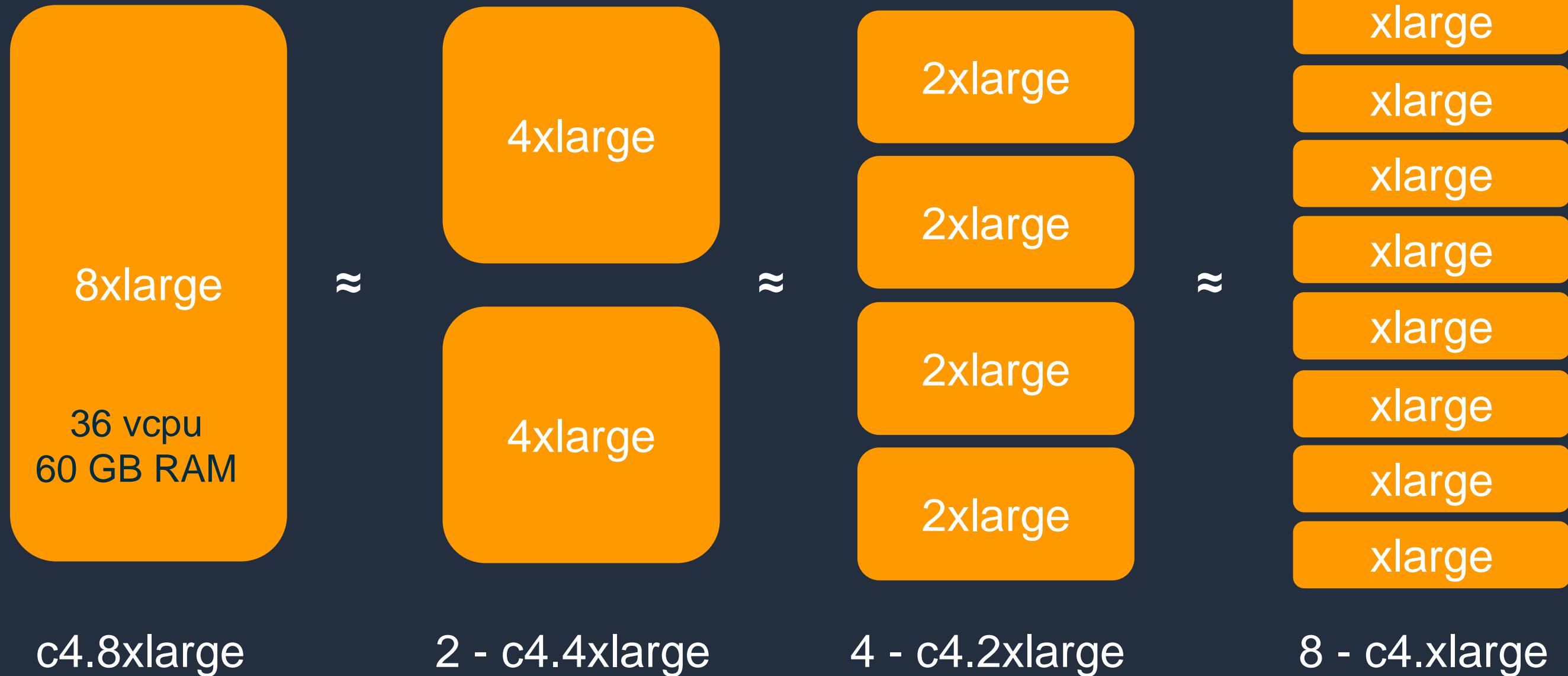
Options

- Elastic Block Store
- Elastic Graphics
- Elastic Inference

200+
instance types
for virtually
every workload
and business need



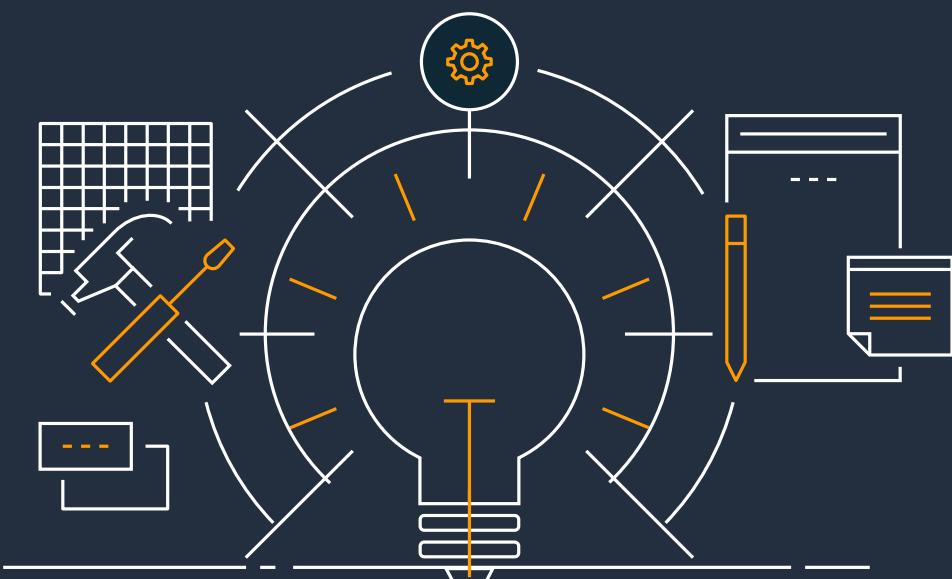
Instance sizing



NEW!

Introducing AWS Compute Optimizer

DELIVERS INSTANCE TYPE AND AUTO SCALING GROUP RECOMMENDATIONS



Applies insights from millions of workloads to make recommendations

Saves time comparing and selecting optimal compute resources for your workload

Recommends 3 optimal instance options for EC2 and EC2 Auto Scaling groups from 140+ instances from M, C, R, T, and X families

Available at no additional charge

Demo 1: Let's launch EC2 instance

1. Create SSH key pair
2. Launch EC2 instance
 1. Select AMI
 2. Select Instance Type
 3. Select Network
 4. Select Storage
 5. Add Tags
 6. Select SSH key pair
 7. Launch
3. Connect via SSH client
4. Install a sample web server
5. Access web server over internet

How to login/authenticate to EC2 instance?

EC2 key pairs

- Linux – SSH key pair for first-time host login
- Windows – Retrieve Administrator password

Standard SSH RSA key pair

- Public/Private Keys
- Private keys are not stored by AWS

AWS approach for providing **initial** access to a generic OS

- Secure
- Personalized
- Non-generic (NIST, PCI DSS)

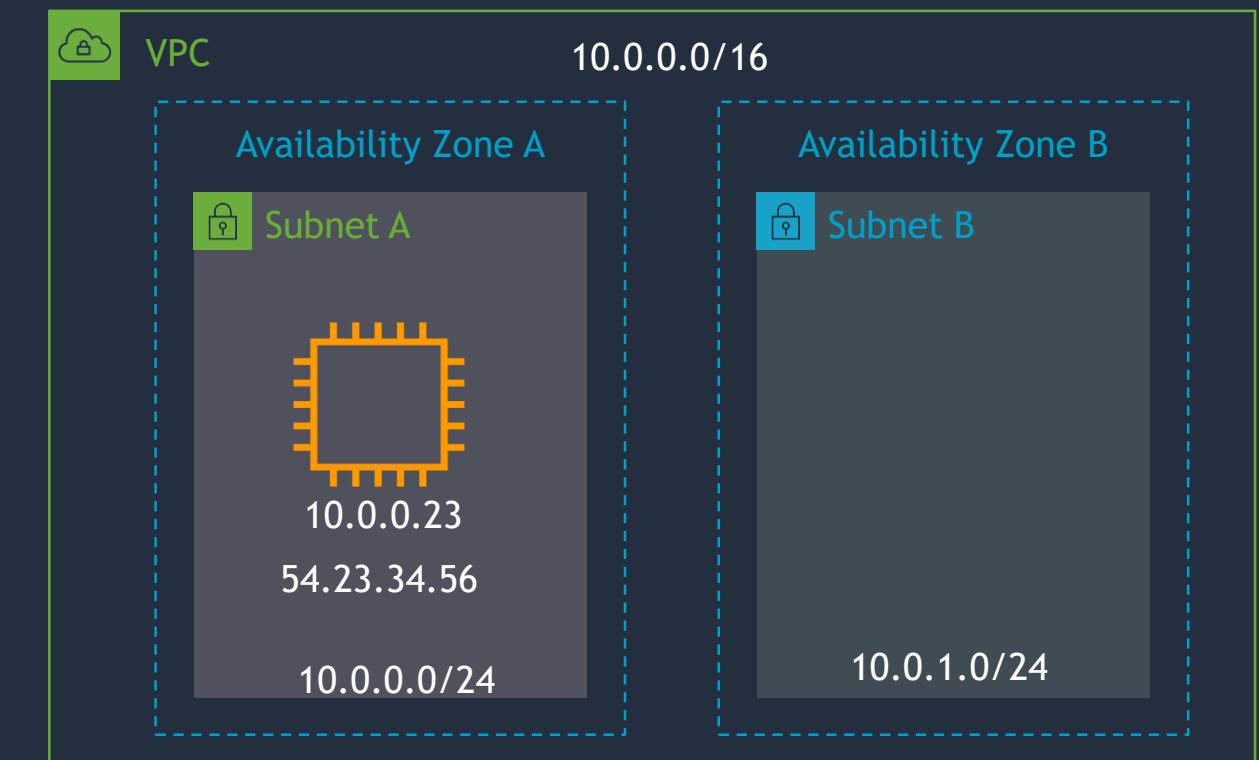
“Public Half” inserted by Amazon into each EC2 instance that you launch



“Private Half” downloaded to your desktop

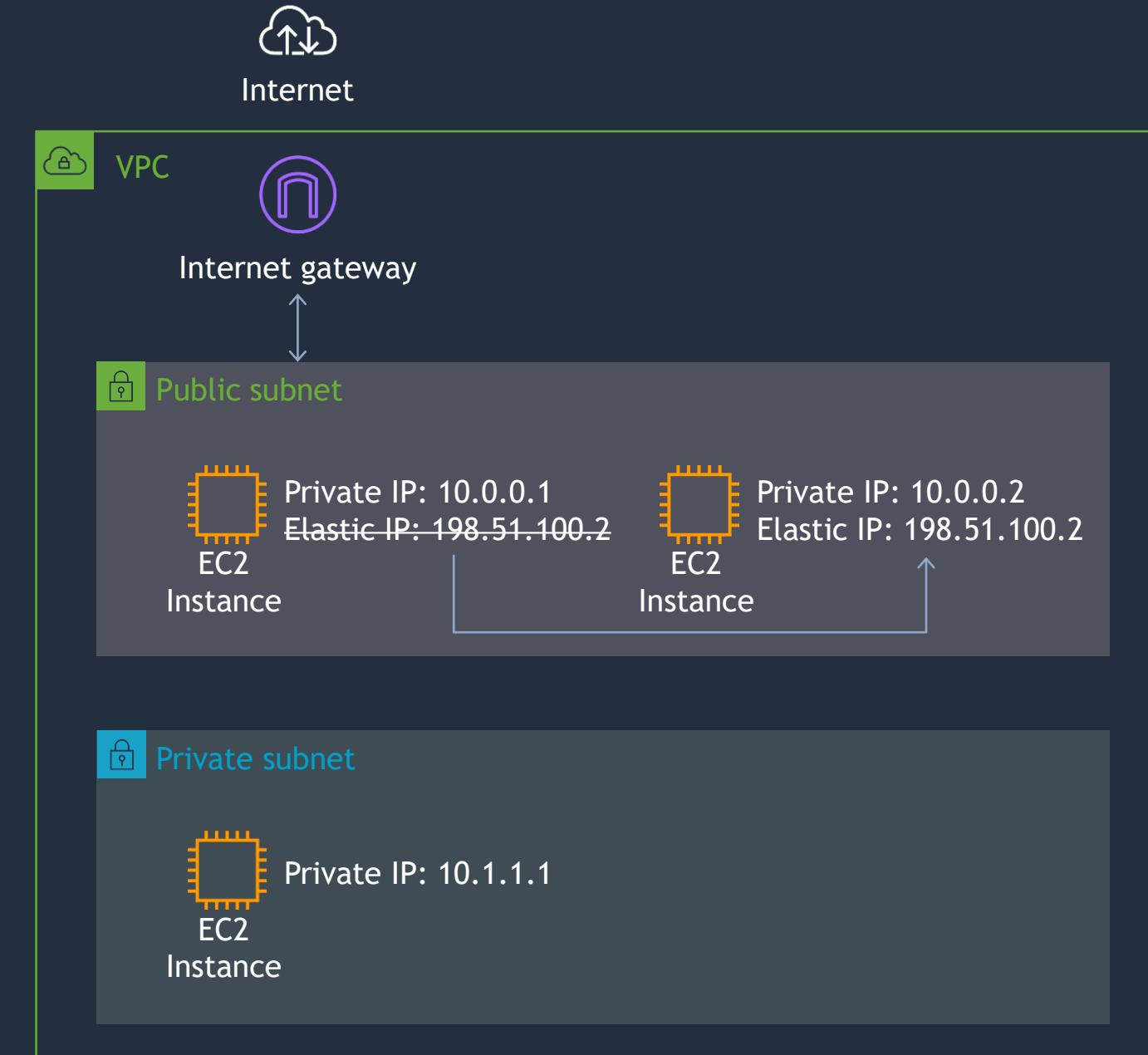
How EC2 gets IP address?

- EC2 is assigned Private IP from Subnet CIDR range
- Subnet also has attribute “Auto Assign Public IP” for instances
- Instance Public IP changes when you stop and start the instance
- You can also attach Elastic IP (Static Public IP) to EC2



Why to use Elastic IP Address?

- Static Public IPv4 address associated with your AWS account
- Can be remapped to another instance in your account
- Useful for redundancy

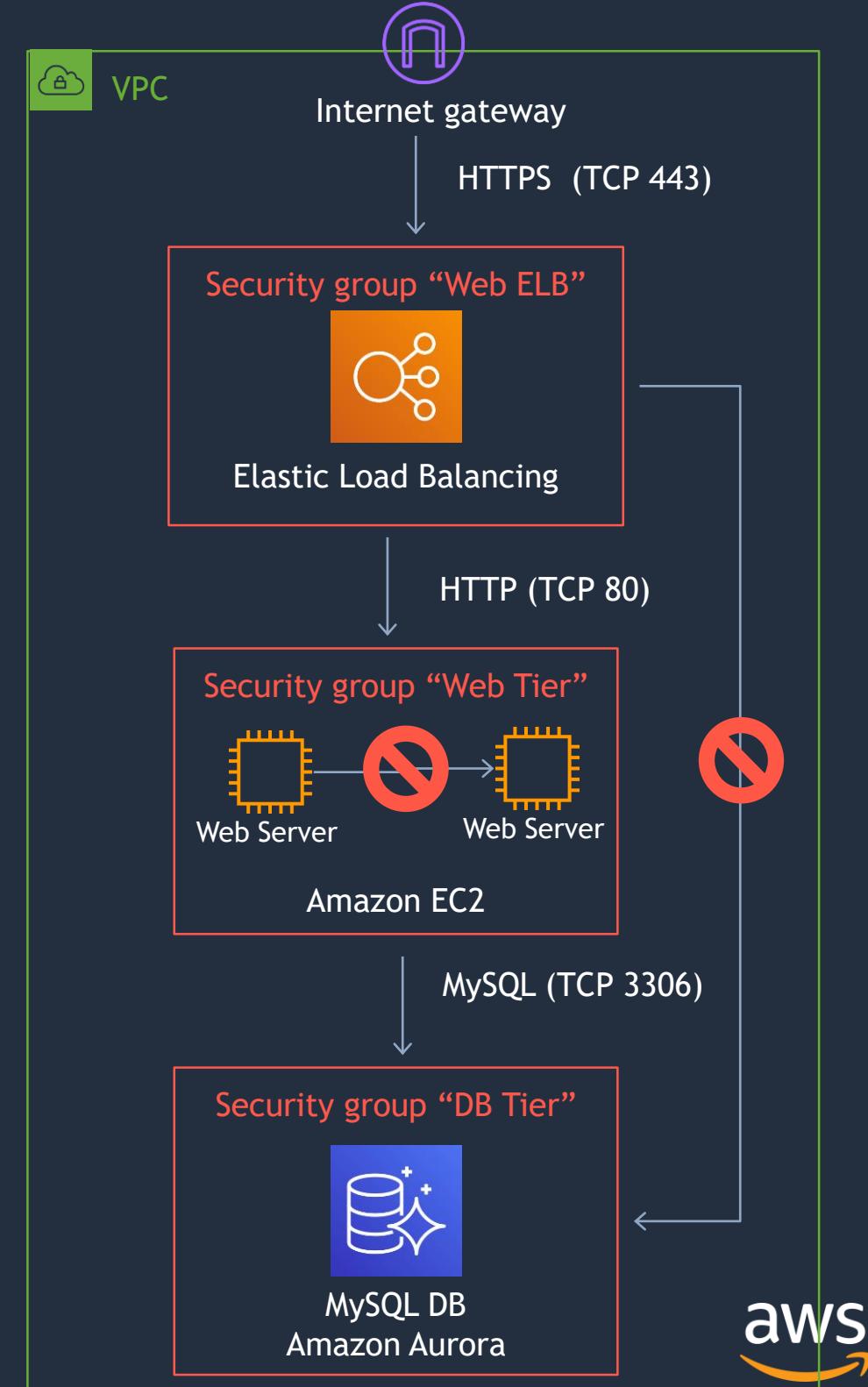


How to secure Network access ?

Security Group

- Virtual stateful firewall
- Inbound and Outbound customer defined rules
- Mandatory, all instances have an associated Security Group
- Only supports allow rules

Outbound rules			
Type	Protocol	Port range	Destination
All traffic	All	All	0.0.0.0/0



How to create custom EC2 image?

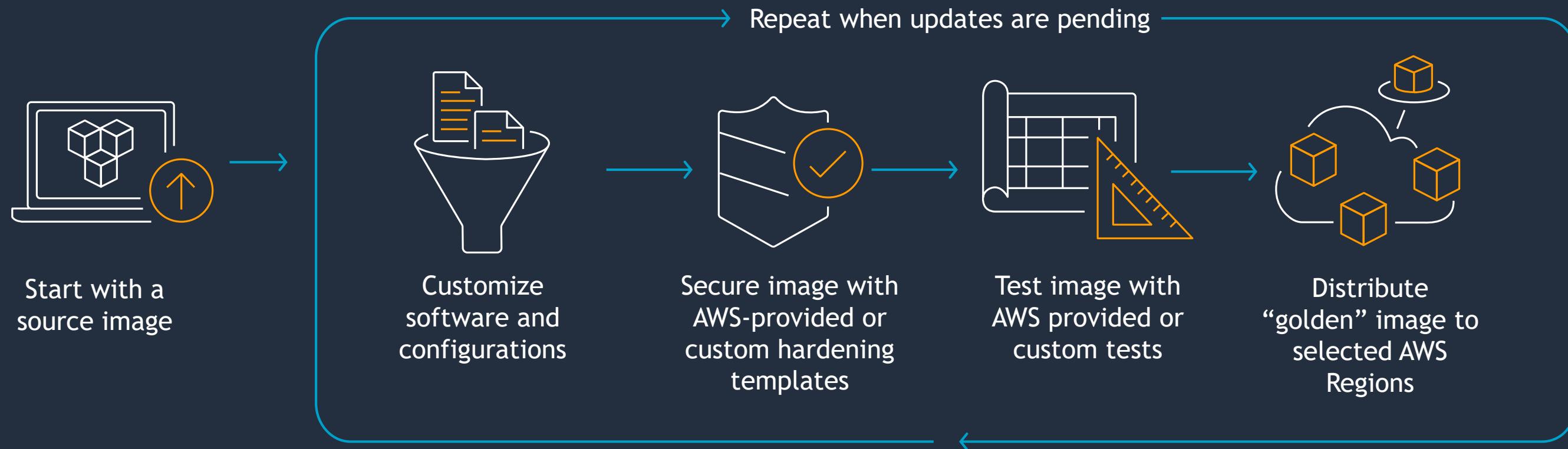
- Select the EC2 instance that you just launched
- Go to Actions -> Image -> Create Image
- Provide suitable image name e.g webserver-image
- Provide suitable description e.g Golden image for X application
- Create image
- Wait for up to 5 minutes
- Go to AMIs -> Select “Owned by me”
- You should see newly created image

Now you can launch new EC2 instances without having to install any applications. AMI contains everything.

NEW!

Introducing EC2 Image Builder

SIMPLIFIES THE CREATION, MAINTENANCE, VALIDATION, SHARING,
AND DEPLOYMENT OF LINUX OR WINDOWS SERVER IMAGES



What if I want to install applications during instance launch?

EC2 Userdata

- You can specify user data when launching an instance
- User data can be:
 - Linux script – executed by cloud-init
 - Windows batch or PowerShell scripts – executed by EC2Config service

Let's do it ..

- We want to launch EC2 instance which automatically installs web server on first boot up
- Also create sample index.html file

Sample userdata:

```
#!/bin/sh  
yum -y install httpd  
chkconfig httpd on  
service httpd start  
echo "<h1>This is my first webserver</h1>" > /var/www/html/index.html  
echo done
```

This time my code is there in S3

- I want to configure a web application or website on EC2
- The website code is stored in S3 which needs to be downloaded on EC2 for installation/configuration



This time my code is there in S3

- Create IAM Role for EC2 instance granting S3 bucket read access
- Use EC2 Userdata to download code and configure during boot up

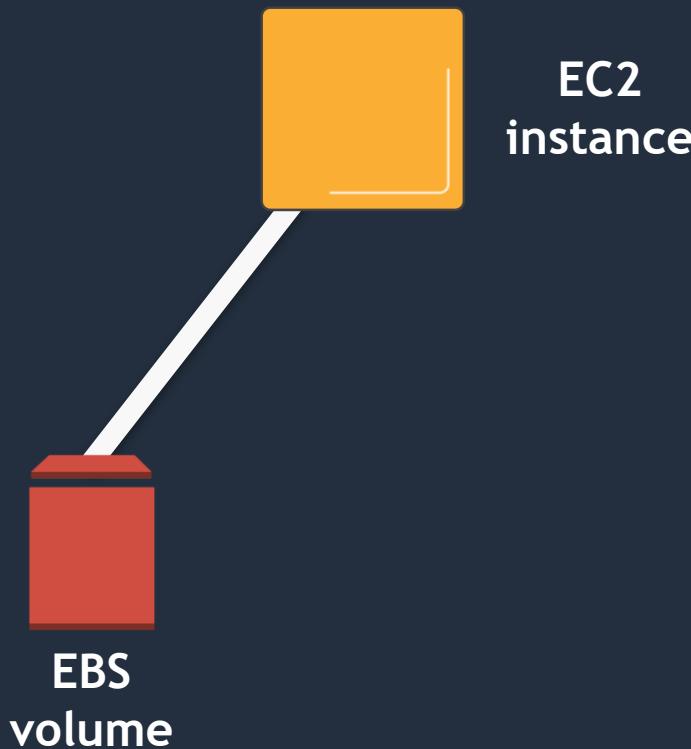
```
#!/bin/bash  
yum install httpd -y  
service httpd start  
chkconfig httpd on  
aws s3 cp s3://chetan-demo/car-repair.zip /home/ec2-user/  
unzip /home/ec2-user/car-repair.zip -d /var/www/html/  
echo successful
```

Instance Metadata

<http://169.254.169.254/latest/meta-data/> contains a wealth of info

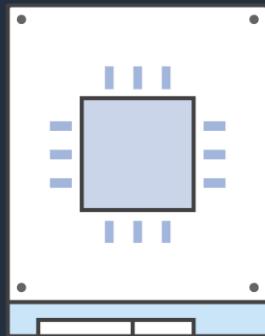
- ami-id
- ami-launch-index
- ami-manifest-path
- block-device-mapping/
- hostname
- instance-action
- ★ **instance-id**
- instance-type
- kernel-id
- local-hostname
- local-ipv4
- mac
- network/
- ★ **placement/availability-zone**
- profile
- public-hostname
- public-ipv4
- public-keys/

How do I add more storage to my EC2 instance?

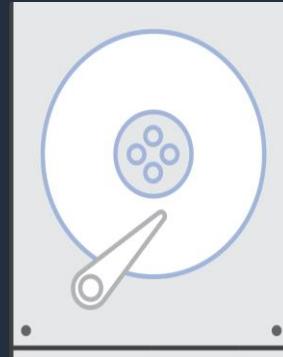


- Block storage as a service
- Create, attach volumes through an API
- Service accessed over the network

Amazon EBS use cases



SSD



HDD



Relational Databases

MySQL, SQL Server,
PostgreSQL, SAP,
Oracle



NoSQL Databases

Cassandra, MongoDB,
CouchDB



Big Data , Analytics

Kafka, Splunk,
Hadoop, Data
Warehousing



File / Media

CIFS/NFS,
Transcoding,
Encoding, Rendering

AWS EBS Features

Durable

Designed for 99.999 reliability

Redundant storage across multiple devices within an AZ

Performance

Low-latency SSD

Consistent I/O Performance

Stripe multiple volumes for higher I/O performance

Secure

Identity and Access Policies

Encryption

Scalable

Capacity when you need it

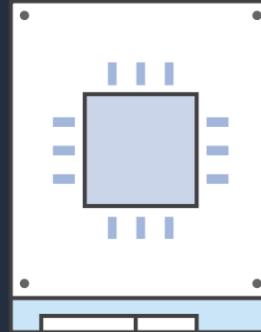
Easily scale up and down

Backup

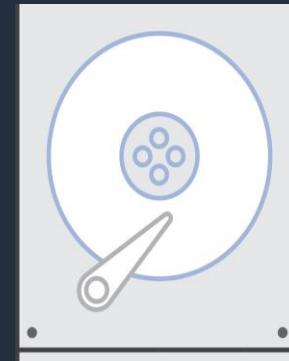
Point-in-time Snapshots

Copy snapshots across AZ and Regions

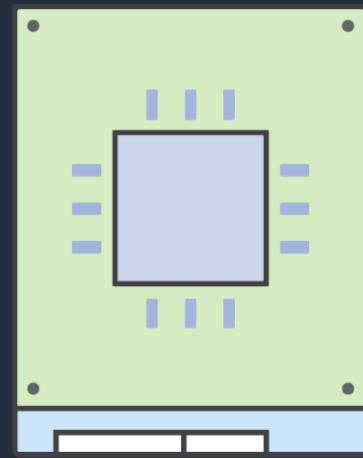
Amazon EBS volume types



SSD

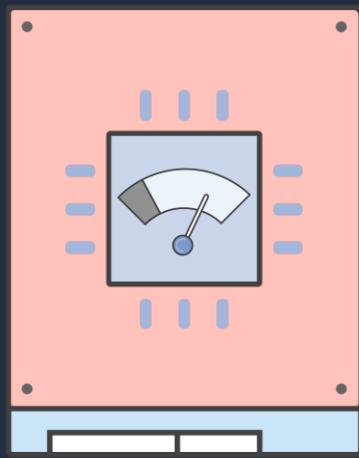


HDD



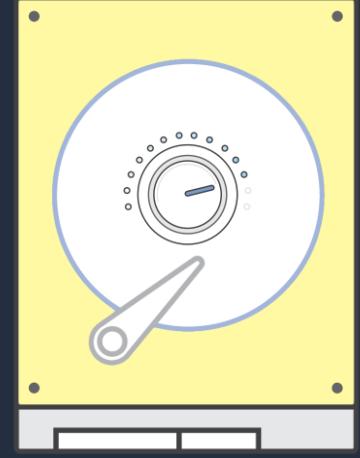
gp2

General Purpose
SSD



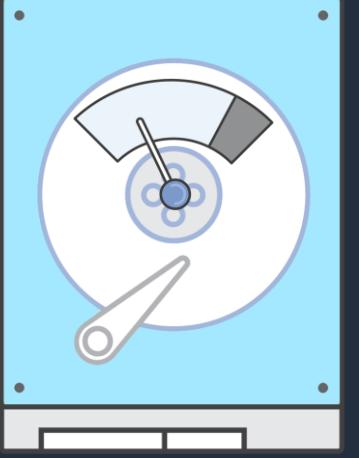
io1

Provisioned IOPS
SSD



st1

Throughput Optimized HDD



scl

Cold HDD

Let's add additional EBS disk to our EC2 instance

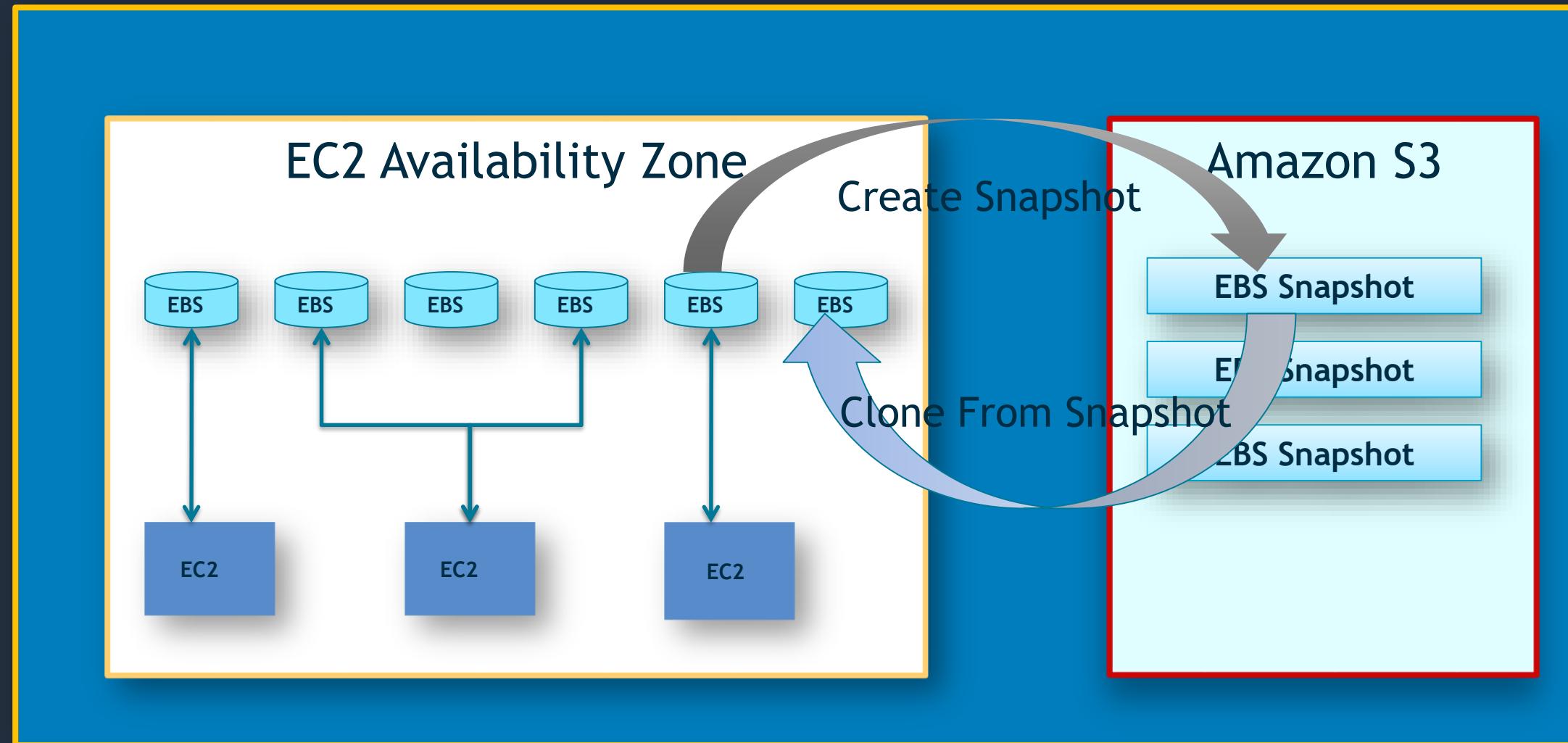
1. Attach additional EBS volume to EC2
2. Create snapshot of EBS Volume

Command to format and mount EBS (Linux)

```
$ lsblk  
$ sudo mkfs -t ext4 /dev/xvdf  
$ sudo mkdir /data-volume  
$ sudo mount /dev/xvdf /data-volume  
$ df
```

How do I take backup of my EBS disk?

EBS Snapshot

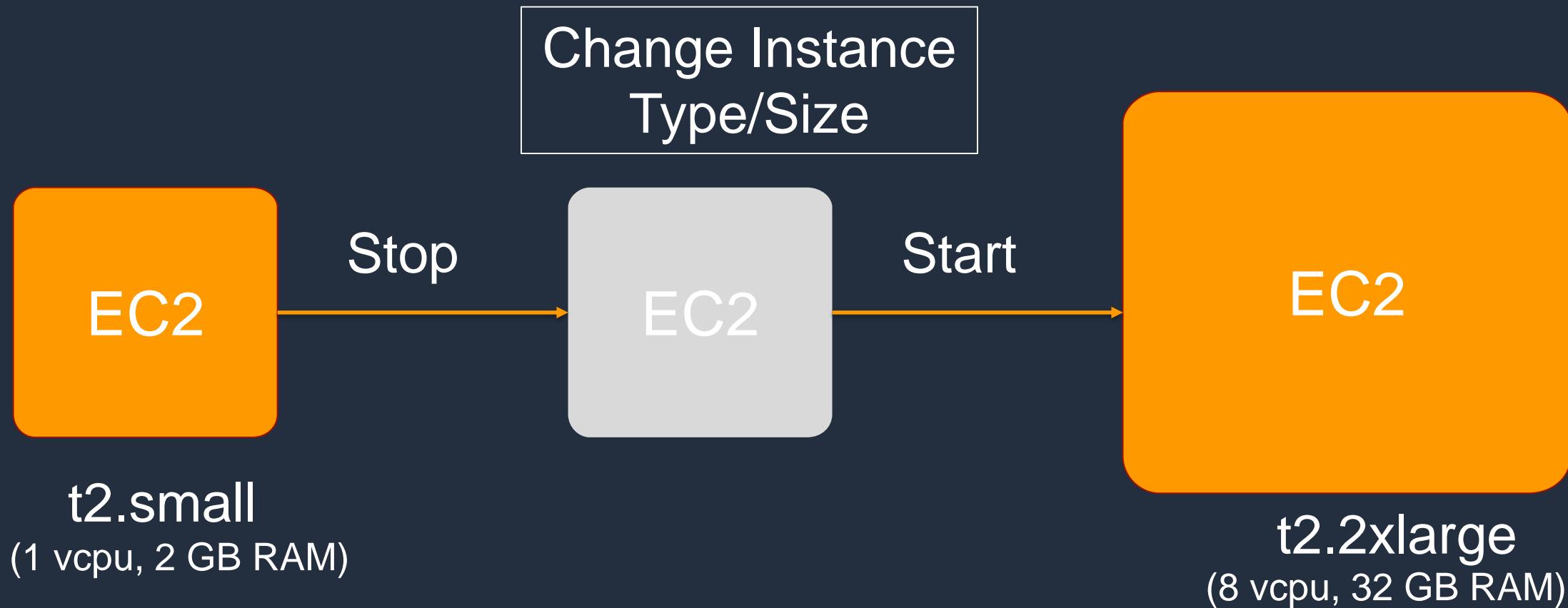


Creating EBS snapshot ..

1. Select EBS volume
2. Create snapshot

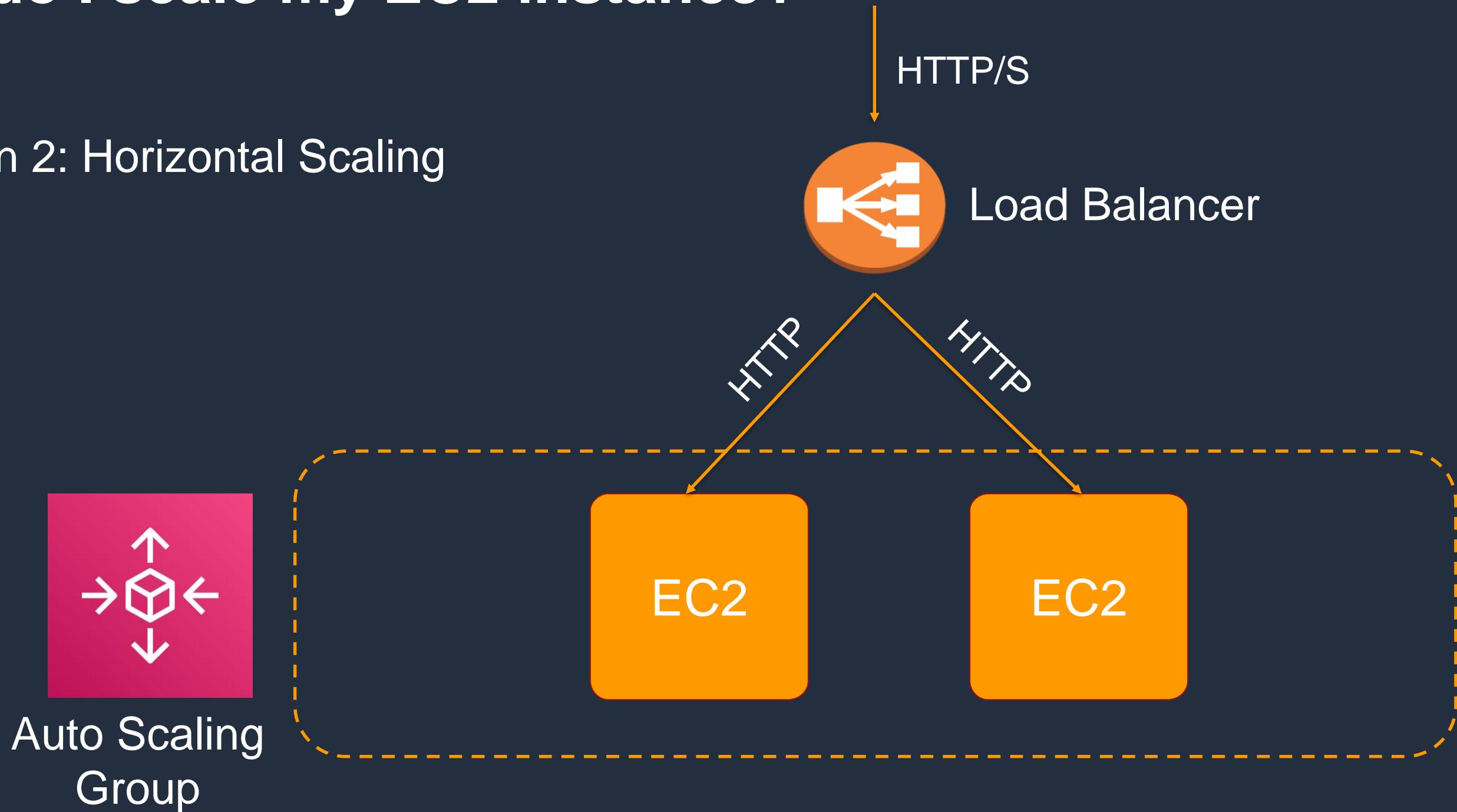
How do I scale my EC2 instance?

Option 1: Vertical Scaling



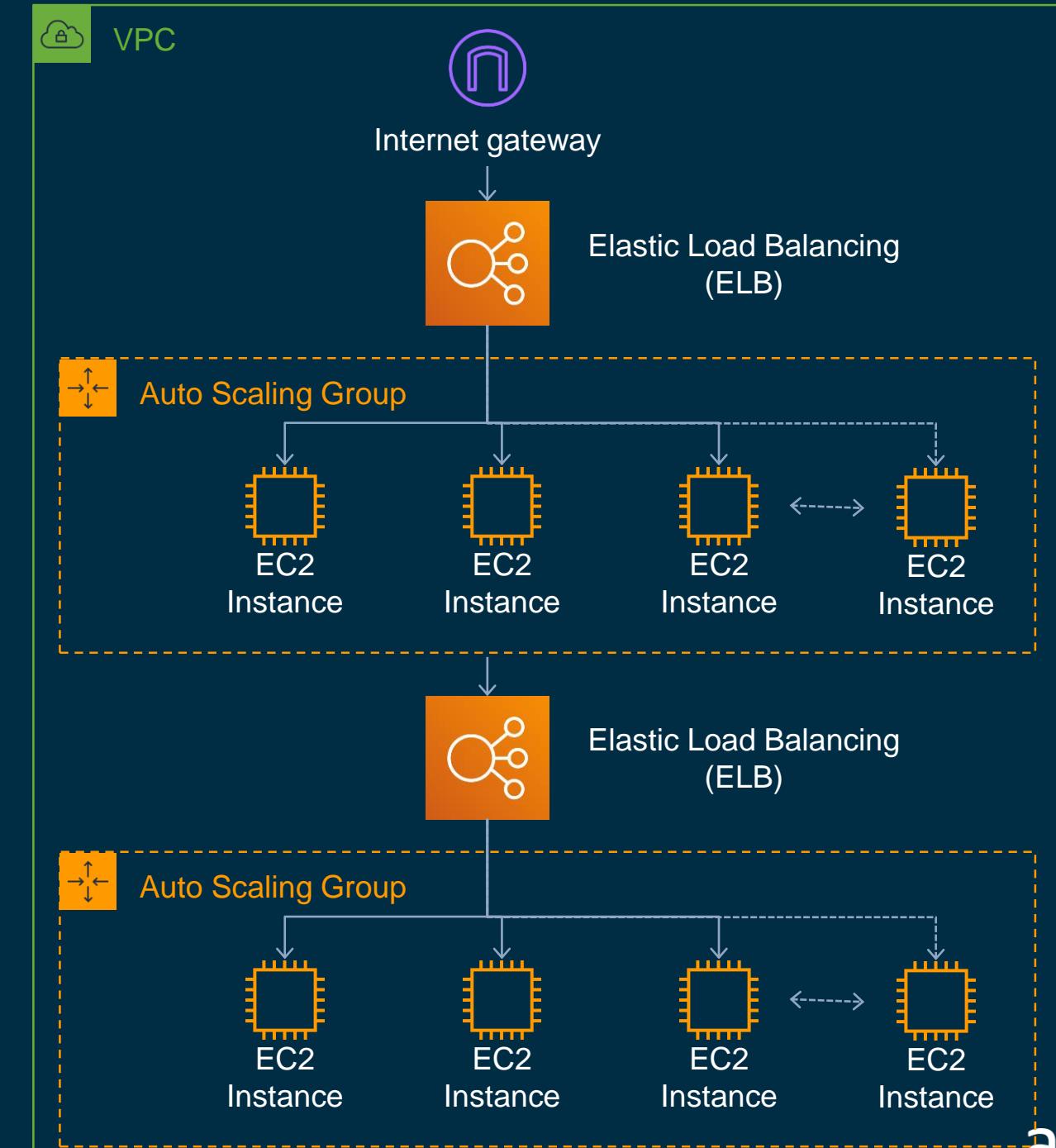
How do I scale my EC2 instance?

Option 2: Horizontal Scaling



Elastic Load Balancing

- Distributes incoming application or network traffic across multiple targets
 - EC2 instances
 - Containers
 - IP address
- Across Availability Zones
- Scales automatically
- Auto Scaling Groups can add or remove instances as required
- Automatically register to the Load Balancer



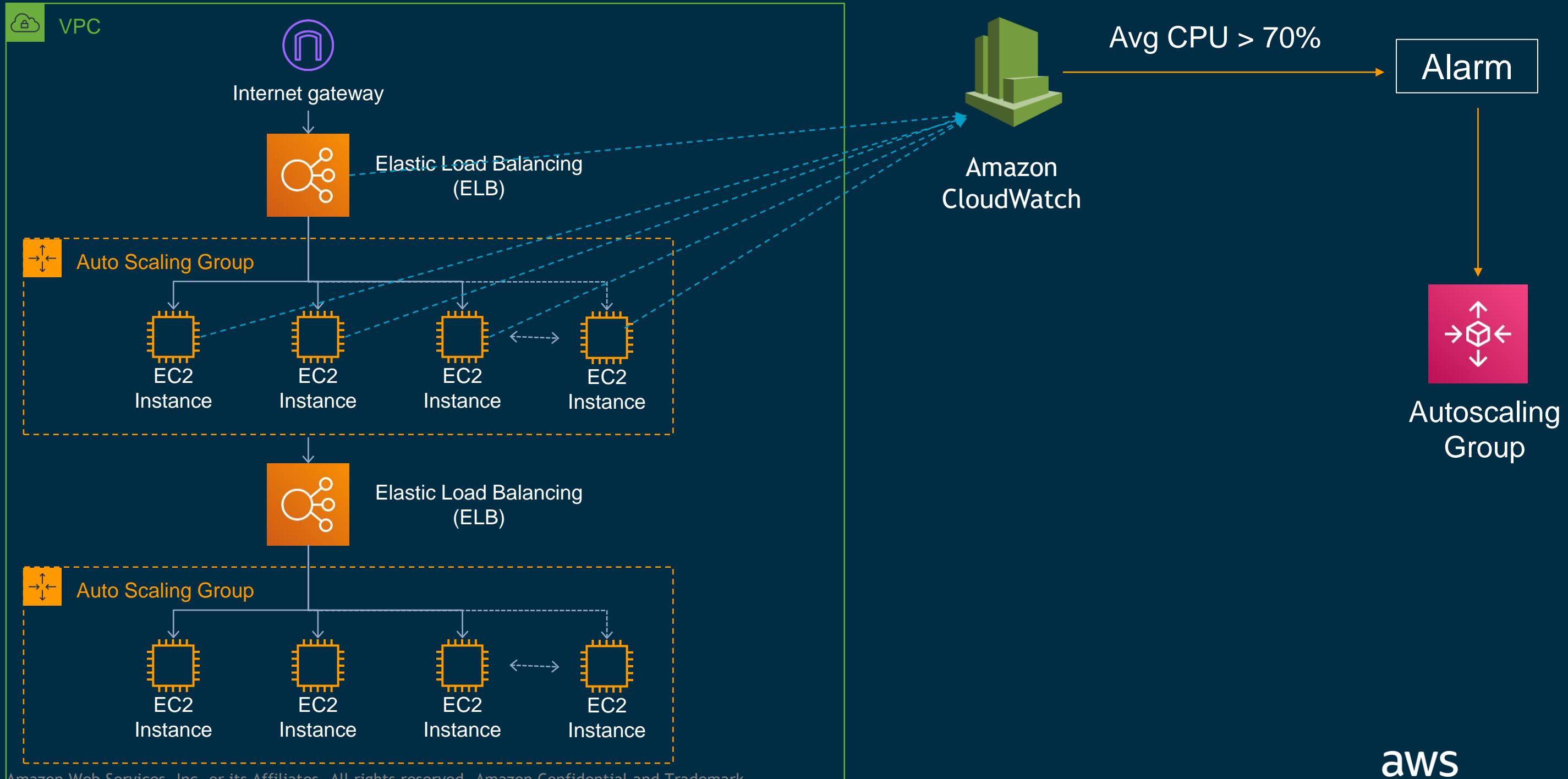
Let's create Load Balancer

1. Create Application Load Balancer
2. Create Target Group
3. Attach backend EC2 instances
4. Access Load Balancer using AWS provided DNS for LB

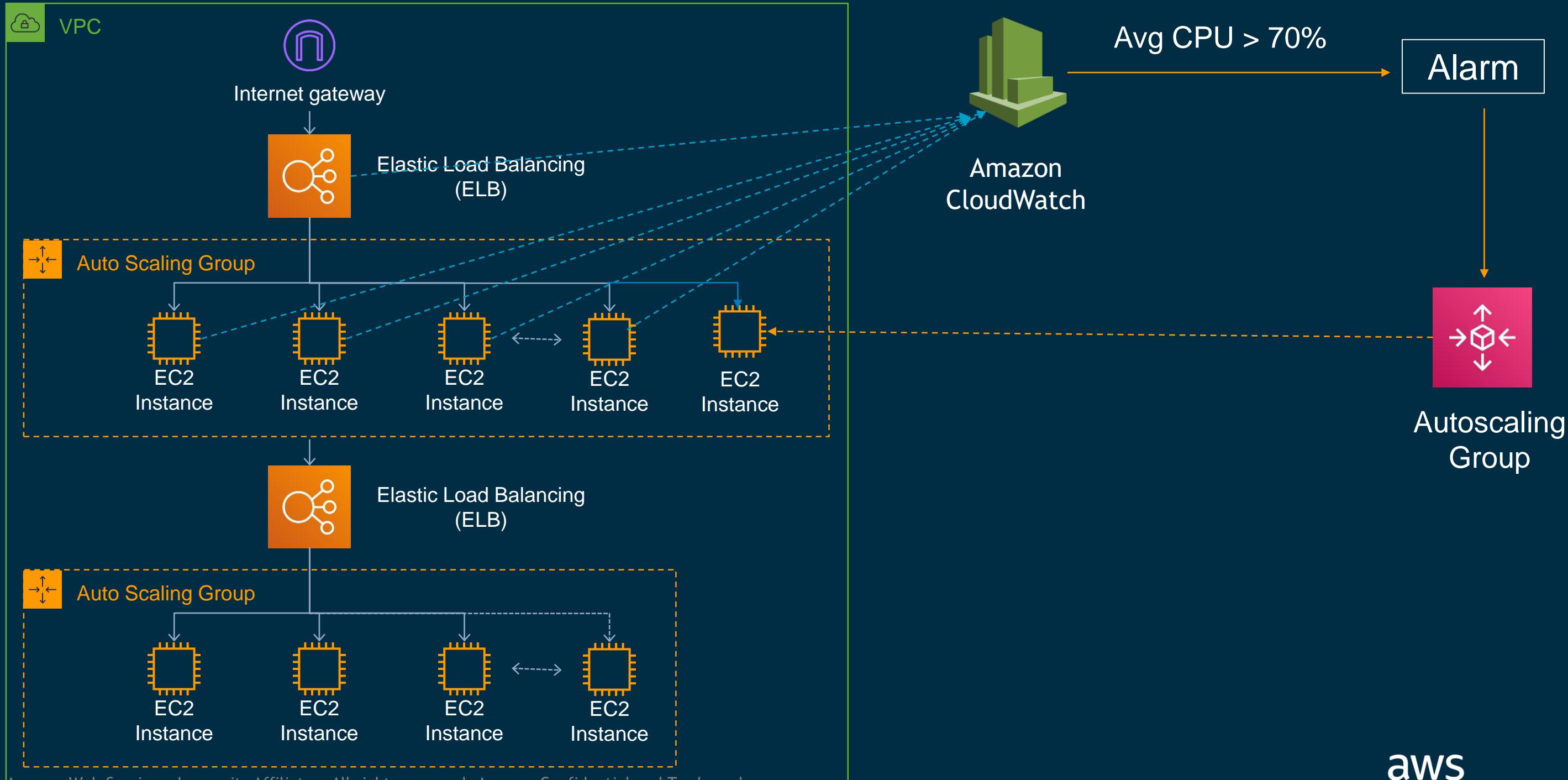
Features Comparison

Feature	Application Load Balancer	Network Load Balancer
Protocols	HTTP, HTTPS	TCP
Platforms	VPC	VPC
Health checks	✓	✓
CloudWatch metrics	✓	✓
Logging	✓	✓
Path-Based Routing	✓	
Host-Based Routing	✓	
Native HTTP/2	✓	
Configurable idle connection timeout	✓	
SSL offloading	✓	
Server Name Indication (SNI)	✓	
Sticky sessions	✓	
Back-end server encryption	✓	
Static IP		✓
Elastic IP address		✓
Preserve Source IP address		✓

Autoscaling Group



Autoscaling Group



Purchasing Options

On-Demand

Pay for compute capacity by **the second** with no long-term commitments

For Spiky workloads or to define needs



Reserved Instances

Make a 1 or 3-year commitment and receive a **significant discount** off On-Demand prices

For committed utilization



Spot Instances

Spare EC2 capacity at **savings of up to 90%** off On-Demand prices

For time-insensitive or transient workloads
Need to be Fault-tolerant, stateless



Savings Plans

Commit to a \$/h spend and **share discount** across compute options and regions

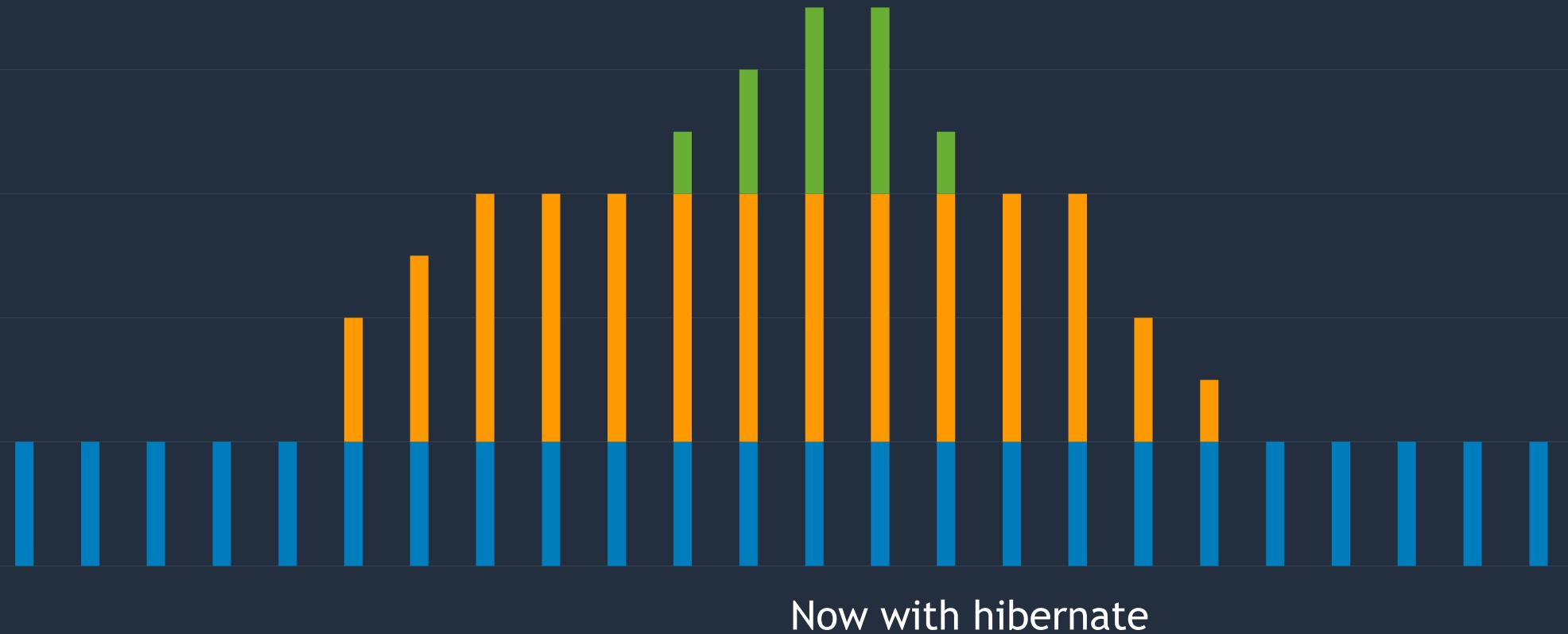
For committed utilization



To optimize EC2, combine all three purchase options!



Simplify capacity and cost optimization



Scale using
Spot,
On-Demand,
or both

Use **Reserved Instances**
for known/steady-state
workloads

AWS services make this easy and efficient



Amazon EC2
Auto Scaling



EC2 Fleet



Amazon Elastic
Container Service



Amazon Elastic
Container Service
for Kubernetes



AWS
Thinkbox



Amazon
EMR

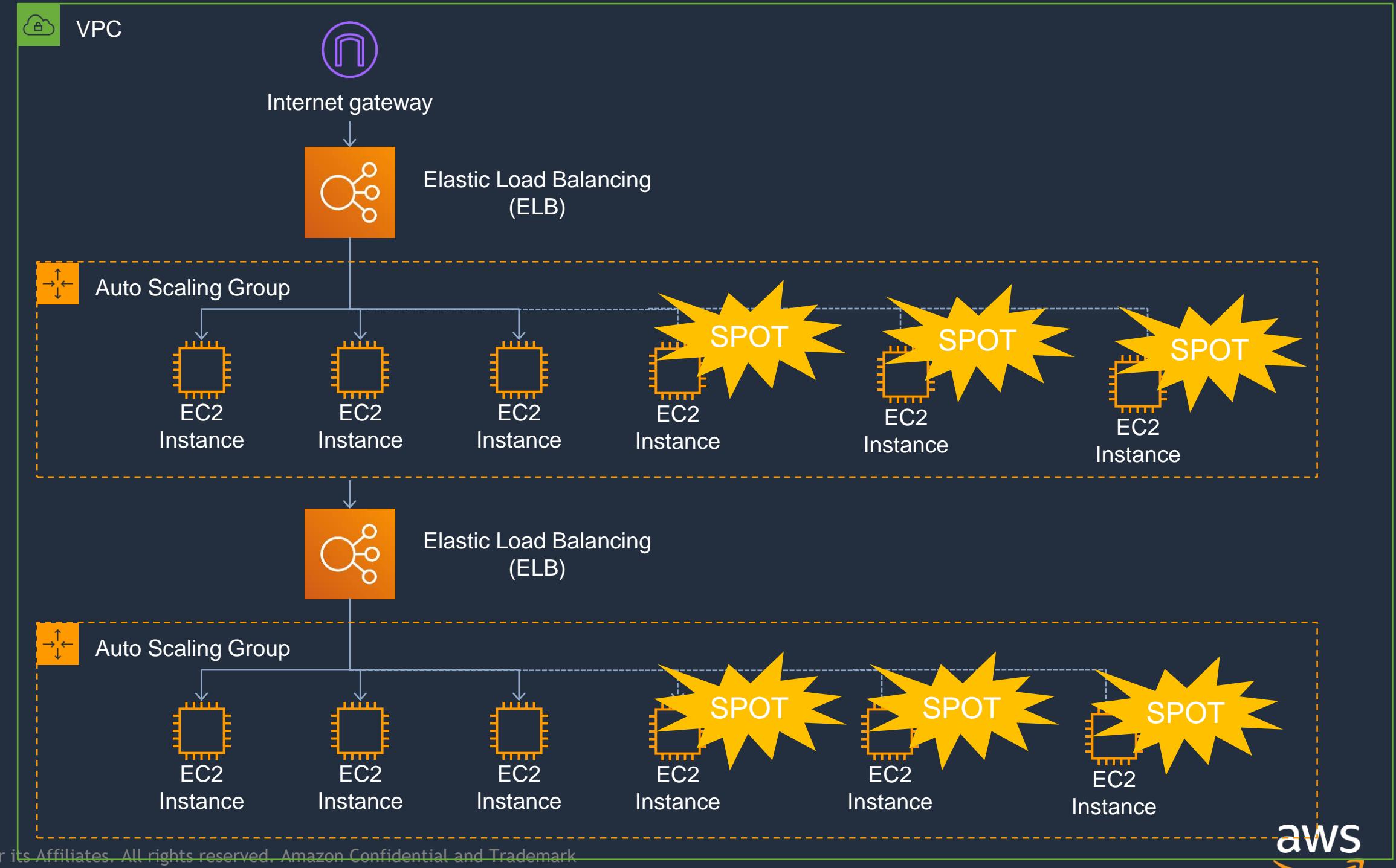


AWS
CloudFormation



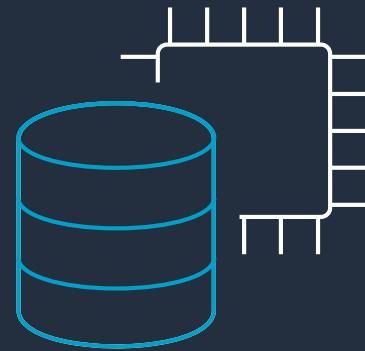
AWS
Batch

Using On-demand and Spot instances together..

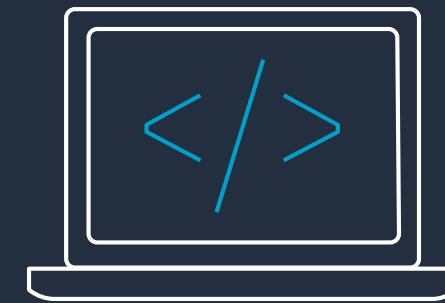


Hibernate Amazon EC2 Instances

Maintain a fleet of pre-warmed instances to quickly get to a productive state



Available with Amazon EBS-backed instances



Use familiar Stop and Start APIs



Memory data saved in EBS root volume



RAM contents are encrypted on EBS

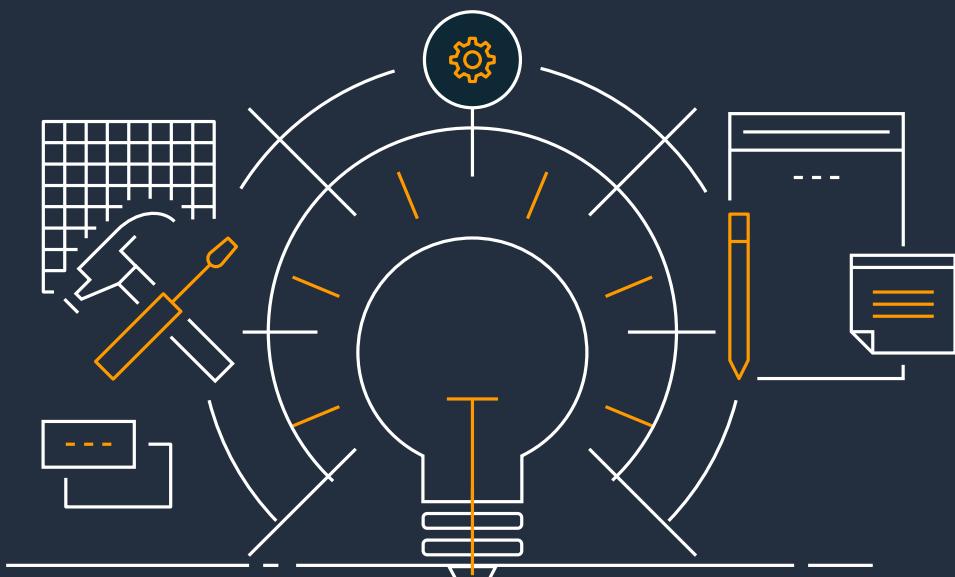
Its just like closing and opening your laptop!

Applications can pick up right where it left off

NEW!

Introducing AWS Compute Optimizer

DELIVERS INSTANCE TYPE AND AUTO SCALING GROUP RECOMMENDATIONS



Applies insights from millions of workloads to make recommendations

Saves time comparing and selecting optimal compute resources for your workload

Recommends 3 optimal instance options for EC2 and EC2 Auto Scaling groups from 140+ instances from M, C, R, T, and X families

Available at no additional charge

Quiz time ..

joinmyquiz.com

Q&A



Thank you

Chetan Agrawal – agrcheta@amazon.com

Deven Suri – dsuri@amazon.com