Module 1: Cloud Concepts Overview

AWS Academy Cloud Foundations

Section 1: Introduction to cloud computing

Module 1: Cloud Concepts Overview

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~~.

Computing / IT

network

Infrastructure as software

because of virtualization

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

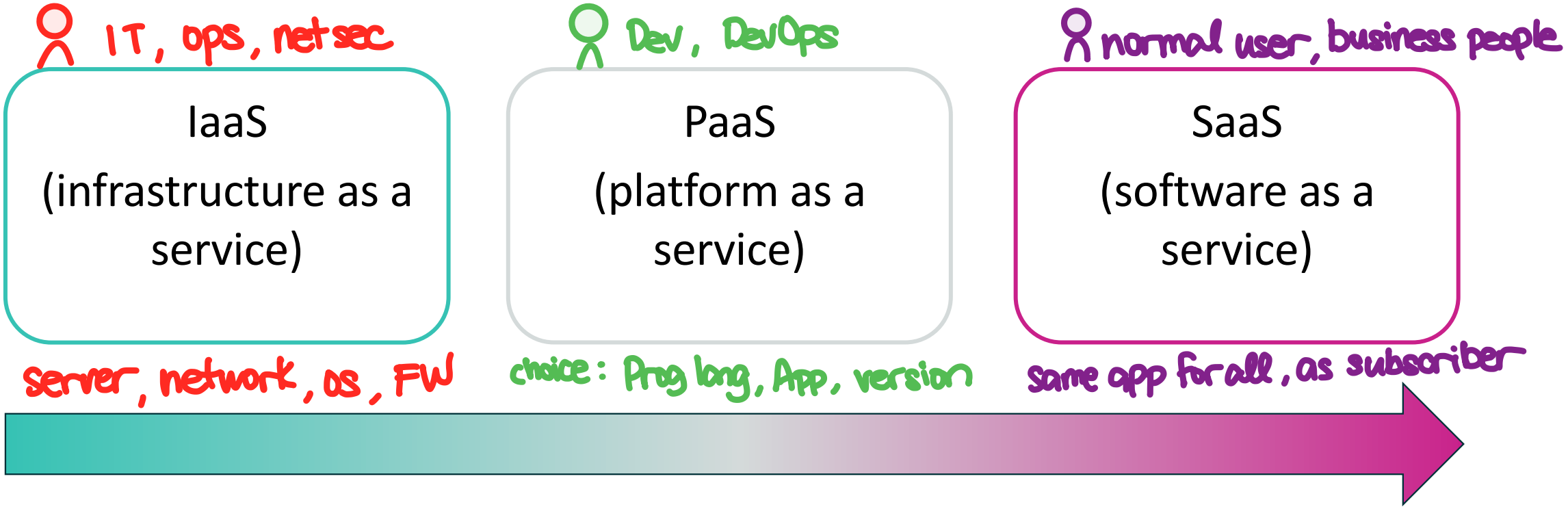
Traditional computing model

- Infrastructure as hardware
- Hardware solutions:
  - Require space, staff, physical security, planning, capital expenditure
    - upfront
  - Have a long hardware procurement cycle
  - Require you to provision capacity by guessing theoretical maximum peaks
    - wait → have
    - guessed theoretical max peak → delay → overestimation / underestimation

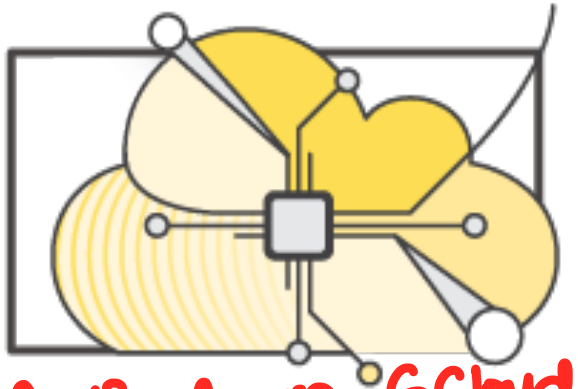
Cloud computing model

- Infrastructure as software → virtualization
- 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



AWS, Azure, GCloud

↓ capex ↑ opex

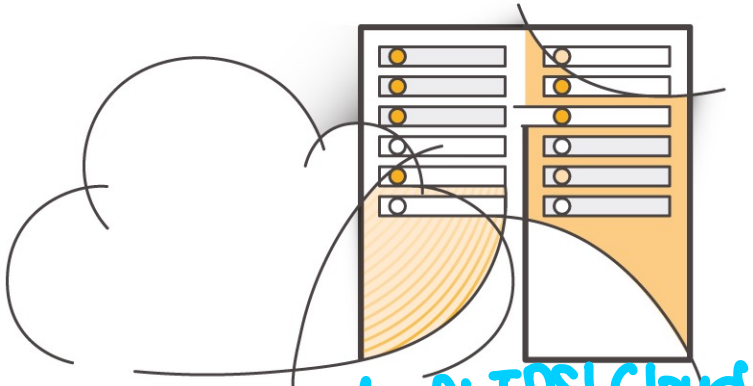
Cloud

Public Cloud
via Internet
Global
Rent resource

best of both

Hybrid

Public + Private
Private can use min resource
Public to rent when usage changes



CCS Cloud, ALTDSI Cloud

↑ capex ↓ opex

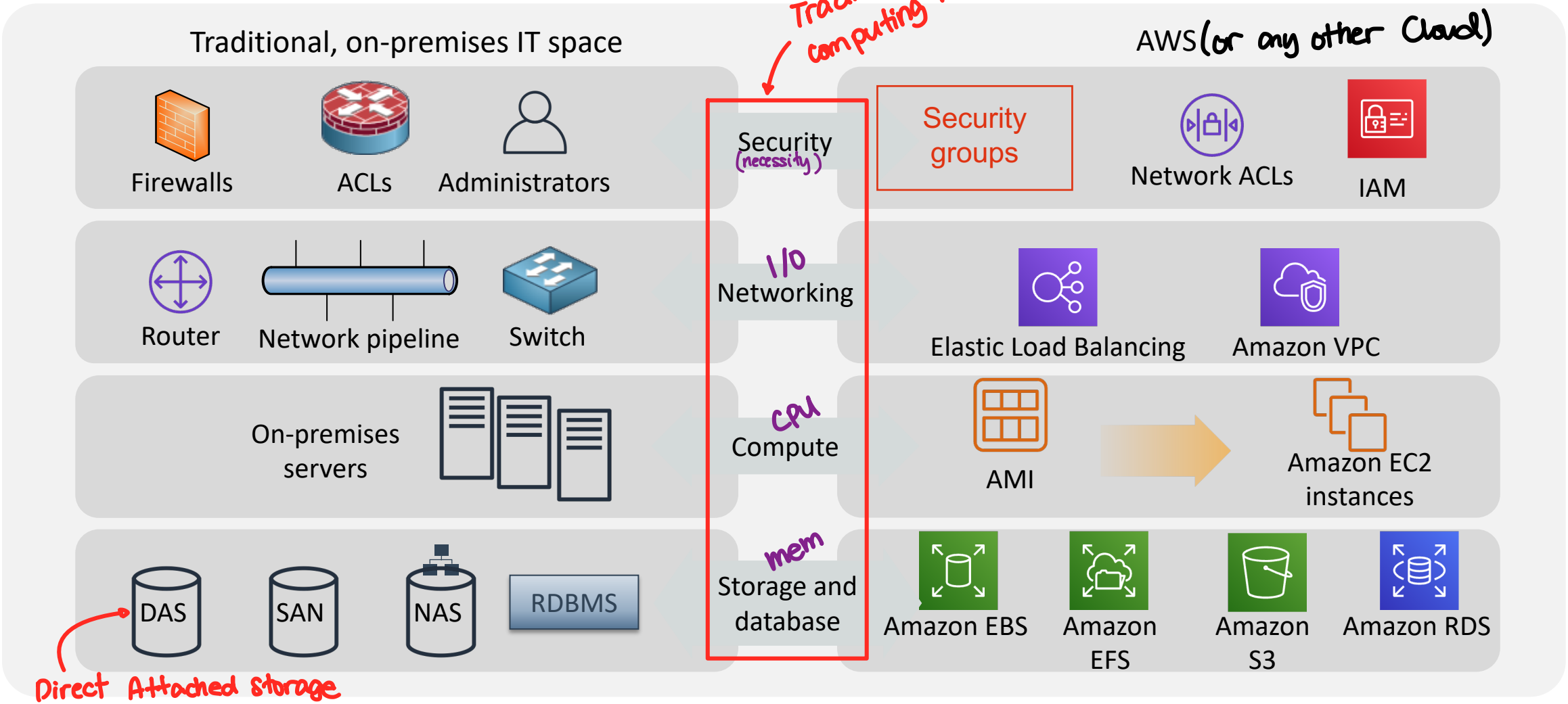
On-premises

(private cloud)

via network / internet
security
full control & responsibility

Similarities between AWS and traditional IT

Takeaway: whatever you can do in cloud you can also do in Traditional



Section 2: Advantages of cloud computing

Module 1: Cloud Concepts Overview

Trade capital expense for variable expense

→ options and accessibility

note: capex is not always bad

↳ capex → upfront

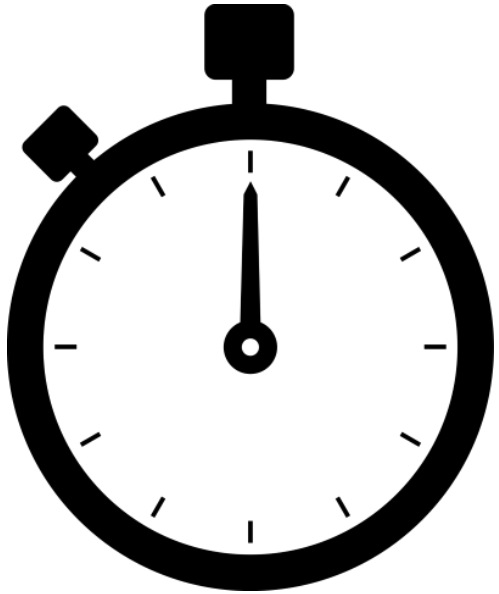
OK for large companies

↳ opex → recurring

OK for startup or testers

Data center investment based on forecast

buy servers



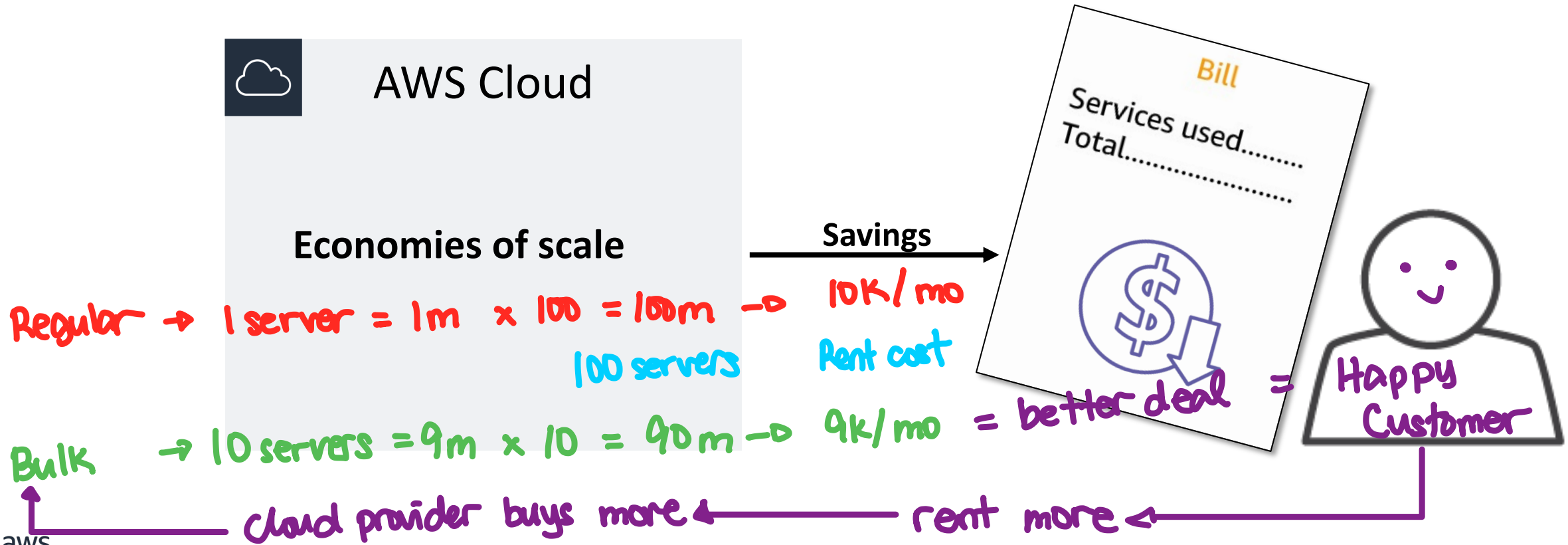
Pay only for the amount you consume

rent servers

Massive economies of scale → bulk use and purchase leads to discounts and savings

massive

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



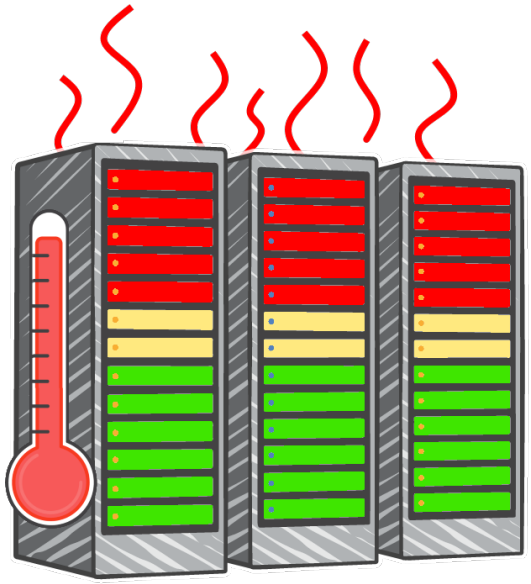
Stop guessing capacity

guess in traditional due to long HW procurement cycles ⇒ no need to guess if you rent

want ——delay——▸ have

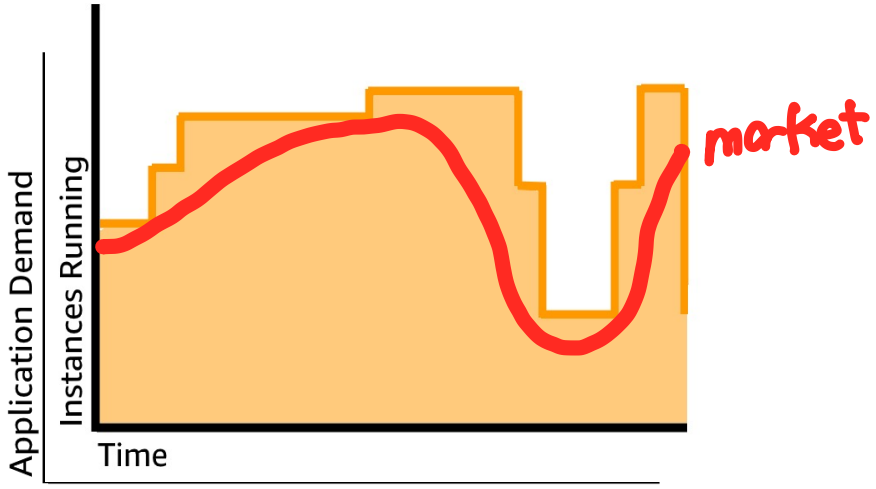
Overestimated server capacity

bought 2 servers but only need 1



Underestimated server capacity

bought 2 servers but needed 3



Scaling on demand

→ Rent more/less depending on your needs

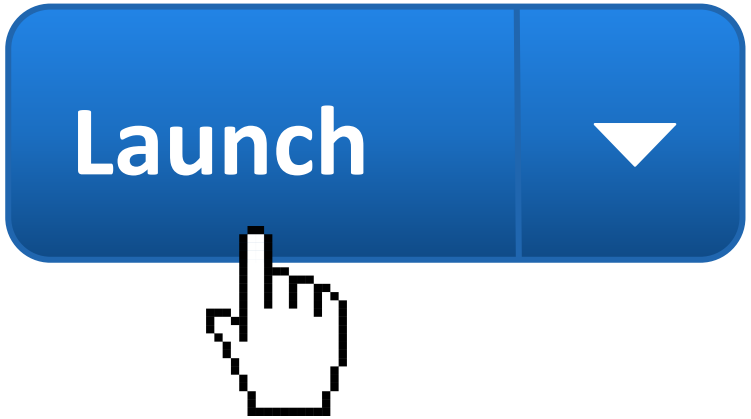
Increase speed and agility

w/ virtualization you treat your infra as software

HW soln take time

Weeks between wanting resources and having resources

SW soln are flexible and agile

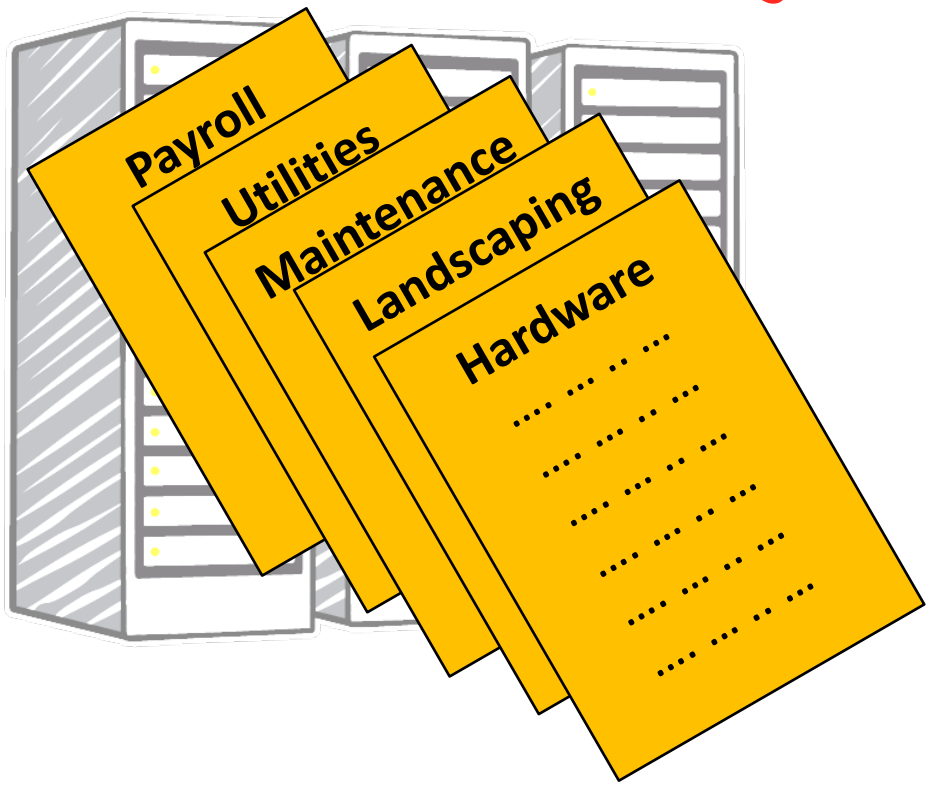


Minutes between wanting resources and having resources

Stop spending money on running and maintaining data centers

→ not about savings but more of where you put your money

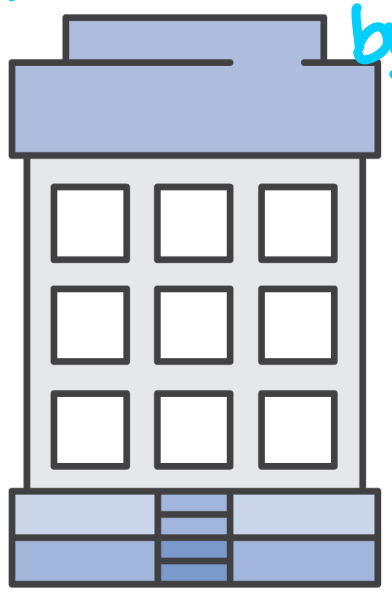
micro manage your money

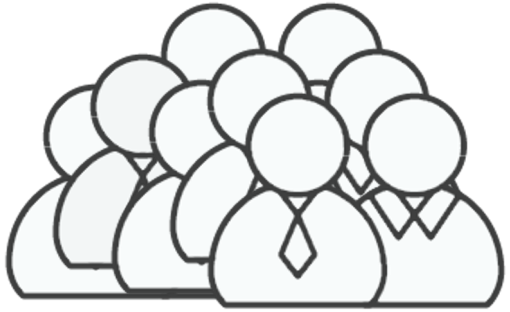


Running data centers

Investment

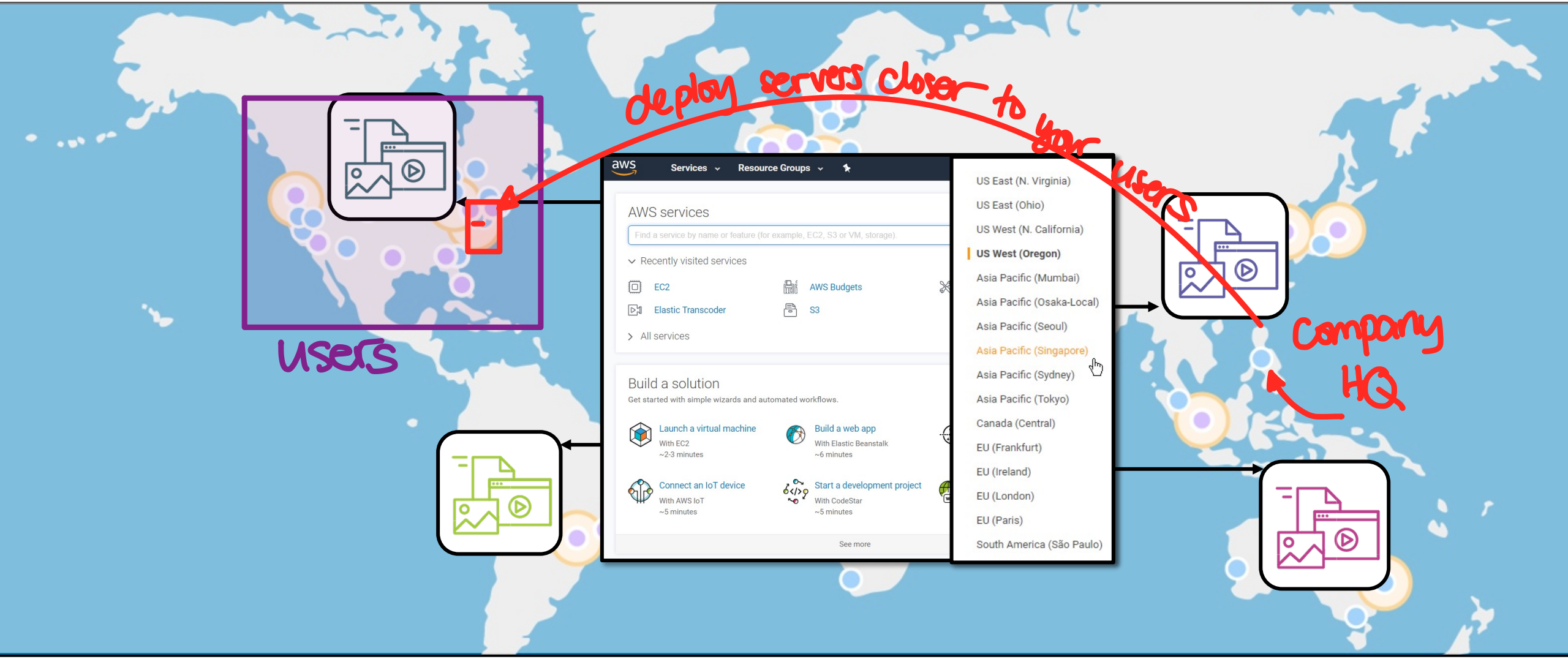
outsource management for a premium but earn more money by focusing on your business





Business and customers

Go global in minutes → reduce latency improve user experience



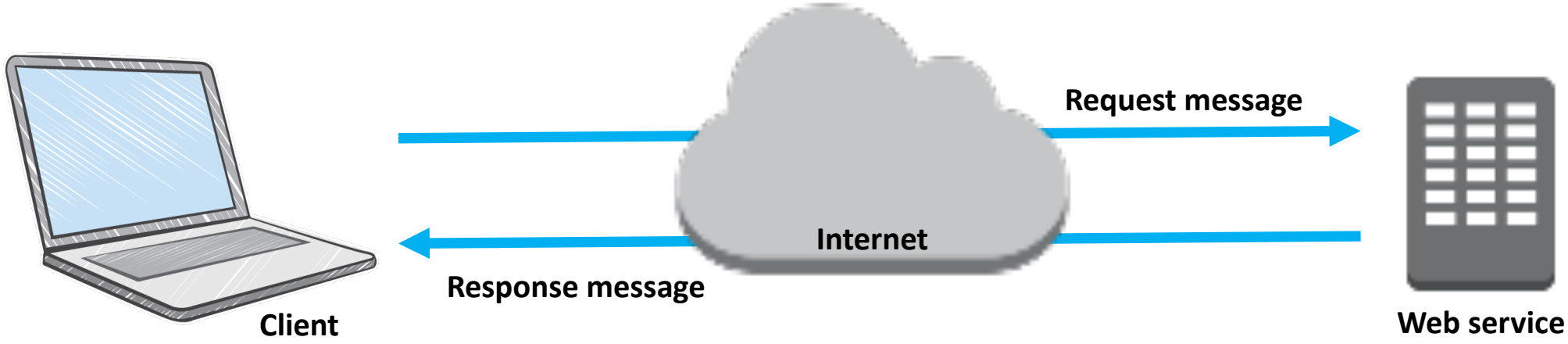
Section 3: Introduction to Amazon Web Services (AWS)

Module 1: Cloud Concepts Overview

What are web services?

the cloud works by having its services integrated together via standard formats and APIs

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**.



Categories of AWS services

(and other cloud providers)

→ many services & options to choose from depending on your need

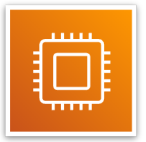

Analytics


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

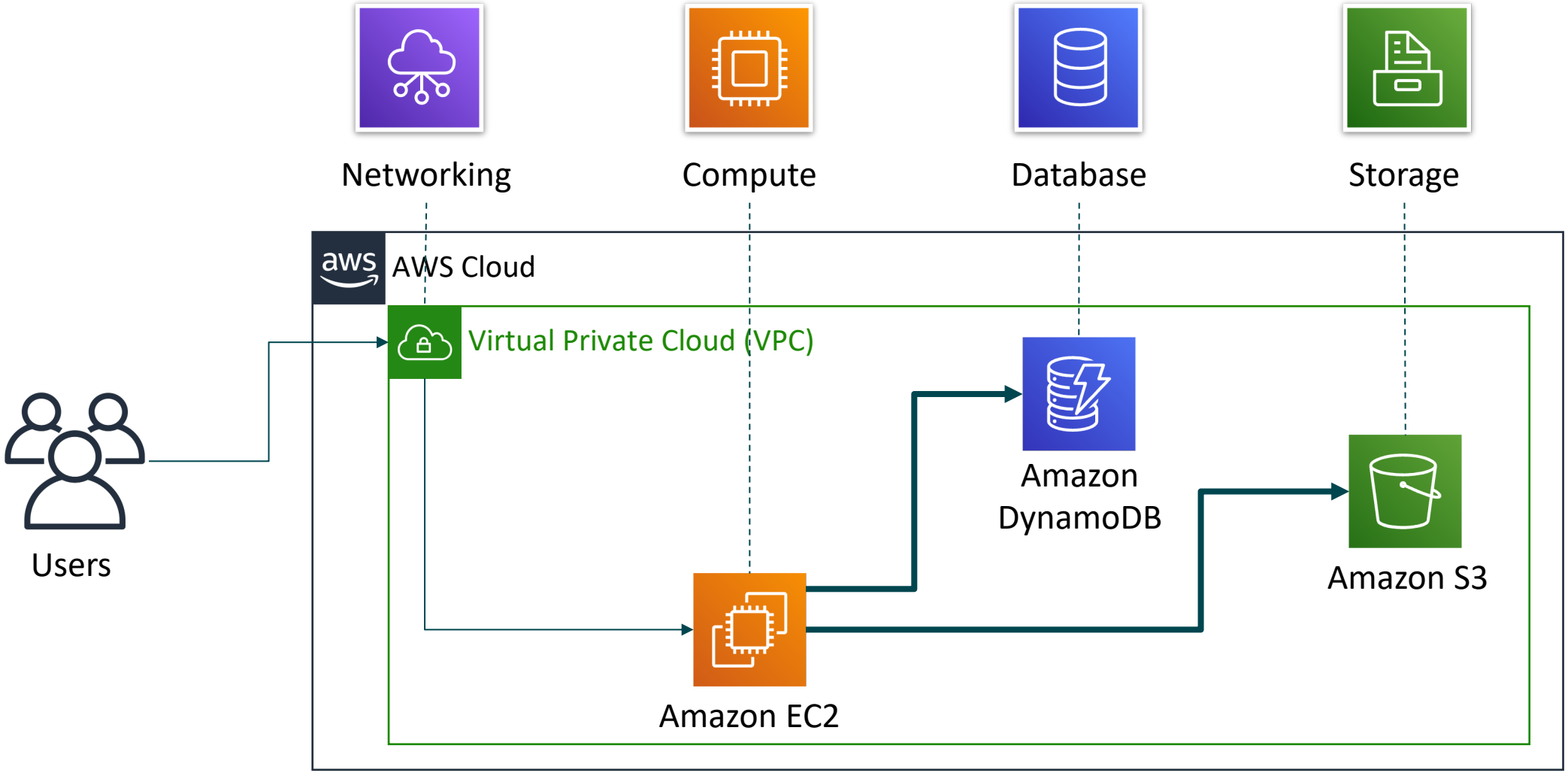

Robotics


Satellite


Security, Identity, and Compliance

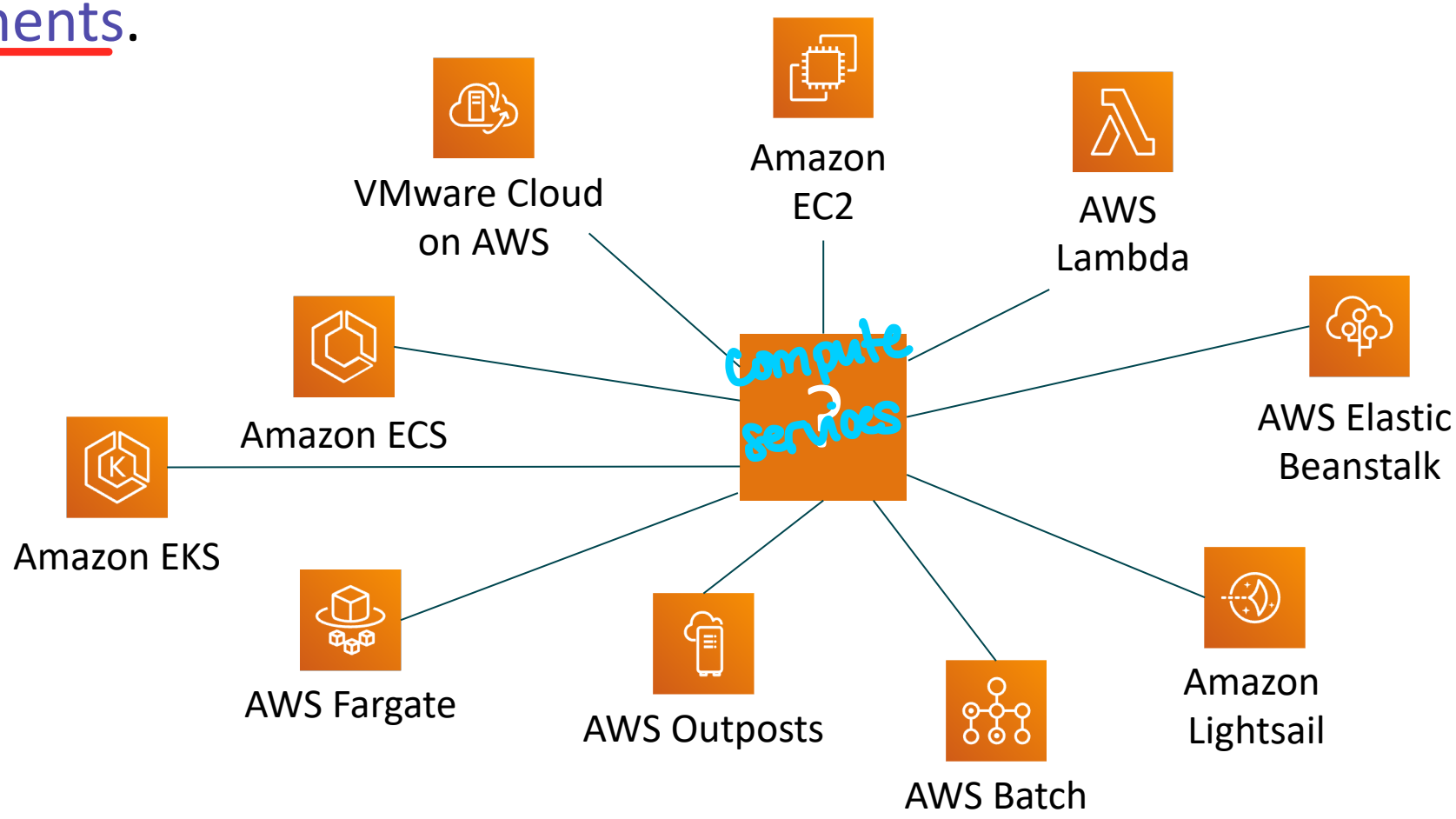

Storage

Simple solution example



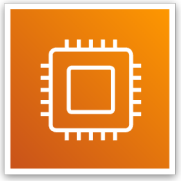
Choosing a service → many services/options to support needs of different users and use cases

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



Many services to support varying use cases

Compute services –



- Amazon EC2
- AWS Lambda
- AWS Elastic Beanstalk
- Amazon EC2 Auto Scaling
- Amazon ECS
- Amazon EKS
- Amazon ECR
- AWS Fargate

Security, Identity, and Compliance services –

- AWS IAM
- Amazon Cognito
- AWS Shield
- AWS Artifact
- AWS KMS

Storage services –

- Amazon S3
- Amazon S3 Glacier
- Amazon EFS
- Amazon EBS

Database services –

- Amazon RDS
- Amazon DynamoDB
- Amazon Redshift
- Amazon Aurora

Networking and Content Delivery services –

- Amazon VPC
- Amazon Route 53
- Amazon CloudFront
- Elastic Load Balancing

Management and Governance services –

- AWS Trusted Advisor
- AWS CloudWatch
- AWS CloudTrail
- AWS Well-Architected Tool
- AWS Auto Scaling
- AWS Command Line Interface
- AWS Config
- AWS Management Console
- AWS Organizations

AWS Cost Management services –

- AWS Cost & Usage Report
- AWS Budgets
- AWS Cost Explorer

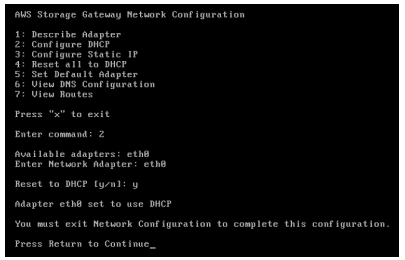
Three ways to interact with AWS

AWS Management Console

Easy-to-use graphical interface

via Web Interface

students
testers
business people

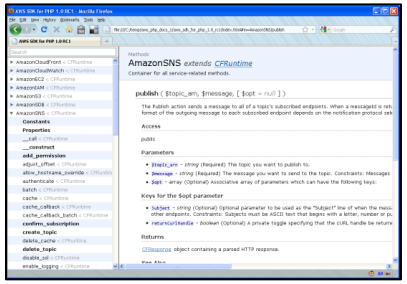


Command Line Interface (AWS CLI)

Access to services by discrete commands or scripts

→ good for automation / batch

IT
ops
net sec



Software Development Kits (SDKs)

Dev, Dev ops

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

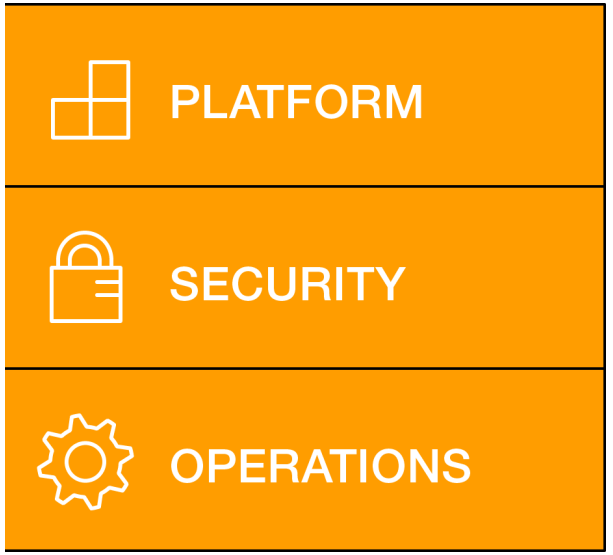
↳ custom apps

Cloud Adoption Framework: Six Perspectives

→ things to think about before you go cloud

Focus on **business** capabilities

will your business benefit?



Focus on **technical** capabilities

are you ready technologically?