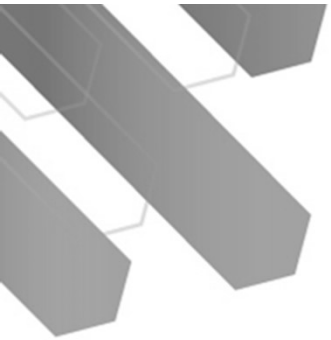


Cloud Computing with Amazon Web Services

Module overview

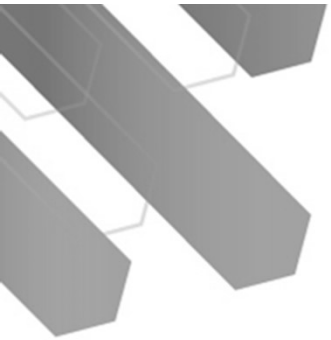
Topics

- Introduction to cloud computing
- Advantages of cloud computing
- Introduction to Amazon Web Services (AWS)

Module objectives

After completing this module, you should be able to:

- Define different types of cloud computing models
- Describe six advantages of cloud computing
- Recognize the main AWS service categories and core services

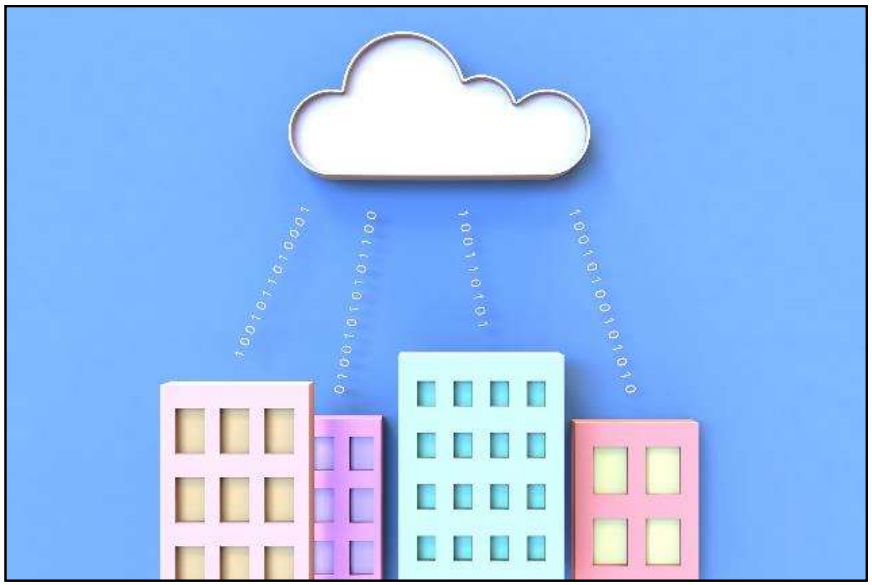


Introduction to cloud computing

What is cloud computing?

Cloud computing defined

Cloud computing is the **on-demand** delivery of compute power, database, storage, applications, and other IT resources **via the internet** with **pay-as-you-go** pricing.



Infrastructure as software

Cloud computing enables you to **stop thinking of your infrastructure as hardware**, and instead **think of (and use) it as software.**

Traditional computing model

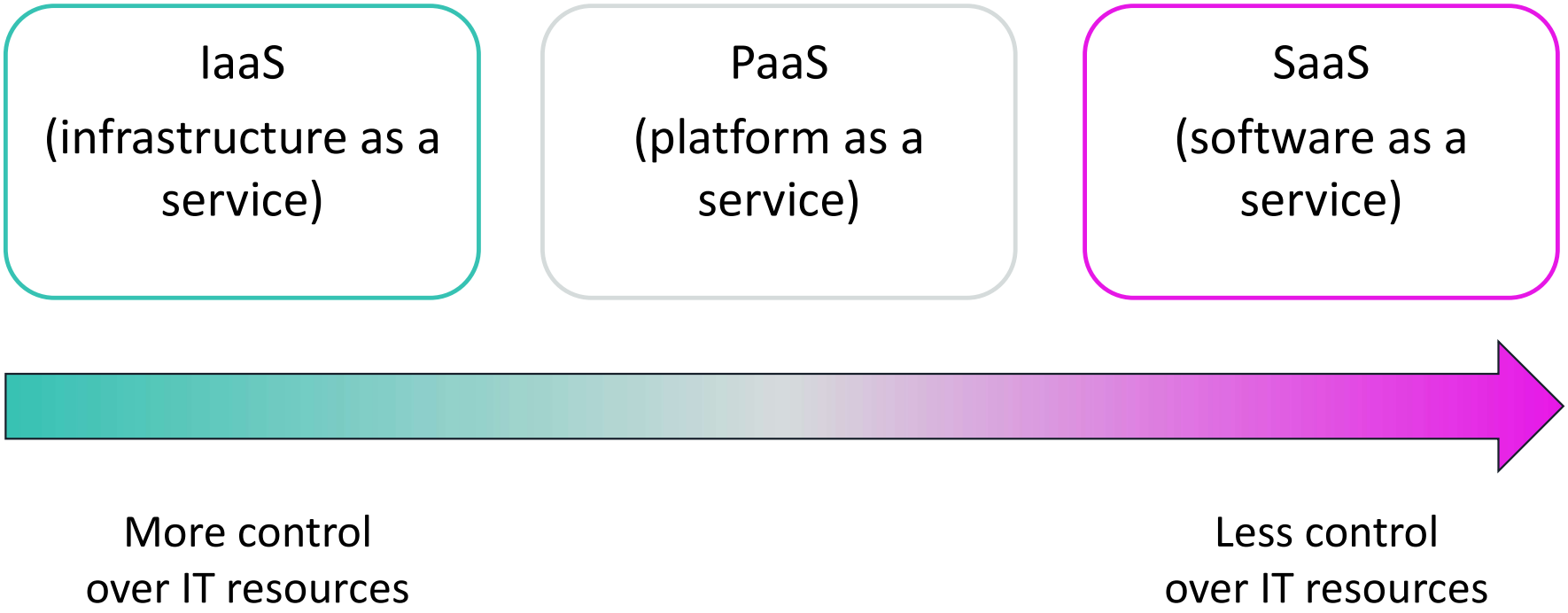
- Infrastructure as hardware
- Hardware solutions:

ical security, planning, capital expenditure
ocurement cycle
capacity by guessing theoretical maximum

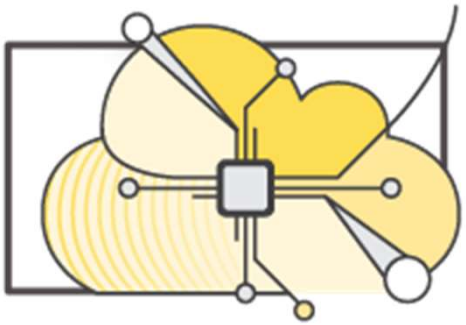
Cloud computing model

- Infrastructure as software
- Software solutions:
  - Are flexible
  - Can change more quickly, easily, and cost-effectively than hardware solutions
  - Eliminate the undifferentiated heavy-lifting tasks

Cloud service models

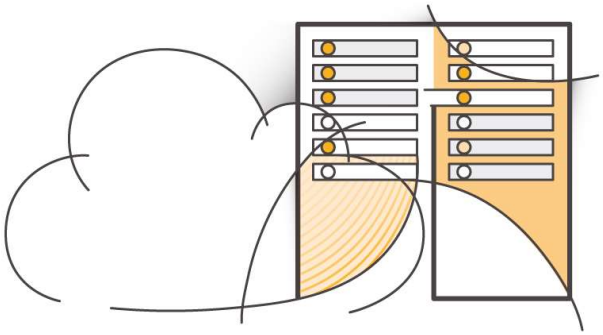


Cloud computing deployment models



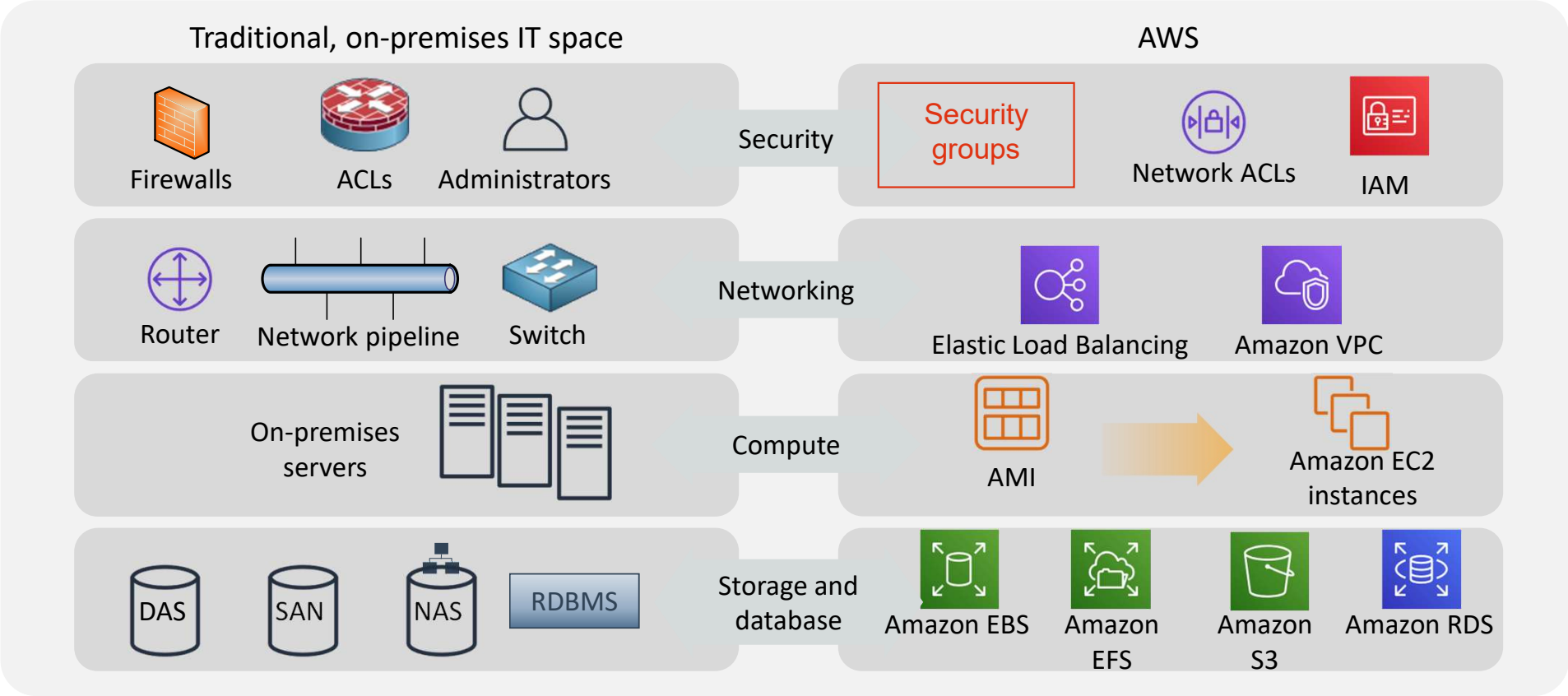
Cloud

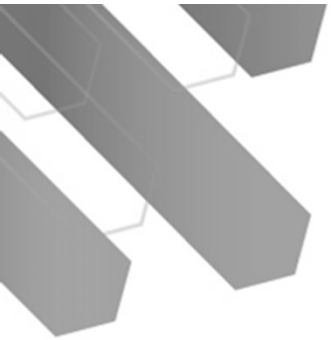
Hybrid



On-premises (private cloud)

Similarities between AWS and traditional IT

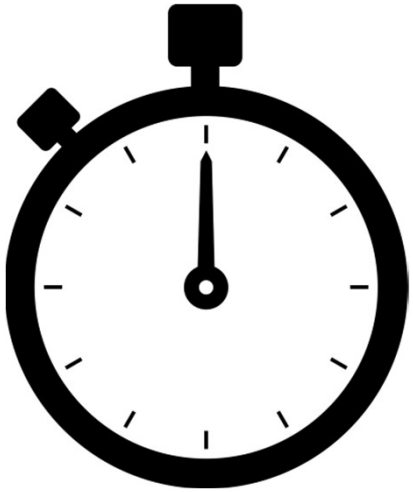




Advantages of cloud computing

Trade capital expense for variable expense

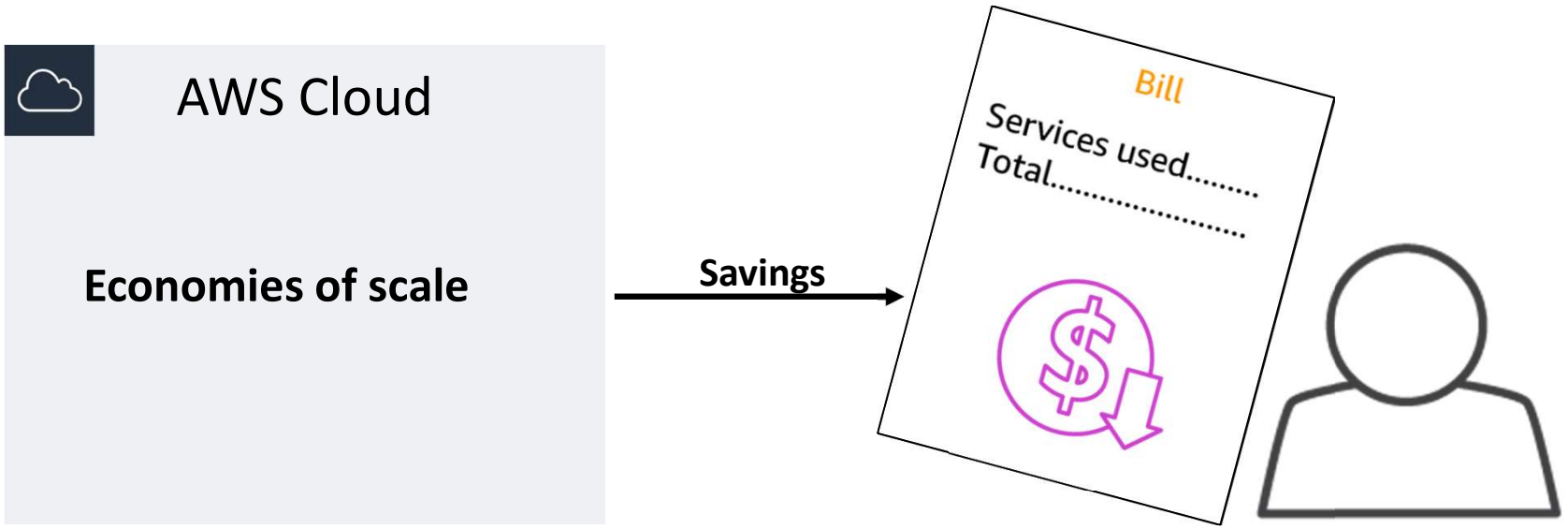
Data center investment based on forecast



Pay only for the amount you consume

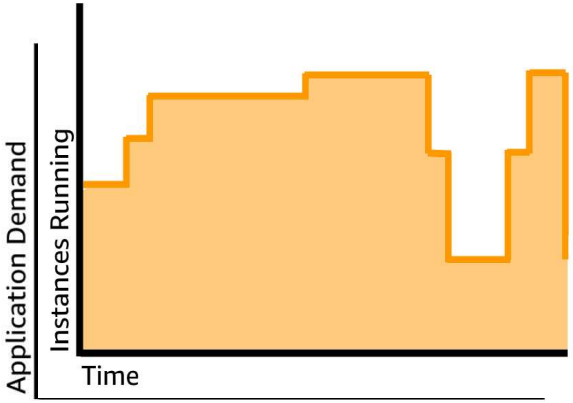
Massive economies of scale

Because of aggregate usage from all customers, AWS can achieve higher economies of scale and pass savings on to customers.



Stop guessing capacity

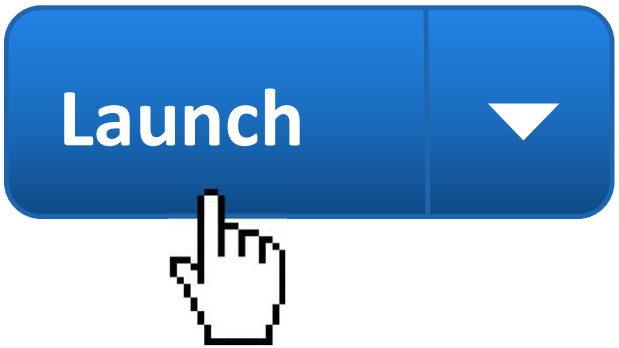
Underestimated server capacity



Scaling on demand

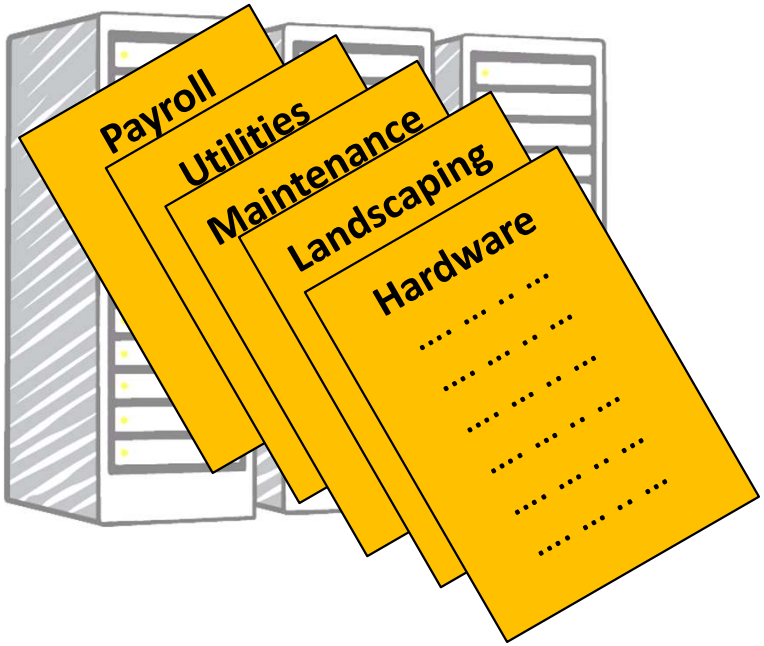
Increase speed and agility

Weeks between wanting resources and having resources

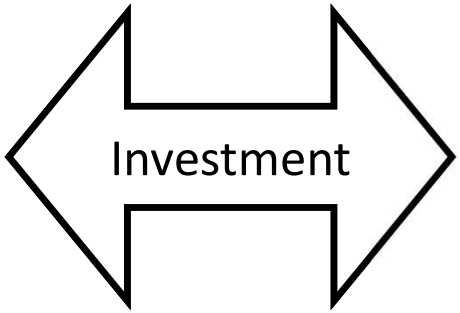


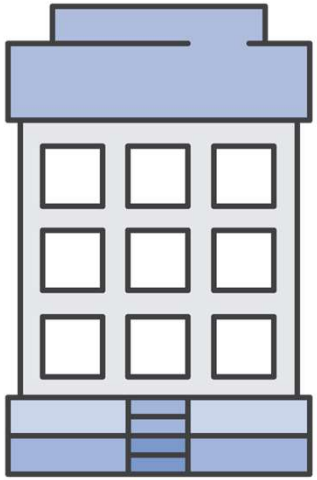
Minutes between wanting resources and having resources

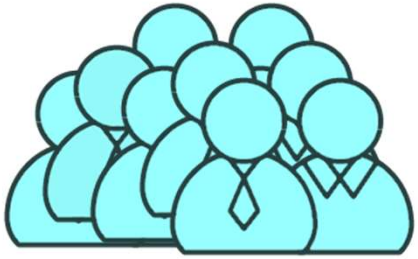
Stop spending money on running and maintaining data centers



Running data centers

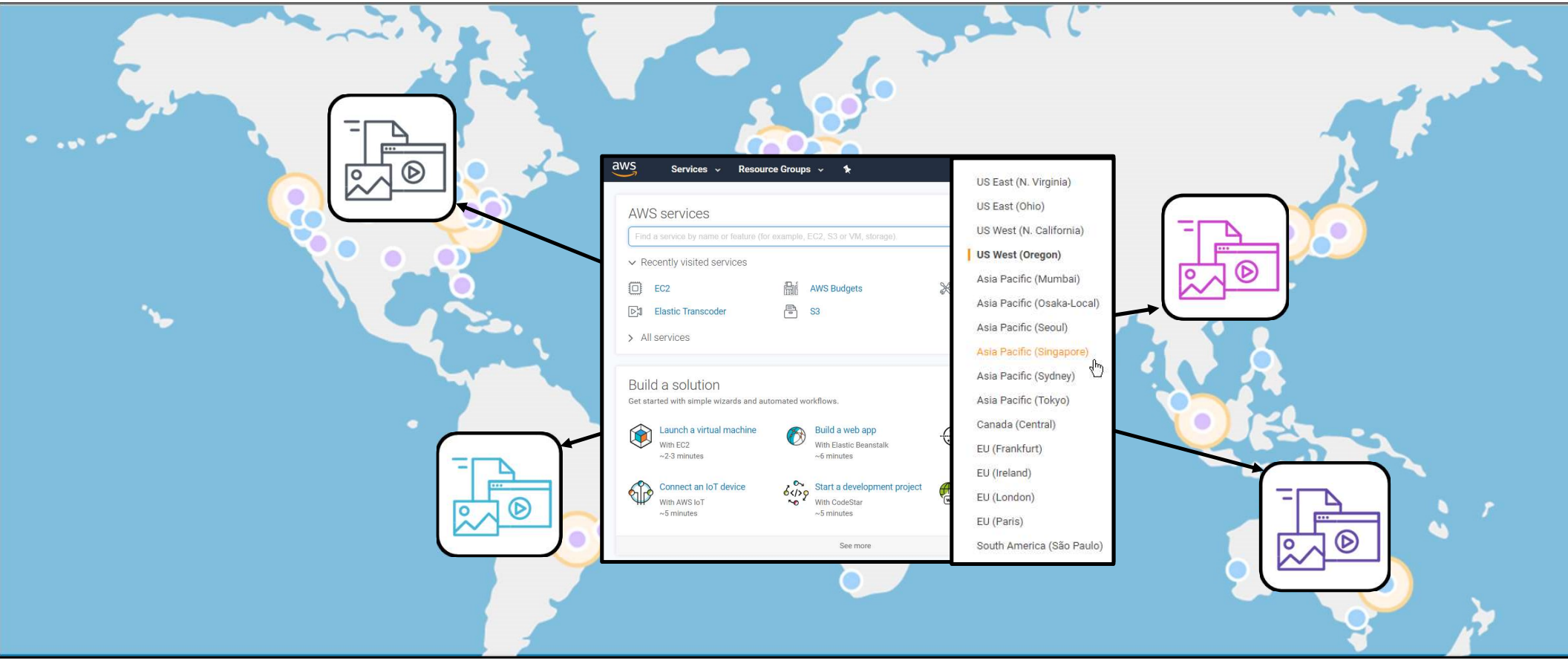


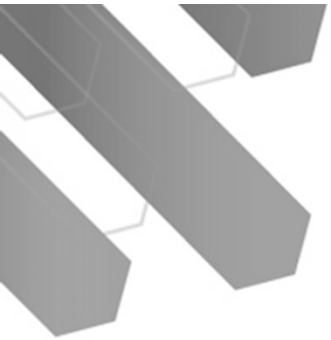




Business and customers

Go global in minutes

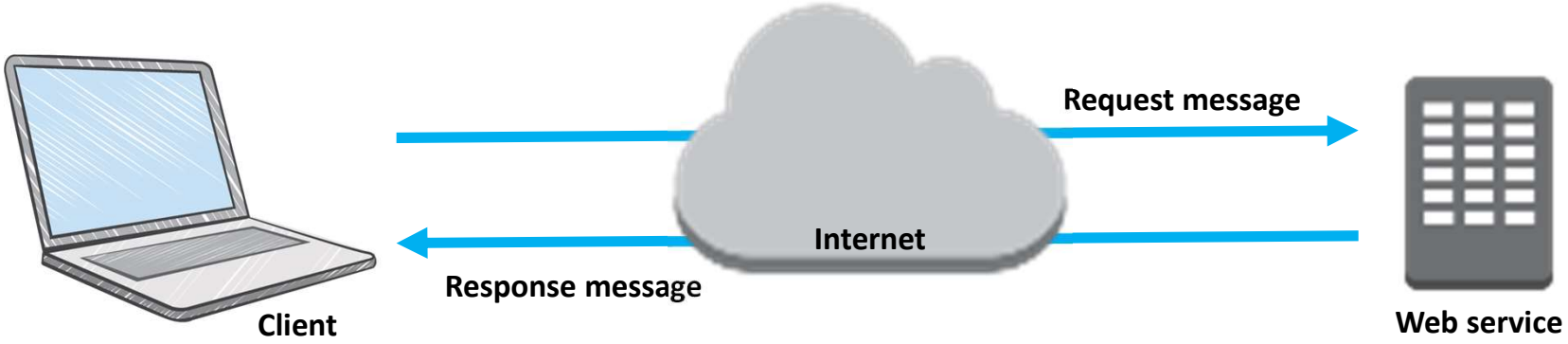




Introduction to Amazon Web Services (AWS)

What are web services?

A **web service** is any piece of software that makes itself available over the internet and uses a **standardized format**—such as Extensible Markup Language (XML) or JavaScript Object Notation (JSON)—for the request and the response of an **application programming interface (API) interaction**.



What is AWS?

- AWS is a **secure cloud platform** that offers a **broad set of global cloud-based products**.
- AWS provides you with **on-demand access** to compute, storage, network, database, and other IT resources and management tools.
- AWS offers **flexibility**.
- You **pay only for the individual services you need**, for **as long as you use them**.
- AWS services **work together** like building blocks.

Categories of AWS services


Analytics


Application Integration


AR and VR


Blockchain


Business Applications


Compute


Cost Management


Customer Engagement


Database


Developer Tools


End User Computing


Game Tech


Internet of Things


Machine Learning


Management and Governance


Media Services


Migration and Transfer


Mobile


Networking and Content Delivery

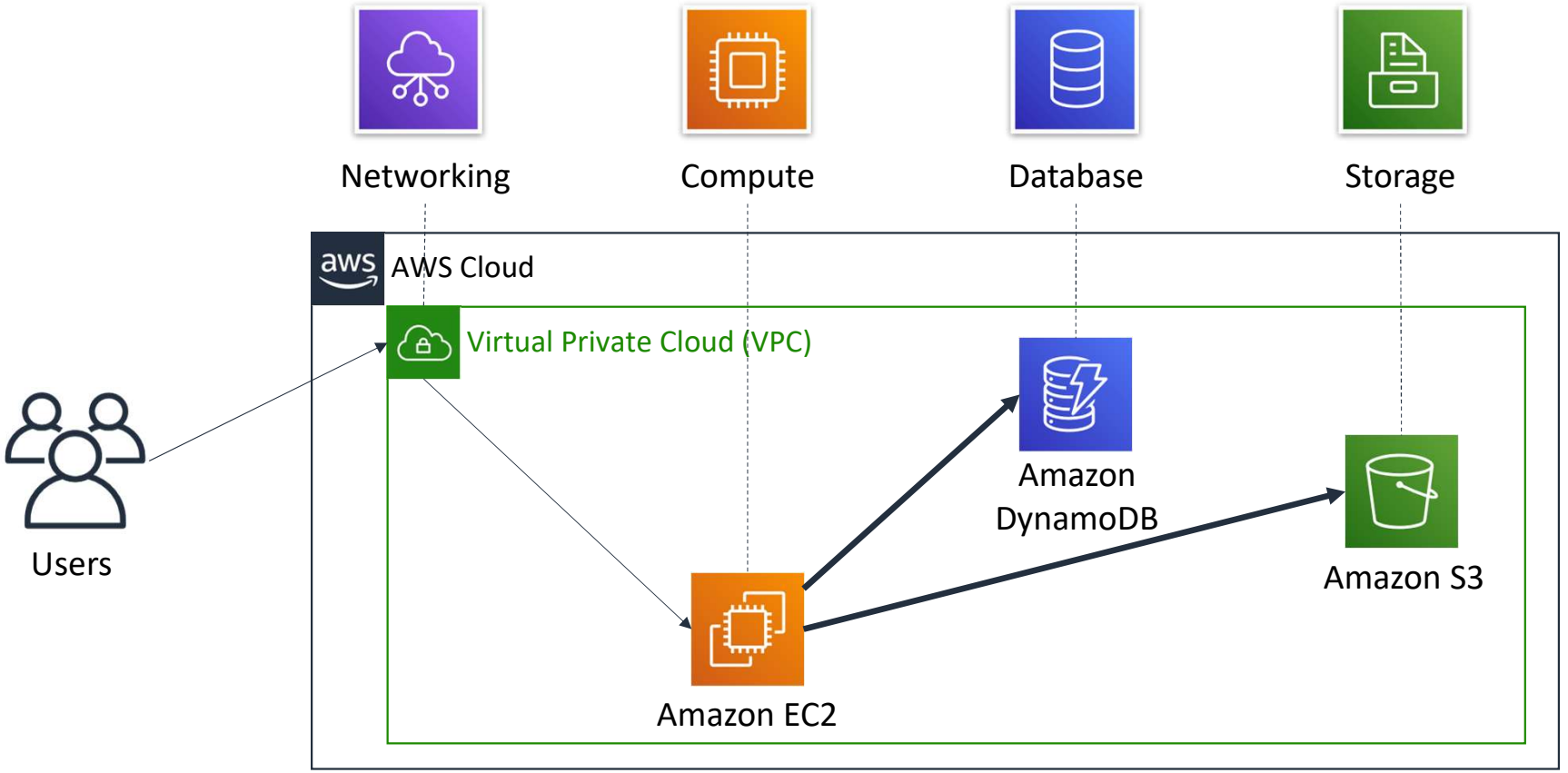

Robotics


Satellite


Security, Identity, and Compliance

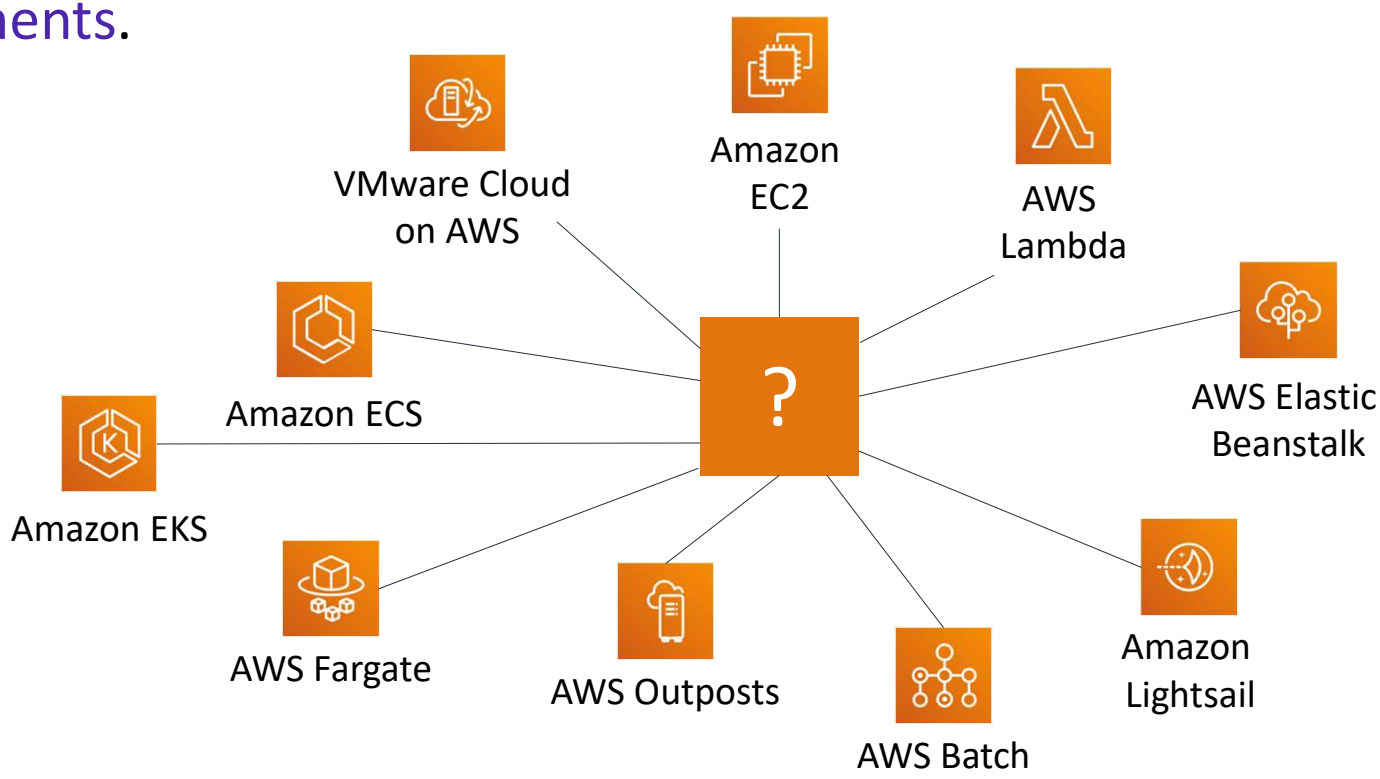

Storage

Simple solution example



Choosing a service

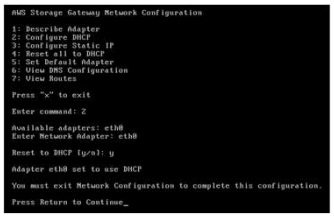
The service you select depends on your business goals and technology requirements.



Three ways to interact with AWS

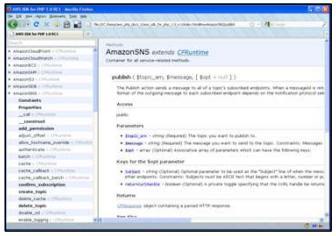
AWS Management Console

Easy-to-use graphical interface



Command Line Interface (AWS CLI)

Access to services by discrete commands or scripts



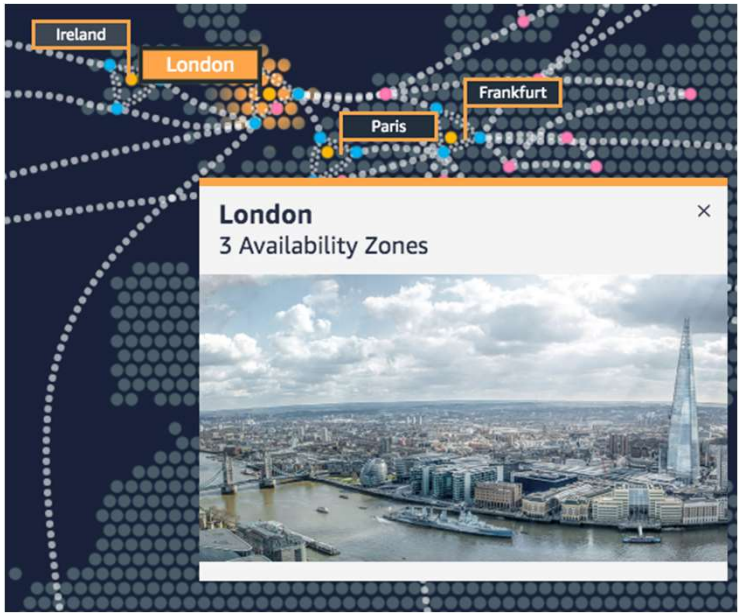
Software Development Kits (SDKs)

Access services directly from your code (such as Java, Python, and others)

AWS Global Infrastructure

AWS Regions

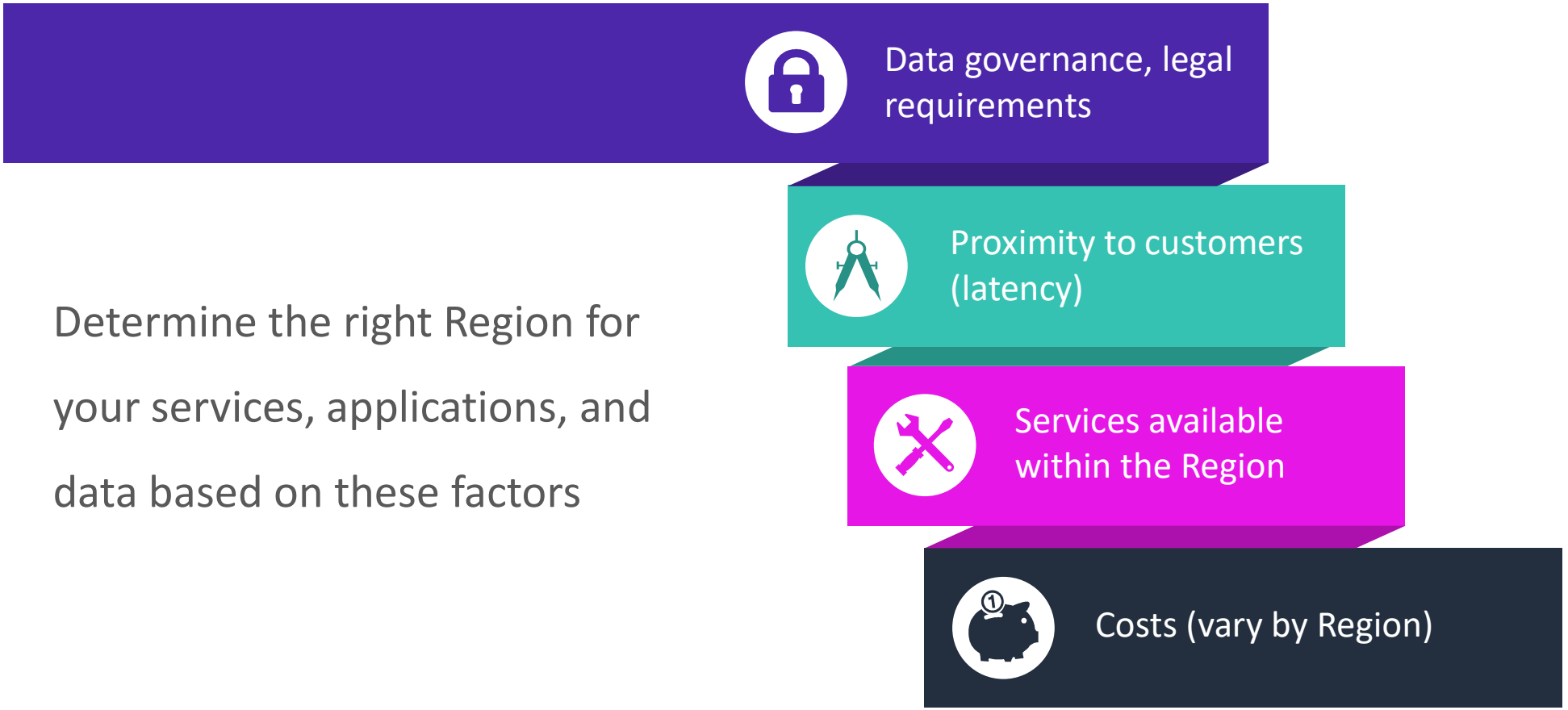
- An **AWS Region** is a geographical area.
  - Data replication across Regions is controlled by you.
  - Communication between Regions uses AWS backbone network infrastructure.
- Each Region provides full redundancy and connectivity to the network.
- A Region typically consists of two or more **Availability Zones**.



Example: London Region

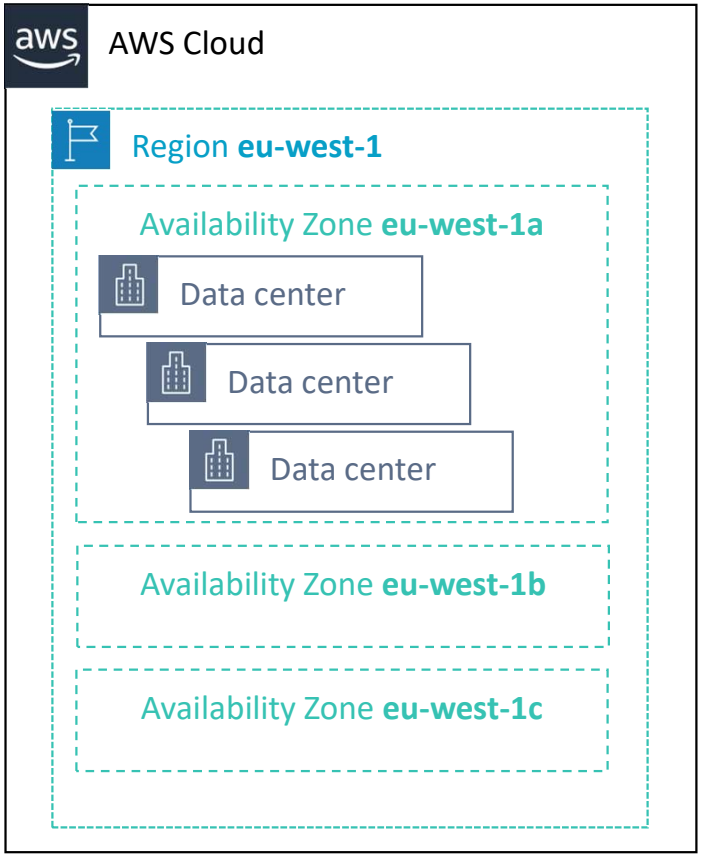
Selecting a Region

Determine the right Region for your services, applications, and data based on these factors



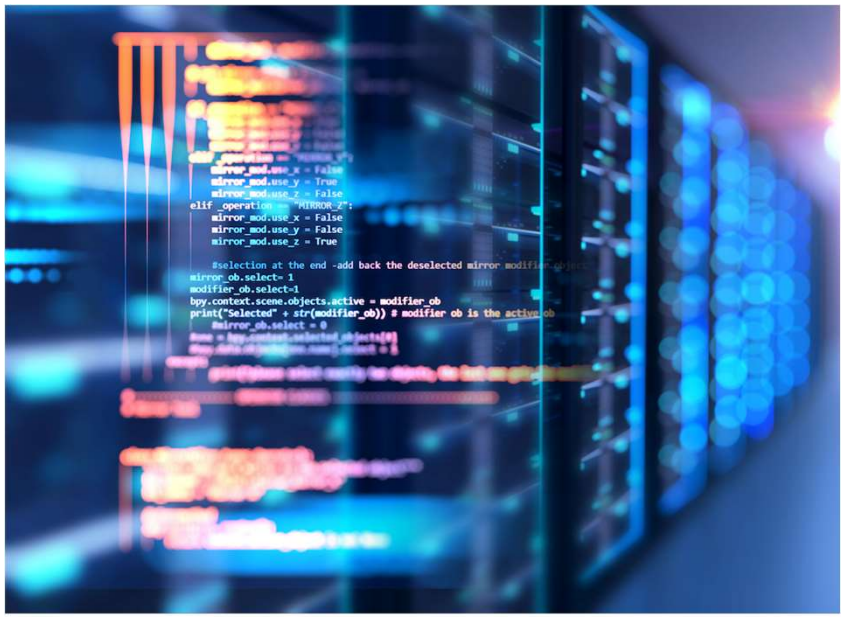
Availability Zones

- Each **Region** has multiple Availability Zones.
- Each **Availability Zone** is a fully isolated partition of the AWS infrastructure.
  - There are currently 69 Availability Zones worldwide
  - Availability Zones consist of discrete **data centers**
  - They are designed for fault isolation
  - They are interconnected with other Availability Zones by using high-speed private networking
  - You choose your Availability Zones.
  - **AWS recommends replicating data and resources across Availability Zones** for resiliency.



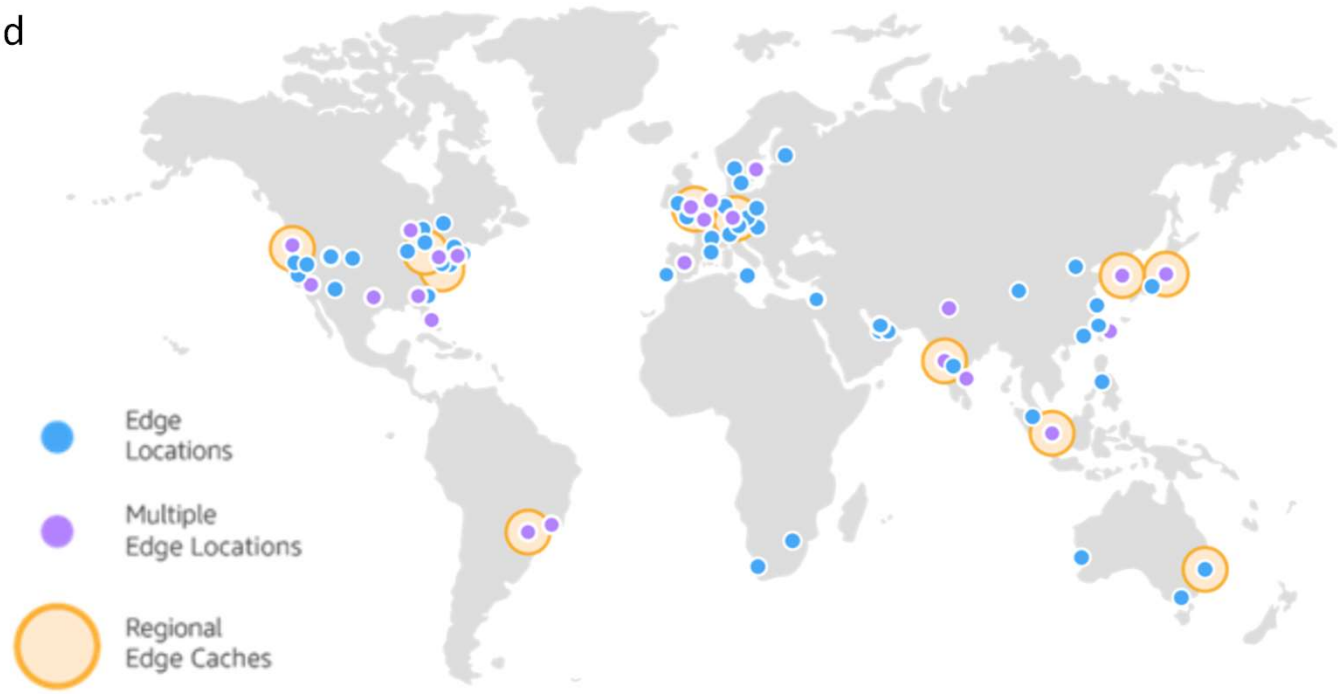
AWS data centers

- AWS data centers are designed for security.
- Data centers are where the data resides and data processing occurs.
- Each data center has redundant power, networking, and connectivity, and is housed in a separate facility.
- A data center typically has 50,000 to 80,000 physical servers.



Points of Presence

- AWS provides a global network of 187 Points of Presence locations
- Consists of 176 **edge locations** and 11 **Regional edge caches**
- Used with Amazon CloudFront
  - A global Content Delivery Network (CDN), that delivers content to end users with reduced latency
- Regional edge caches used for content with infrequent access.



AWS infrastructure features

- Elasticity and scalability
  - Elastic infrastructure; dynamic adaption of capacity
  - Scalable infrastructure; adapts to accommodate growth
- Fault-tolerance
  - Continues operating properly in the presence of a failure
  - Built-in redundancy of components
- High availability
  - High level of operational performance
  - Minimized downtime
  - No human intervention