Cloud computing

UNIVERSITÉ
CÔTE D'AZUR

Definition and principles

Definition

Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources that can be rapidly provisioned and released with minimal management effort or service provider interaction.

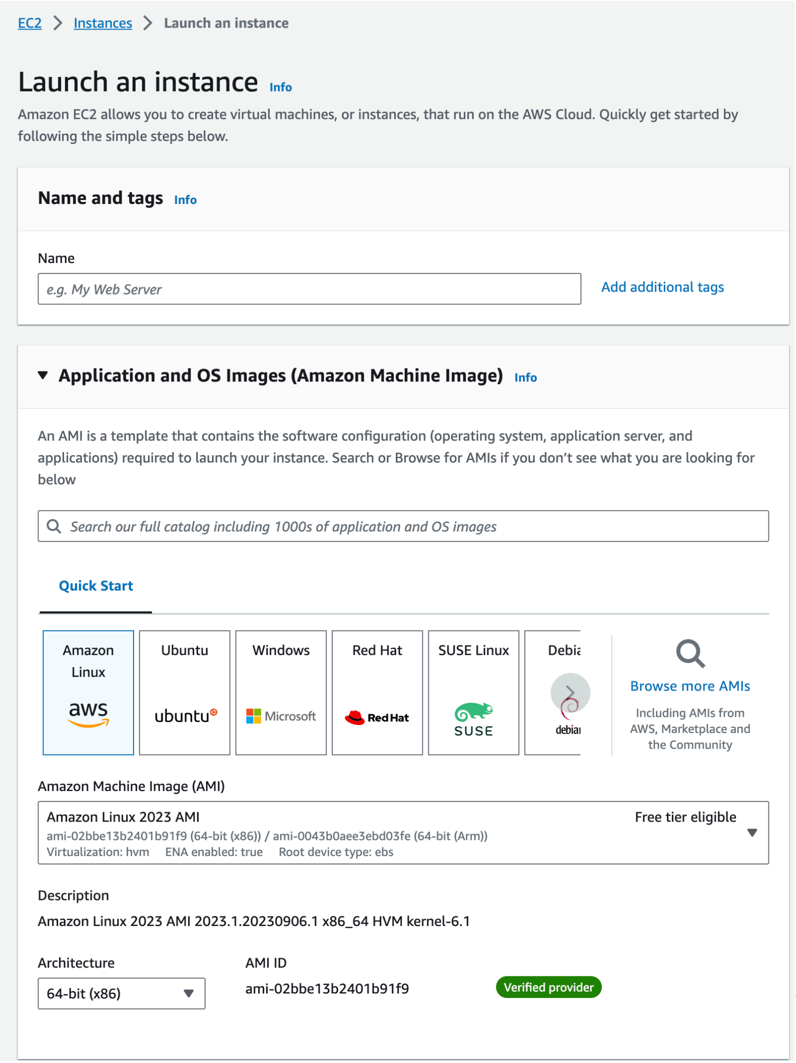
National Institute of Standards and Technology (2011)

5 Principles

- On-demand and self service
- Network access
- Resource pooling
- Rapid elasticity
- Measured service

1- On demand self service

- Self provisioning
  - User can provision resources by themselves
- No need for dedicated IT
- Not even the need for human
  - Add resources with programs/automation



Launch an instance Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags Info

Name: *e.g. My Web Server*

▼ Application and OS Images (Amazon Machine Image) Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Search our full catalog including 1000s of application and OS images

Amazon Machine Image (AMI)

Amazon Linux 2023 AMI — Free tier eligible
ami-02bbe13b2401b91f9 (64-bit (x86)) / ami-0043b0aee3ebd03fe (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs

Description

Amazon Linux 2023 AMI 2023.1.20230906.1 x86_64 HVM kernel-6.1

Architecture: 64-bit (x86)

AMI ID: ami-02bbe13b2401b91f9

2- Broad network access

- The Cloud is accessible from any device anywhere
- WAN
    - Clients and Cloud can be 100s of kms apart
- Standard protocols
    - SSH/SSL
    - HTTP
- Virtual Private network, firewall…

3- Resource pooling

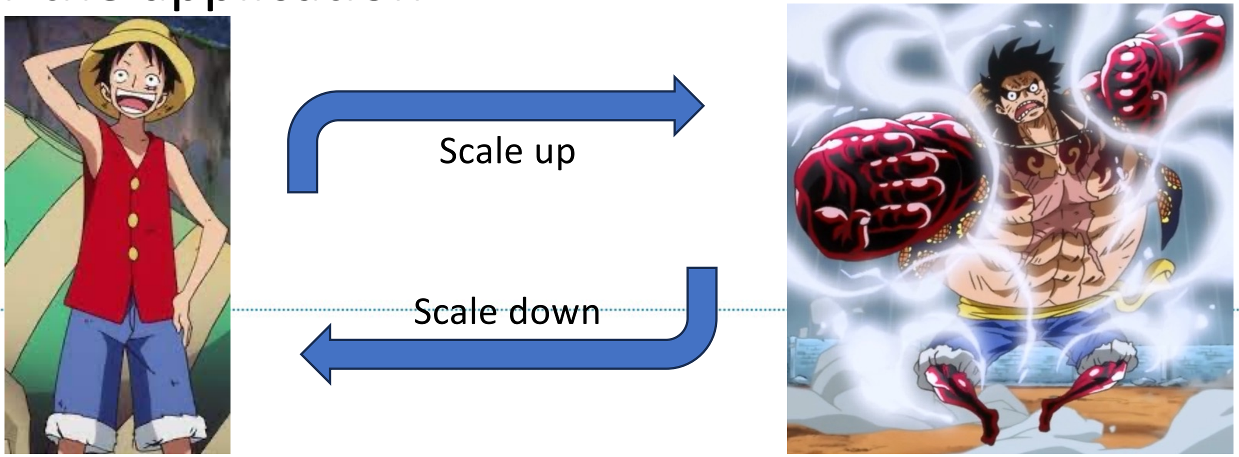
- Multi-tenant
  - Different clients use the same resource
  - Safety is paramount
- Physical or virtual resources
  - Servers vs VM vs Containers
- On-demand allocation
- Location independent
  - Don’t care where the resource is (really?)

4- Rapid elasticity

- Fast (de-)allocation of resources
  - Minutes to add resources vs days/weeks in standard IT
- Scale to hundred of machines (virtual) easily
  - Doesn’t mean applications scale the same
- How to scale apps ?
  - Depends on the architecture
  - Horizontal vs Vertical scaling

Vertical elasticity

- Add/remove resources in a single instance
- Example
  - Add memory, upgrade CPU…
- Hard to do quickly and limited
- But fully transparent for the application

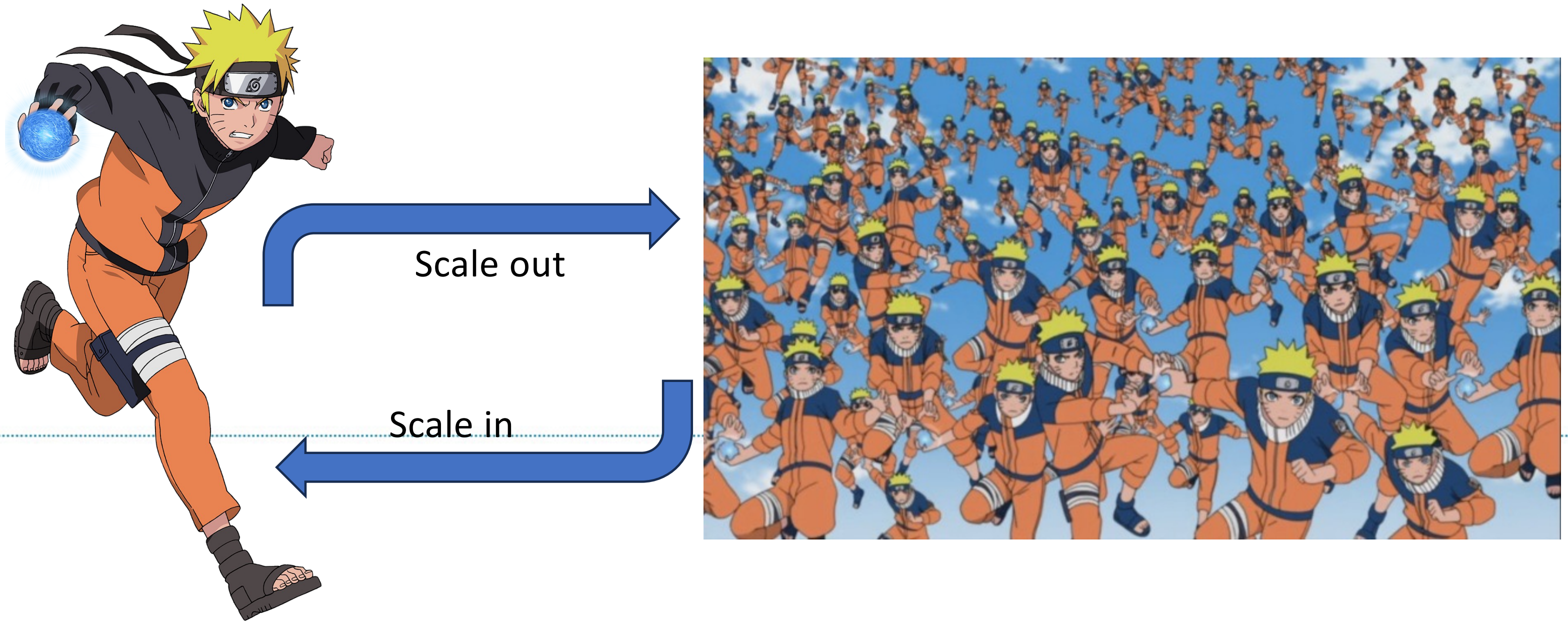


Scale up

Scale down

Horizontal elasticity

- Add/remove instances as needed
- Fast and potentially unlimited scaling
- But need application support



Scale out

Scale in

5- Measured service

- Need to follow resource usage
- Transparent billing for users

as a service

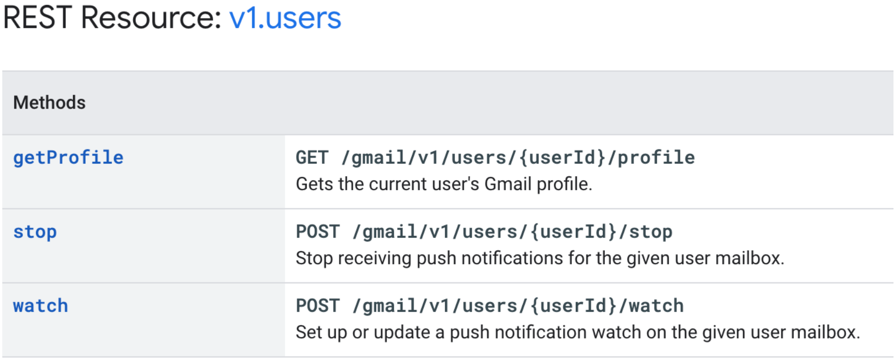
?aaS

Introduction

- Based on the previous 5 principles, many matching services
- E.g :
  - A server you can rent in a datacenter
  - Some services remotely accessible through a remote API
  - Some software hosted by another company
- But very different in practice
  - ? as a Service

Software as a Service

- Web access to some software
- Customers pay for access
  - Upgrades/features are handled by provider
- Sometimes provide an API for integration
- Examples :
  - Gmail
  - Microsoft 365



REST Resource: v1.users

Methods	
`getProfile`	`GET /gmail/v1/users/{userId}/profile` Gets the current user's Gmail profile.
`stop`	`POST /gmail/v1/users/{userId}/stop` Stop receiving push notifications for the given user mailbox.
`watch`	`POST /gmail/v1/users/{userId}/watch` Set up or update a push notification watch on the given user mailbox.

Platform as a Service PaaS

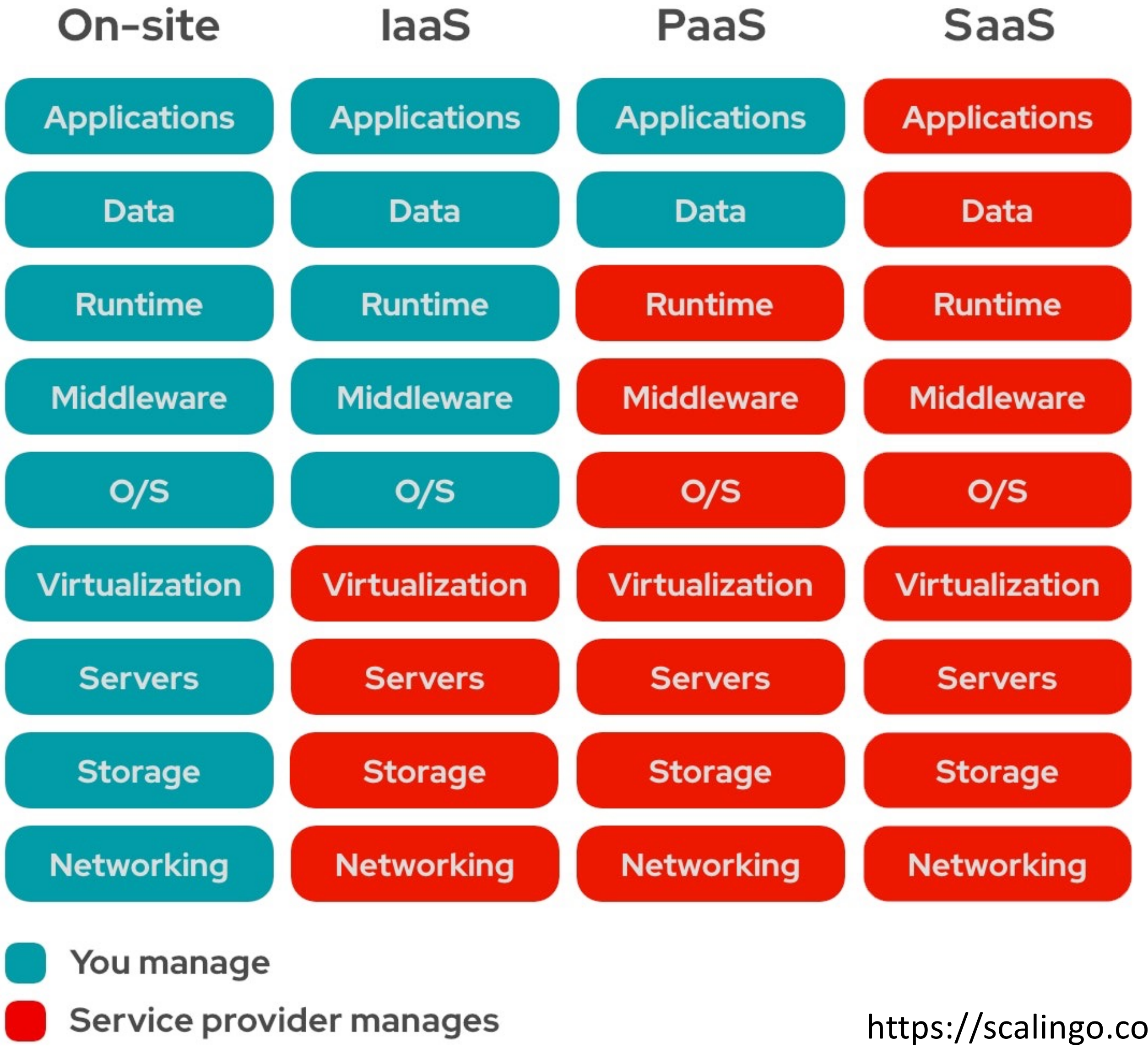
- Jailed runtime to host applications
  - Client write applications using generic or provider-specific API
  - Provider executes applications
  - Applications are accessible over Internet
- No fine control over the environment
  - Client can choose specific runtime versions or memory
  - But cannot install its own versions.
- Example :
  - Google App Engine
  - Aws Elastic Beanstalk

Infrastructure as a Service

- Low level resources to deploy arbitrary software stack
- Client has complete control over
  - Network
  - Storage
  - OS
  - Applications
- Examples
  - Windows Azure
  - Amazon EC2

↳ Elastic Cloud Computing

virtualized



On-site	IaaS	PaaS	SaaS
🟦 Applications	🟦 Applications	🟦 Applications	🟥 Applications
🟦 Data	🟦 Data	🟦 Data	🟥 Data
🟦 Runtime	🟦 Runtime	🟥 Runtime	🟥 Runtime
🟦 Middleware	🟦 Middleware	🟥 Middleware	🟥 Middleware
🟦 O/S	🟦 O/S	🟥 O/S	🟥 O/S
🟦 Virtualization	🟥 Virtualization	🟥 Virtualization	🟥 Virtualization
🟦 Servers	🟥 Servers	🟥 Servers	🟥 Servers
🟦 Storage	🟥 Storage	🟥 Storage	🟥 Storage
🟦 Networking	🟥 Networking	🟥 Networking	🟥 Networking

🟦 You manage

🟥 Service provider manages

vendor lock-in

https://scalingo.com/blog/paas-guide-complet

Virtualization

- Virtualization is a key component of Cloud Computing
- Lots of benefits
- Sharing a single server with multiple tenants
- Server isolation
- Ease of upgrade/migration
- Ease of backup