

ĐIỆN TOÁN ĐÁM MÂY

Bài 6. Cloud Infrastructure

Nguyễn Thành Trung
trung.nguyenthanh@phenikaa-uni.edu.vn

Contents

- IaaS services from Amazon.
 - Regions and availability zones for Amazon Web Services.
 - Instances – attributes and cost.
 - A repertoire of Amazon Web Services.
- SaaS and PaaS services from Google.
- SaaS and PaaS services from Microsoft.
- Open-source platforms for private clouds.

Existing cloud infrastructure

- The cloud computing infrastructure at Amazon, Google, and Microsoft (as of mid 2012).
 - Amazon is a pioneer in Infrastructure-as-a-Service (IaaS).
 - Google's efforts are focused on Software-as-a-Service (SaaS) and Platform-as-a-Service (PaaS).
 - Microsoft is involved in PaaS.
- Private clouds are an alternative to public clouds. Open-source cloud computing platforms such as:
 - Eucalyptus,
 - OpenNebula,
 - Nimbus,
 - OpenStack

can be used as a control infrastructure for a private cloud.

Amazon Web Services (AWS)

- AWS → IaaS cloud computing services launched in 2006.
- Businesses in 200 countries used AWS in 2012.
- The infrastructure consists of computing and storage servers interconnected by high-speed networks and supports a set of services.
- An application developer:
 - Installs applications on a platform of his/her choice.
 - Manages resources allocated by Amazon.

AWS regions and availability zones

- Amazon offers cloud services through a network of data centers on several continents.
- In each *region* there are several availability zones interconnected by high-speed networks.
- An *availability zone* is a data center consisting of a large number of servers.

Region	Location	Availability zones	Cost
US West	Oregon	us-west-2a/2b/2c	Low
US West	North California	us-west-1a/1b/1c	High
US East	North Virginia	us-east-1a/2a/3a/4a	Low
Europe	Ireland	eu-west-1a/1b/1c	Medium
South America	Sao Paulo, Brazil	sa-east-1a/1b	Very high
Asia Pacific	Tokyo, Japan	ap-northeast-1a/1b	High
Asia Pacific	Singapore	ap-southeast-1a/1b	Medium

- Regions do not share resources and communicate through the Internet.



Projection Type

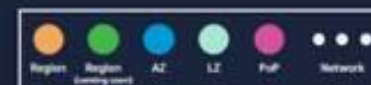


DISCOVER HOW WE DO IT

- Home >>
- Global Infrastructure >>
- Regions >>
- Availability Zones >>>
- Local Zones >>>
- Points of Presence >>>
- Network >>>
- Custom Hardware >>>
- Benefits >>



Key



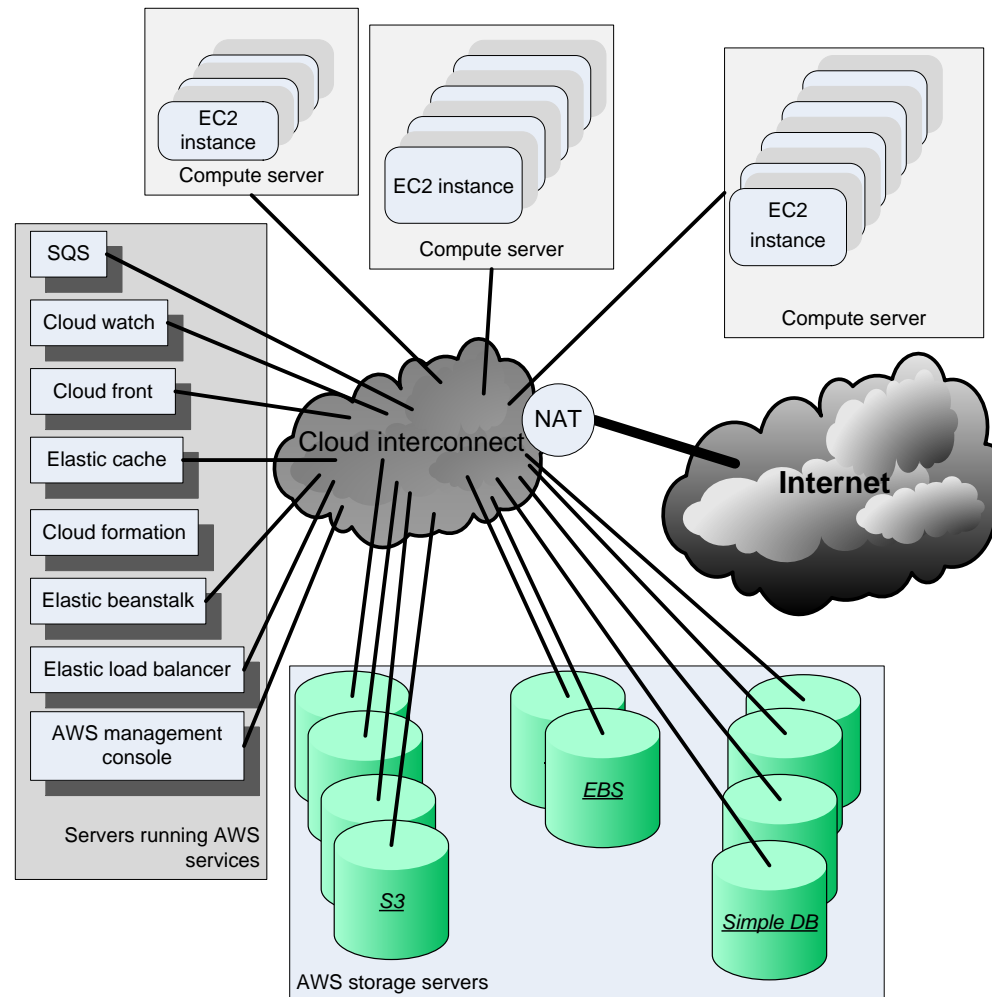


AWS instances

- An instance is a virtual server with a well specified set of resources, including CPU cycles, main memory, secondary storage, communication, and I/O bandwidth.
- The user chooses:
 - The region and the availability zone where this virtual server should be placed.
 - An instance type from a limited menu of instance types.
- When launched, an instance is provided with a DNS name; this name maps to a
 - *private IP address* → for internal communication within the internal EC2 communication network.
 - *public IP address* → for communication outside the internal Amazon network, e.g., for communication with the user that launched the instance.

AWS instances (cont'd)

- Network Address Translation (NAT) maps external IP addresses to internal ones.
- The public IP address is assigned for the lifetime of an instance.
- An instance can request an *elastic IP address*, rather than a public IP address. The elastic IP address is a static public IP address allocated to an instance from the available pool of the availability zone.
- An elastic IP address is not released when the instance is stopped or terminated and must be released when no longer needed.



Steps to run an application

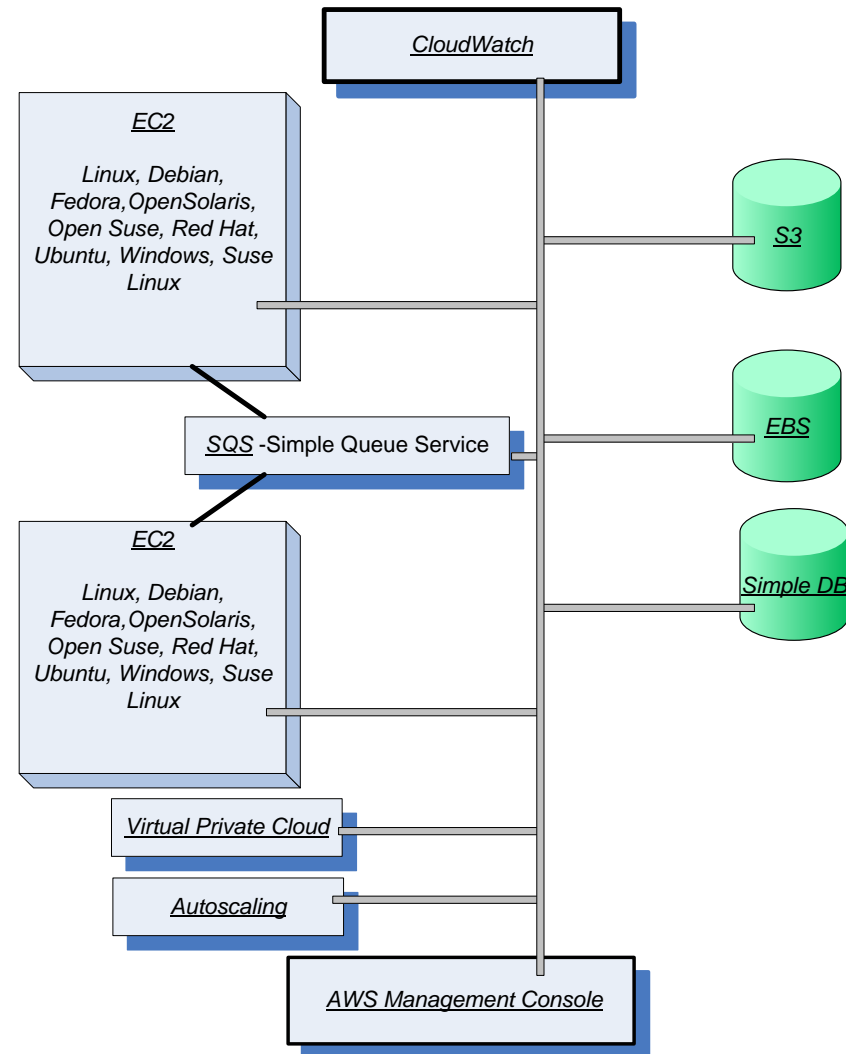
- Retrieve the user input from the front-end.
- Retrieve the disk image of a VM (Virtual Machine) from a repository.
- Locate a system and requests the VMM (Virtual Machine Monitor) running on that system to setup a VM.
- Invoke the Dynamic Host Configuration Protocol (DHCP) and the IP bridging software to set up MAC and IP addresses for the VM.

User interactions with AWS

- The AWS Management Console. The easiest way to access all services, but not all options may be available.
- AWS SDK libraries and toolkits are provided for several programming languages including Java, PHP, C#, and Objective-C.
- Raw REST requests.

Examples of Amazon Web Services

- *AWS Management Console* - allows users to access the services offered by AWS.
- *Elastic Cloud Computing (EC2)* - allows a user to launch various operating systems.
- *Simple Queuing Service (SQS)* – allows multiple *EC2* instances to communicate with one another.
- *Simple Storage Service (S3)*, *Simple DB*, and *Elastic Bloc Storage (EBS)* - storage services.
- *Cloud Watch* - supports performance monitoring.
- *Auto Scaling* - supports elastic resource management.
- *Virtual Private Cloud* - allows direct migration of parallel applications.



EC2 – Elastic Cloud Computing

- *EC2* - web service for launching instances of an application under several operating systems, such as:
 - Several Linux distributions.
 - Microsoft Windows Server 2003 and 2008.
 - OpenSolaris.
 - FreeBSD.
 - NetBSD.
- A user can
 - Load an *EC2* instance with a custom application environment.
 - Manage network's access permissions.
 - Run the image using as many or as few systems as desired.

EC2 (cont'd)

- Import virtual machine (VM) images from the user environment to an instance through *VM import*.
- *EC2* instances boot from an AMI (Amazon Machine Image) digitally signed and stored in S3.
- Users can access:
 - Images provided by Amazon.
 - Customize an image and store it in S3.
- An *EC2* instance is characterized by the resources it provides:
 - VC (Virtual Computers) – virtual systems running the instance.
 - CU (Compute Units) – measure computing power of each system.
 - Memory.
 - I/O capabilities.

Instance types

- Standard instances: micro (StdM), small (StdS), large (StdL), extra large (StdXL); small is the default.
- High memory instances: high-memory extra large (HmXL), high-memory double extra large (Hm2XL), and high-memory quadruple extra large (Hm4XL).
- High CPU instances: high-CPU extra large (HcpuXL).
- Cluster computing: cluster computing quadruple extra large (Cl4XL).

Instance name	API name	Platform (32/64-bit)	Memory (GB)	Max EC2 compute units	I-memory (GB)	I/O (M/H)
StdM		32 and 64	0.633	1 VC; 2 CUs		
StdS	m1.small	32	1.7	1 VC; 1 CU	160	M
StdL	m1.large	64	7.5	2 VCs; 2 × 2 CUs	85	H
StdXL	m1.xlarge	64	15	4 VCs; 4 × 2 CUs	1,690	H
HmXL	m2.xlarge	64	17.1	2 VCs; 2 × 3.25 CUs	420	M
Hm2XL	m2.2xlarge	64	34.2	4 VCs; 4 × 3.25 CUs	850	H
Hm4XL	m2.4xlarge	64	68.4	8 VCs; 8 × 3.25 CUs	1,690	H
HcpuXL	c1.xlarge	64	7	8 VCs; 8 × 2.5 CUs	1,690	H
Cl4XL	cc1.4xlarge	64	18	33.5 CUs	1,690	H

Instance cost

- A main attraction of Amazon cloud computing is the low cost.

< PAGE CONTENT

On-Demand Pricing

Data Transfer

Data Transfer within the same AWS Region

EBS-Optimized Instances

Elastic IP Addresses

Carrier IP Addresses

Elastic Load Balancing

On-Demand Capacity Reservations

T2/T3/T4g Unlimited Mode Pricing

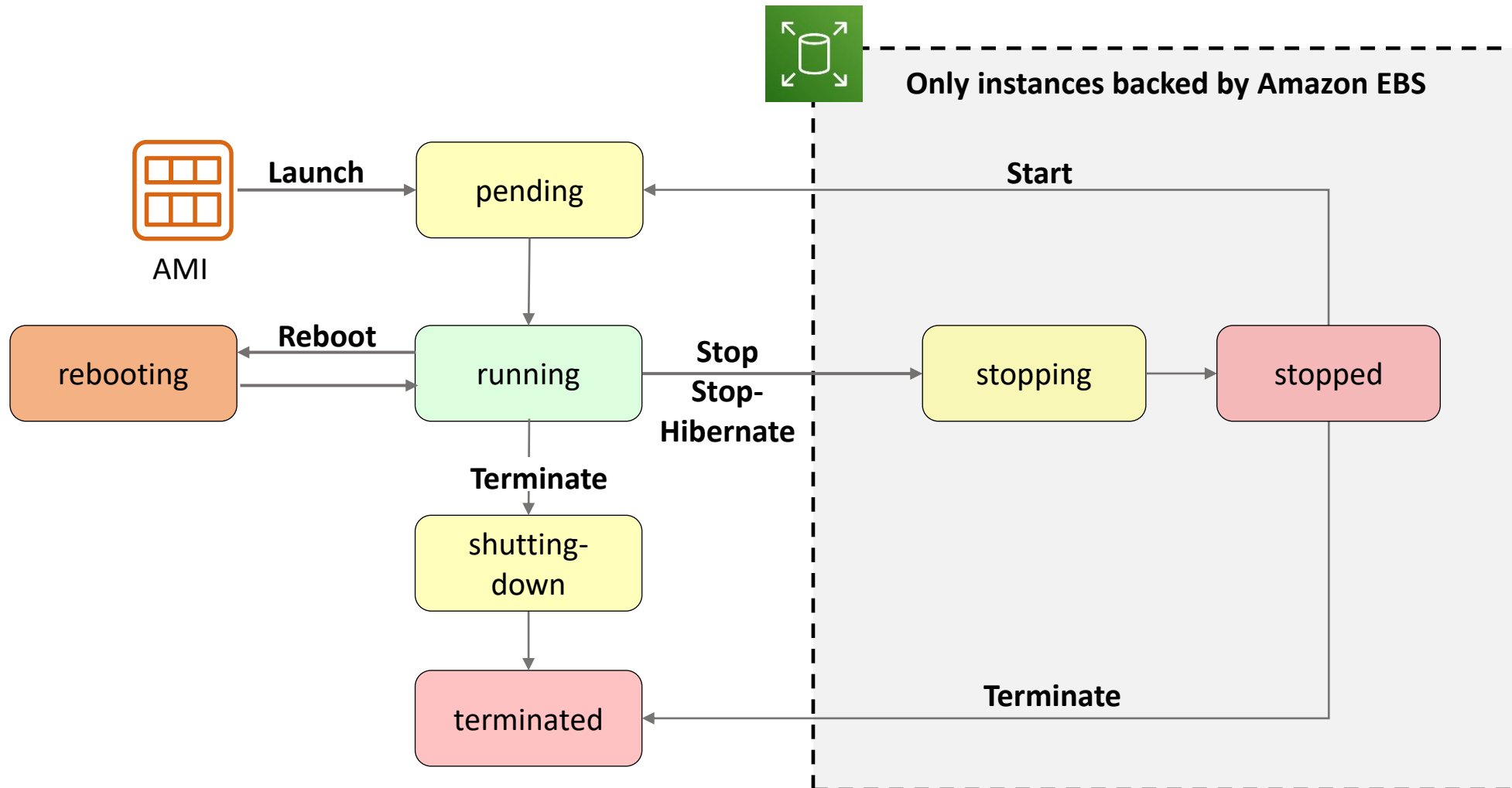
Amazon CloudWatch

Amazon Elastic Block Store

Viewing 489 of 489 available instances

<input type="text"/>						< 1 2 3 4 5 6 7 ... 25 >
Instance name ▲	On-Demand hourly rate ▼	vCPU ▼	Memory ▼	Storage ▼	Network performance ▼	
a1.medium	\$0.0255	1	2 GiB	EBS Only	Up to 10 Gigabit	
a1.large	\$0.051	2	4 GiB	EBS Only	Up to 10 Gigabit	
a1.xlarge	\$0.102	4	8 GiB	EBS Only	Up to 10 Gigabit	
a1.2xlarge	\$0.204	8	16 GiB	EBS Only	Up to 10 Gigabit	
a1.4xlarge	\$0.408	16	32 GiB	EBS Only	Up to 10 Gigabit	
a1.metal	\$0.408	16	32 GiB	EBS Only	Up to 10 Gigabit	
t4g.nano	\$0.0042	2	0.5 GiB	EBS Only	Up to 5 Gigabit	
t4g.micro	\$0.0084	2	1 GiB	EBS Only	Up to 5 Gigabit	

Amazon EC2 instance lifecycle



S3 – Simple Storage System

- Service designed to store large objects; an application can handle an unlimited number of objects ranging in size from 1 byte to 5 TB.
- An object is stored in a bucket and retrieved via a unique, developer-assigned key; a bucket can be stored in a Region selected by the user.
- Supports a minimal set of functions: write, read, and delete; it does not support primitives to copy, to rename, or to move an object from one bucket to another.
- The object names are global.
- S3 maintains for each object: the name, modification time, an access control list, and up to 4 KB of user-defined metadata.

S3 (cont'd)

- Authentication mechanisms ensure that data is kept secure.
- Objects can be made public, and rights can be granted to other users.
- S3 computes the MD5 of every object written and returns it in a field called ETag.
- A user is expected to compute the MD5 of an object stored or written and compare this with the ETag; if the two values do not match, then the object was corrupted during transmission or storage.

Elastic Block Store (EBS)

- Provides persistent block level storage volumes for use with *EC2* instances; suitable for database applications, file systems, and applications using raw data devices.
- A volume appears to an application as a raw, unformatted and reliable physical disk; the range 1 GB -1 TB.
- An *EC2* instance may mount multiple volumes, but a volume cannot be shared among multiple instances.
- EBS supports the creation of snapshots of the volumes attached to an instance and then uses them to restart the instance.
- The volumes are grouped together in Availability Zones and are automatically replicated in each zone.

SimpleDB

- Non-relational data store. Supports store and query functions traditionally provided only by relational databases.
- Supports high performance Web applications; users can store and query data items via Web services requests.
- Creates multiple geographically distributed copies of each data item.
- It manages automatically:
 - The infrastructure provisioning.
 - Hardware and software maintenance.
 - Replication and indexing of data items.
 - Performance tuning.

SQS - Simple Queue Service

- Hosted message queues are accessed through standard SOAP and Query interfaces.
- Supports automated workflows - *EC2* instances can coordinate by sending and receiving SQS messages.
- Applications using SQS can run independently and asynchronously and do not need to be developed with the same technologies.
- A received message is "locked" during processing; if processing fails, the lock expires, and the message is available again.
- Queue sharing can be restricted by IP address and time-of-day.

CloudWatch

- Monitoring infrastructure used by application developers, users, and system administrators to collect and track metrics important for optimizing the performance of applications and for increasing the efficiency of resource utilization.
- Without installing any software a user can monitor either seven or eight pre-selected metrics and then view graphs and statistics for these metrics.
- When launching an Amazon Machine Image (AMI) the user can start the CloudWatch and specify the type of monitoring:
 - Basic Monitoring - free of charge; collects data at five-minute intervals for up to seven metrics.
 - Detailed Monitoring - subject to charge; collects data at one minute interval.

AWS services introduced in 2012

- *Route 53* - low-latency DNS service used to manage user's DNS public records.
- *Elastic MapReduce (EMR)* - supports processing of large amounts of data using a hosted Hadoop running on *EC2*.
- *Simple Workflow Service (SWF)* - supports workflow management; allows scheduling, management of dependencies, and coordination of multiple *EC2* instances.
- *ElastiCache* - enables web applications to retrieve data from a managed in-memory caching system rather than a much slower disk-based database.
- *DynamoDB* - scalable and low-latency fully managed NoSQL database service.

AWS services introduced in 2012 (cont'd)

- *CloudFront* - web service for content delivery.
- *Elastic Load Balancer* - automatically distributes the incoming requests across multiple instances of the application.
- *Elastic Beanstalk* - handles automatically deployment, capacity provisioning, load balancing, auto-scaling, and application monitoring functions.
- *CloudFormation* - allows the creation of a stack describing the infrastructure for an application.

SaaS services offered by Google

- *Gmail* - hosts Emails on Google servers and provides a web interface to access the Email.
- *Google docs* - a web-based software for building text documents, spreadsheets and presentations.
- *Google Calendar* - a browser-based scheduler; supports multiple user calendars, calendar sharing, event search, display of daily/weekly/monthly views, and so on.
- *Google Groups* - allows users to host discussion forums to create messages online or via Email.
- *Picasa* - a tool to upload, share, and edit images.
- *Google Maps* - web mapping service; offers street maps, a route planner, and an urban business locator for numerous countries around the world

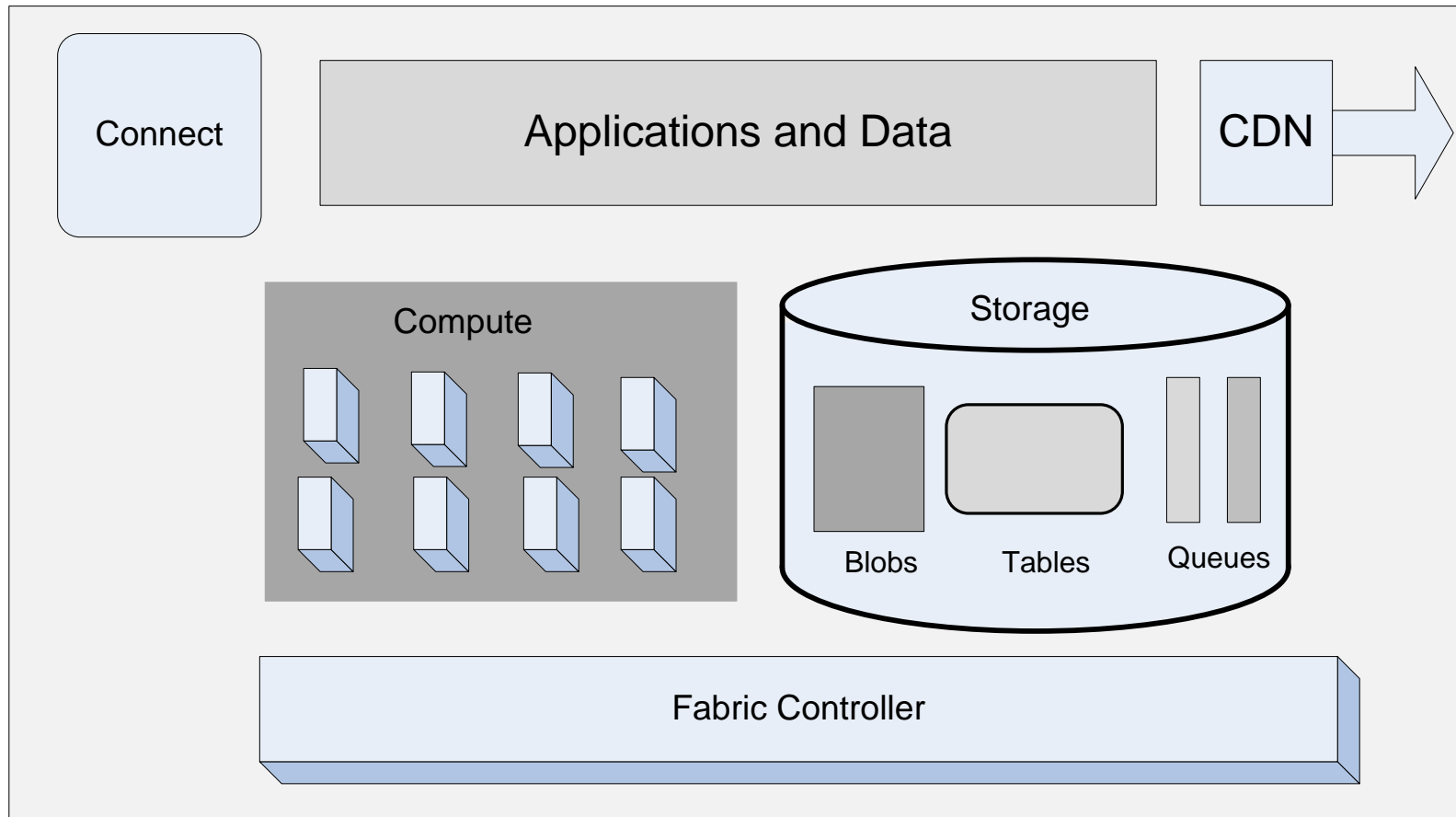
PaaS services offered by Google

- *AppEngine* - a developer platform hosted on the cloud.
 - Initially supported Python, Java was added later.
 - The database for code development can be accessed with GQL (Google Query Language) with a SQL-like syntax.
- *Google Co-op* - allows users to create customized search engines based on a set of facets/categories.
- *Google Drive* - an online service for data storage.
- *Google Base* - allows users to load structured data from different sources to a central repository, a very large, self-describing, semi-structured, heterogeneous database.

PaaS and SaaS services from Microsoft

- *Windows Azure* - an operating system; has 3 components:
 - Compute - provides a computation environment.
 - Storage - for scalable storage.
 - Fabric Controller - deploys, manages, and monitors applications.
- *SQL Azure* - a cloud-based version of the SQL Server.
- *Azure AppFabric*, formerly .NET Services - a collection of services for cloud applications.

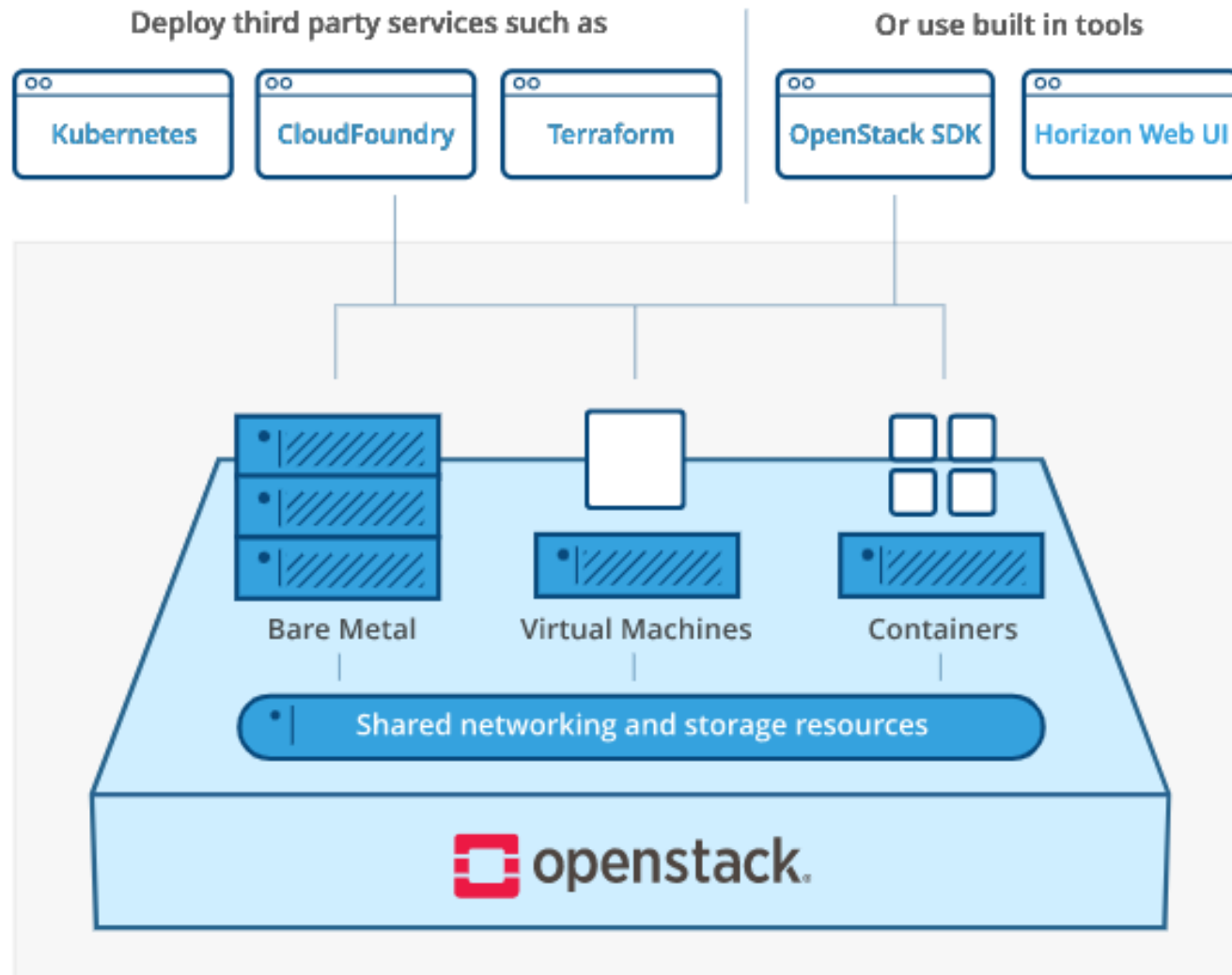
Azure

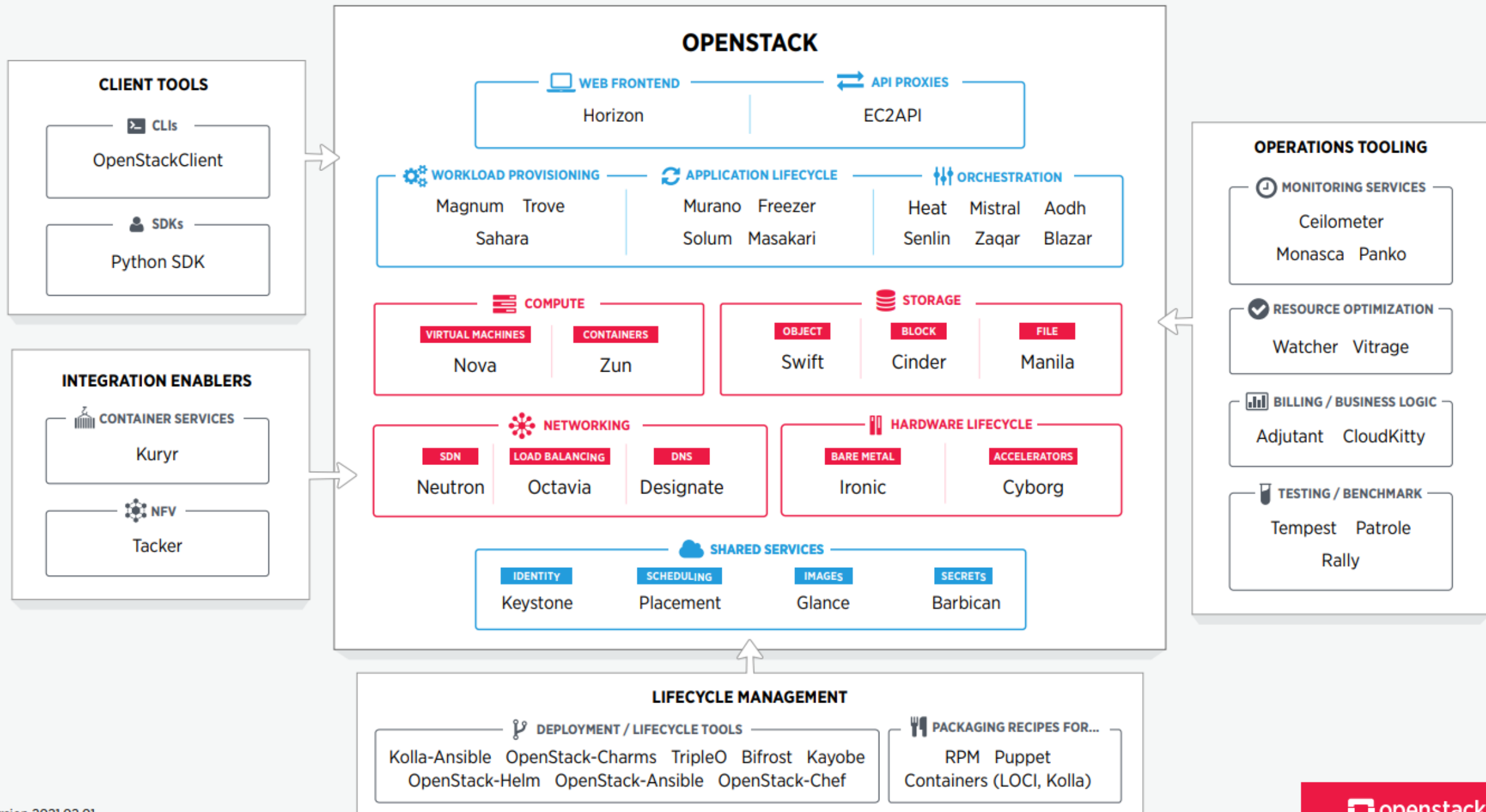


Open-source platforms for clouds

- *Eucalyptus* - can be regarded as an open-source counterpart of Amazon's EC2.
- *Open-Nebula* - a private cloud with users actually logging into the head node to access cloud functions. The system is centralized and its default configuration uses the NFS file system.
- *Openstack* - Cloud Infrastructure for Virtual Machines, Bare Metal, and Containers

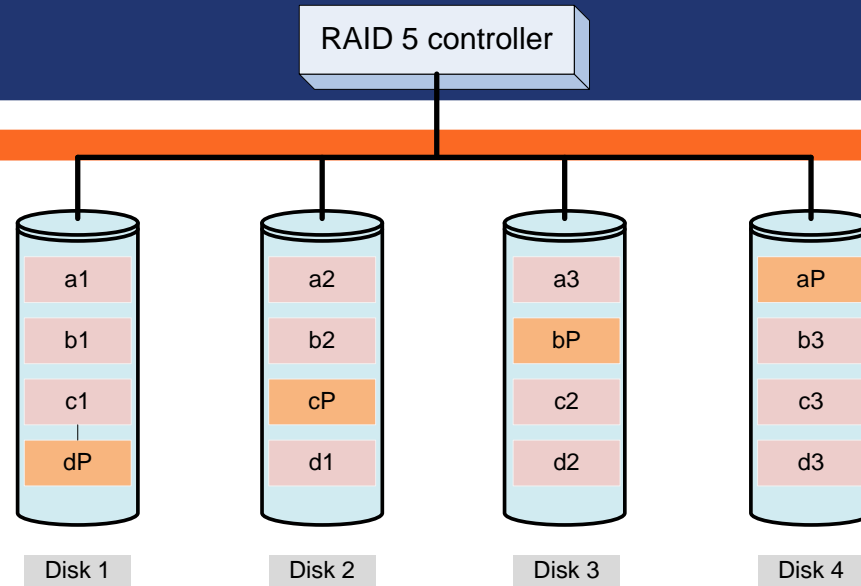
Openstack



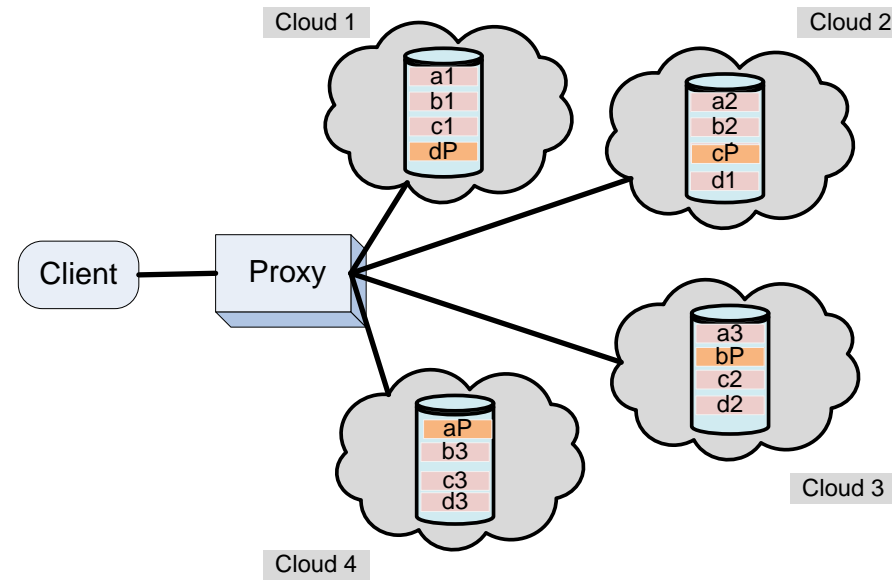


Cloud storage diversity and vendor lock-in

- Risks when a large organization relies on a single cloud service provider:
 - Cloud services may be unavailable for a short or an extended period of time.
 - Permanent data loss in case of a catastrophic system failure.
 - The provider may increase the prices for service.
- Switching to another provider could be very costly due to the large volume of data to be transferred from the old to the new provider.
- A solution is to replicate the data to multiple cloud service providers, similar to data replication in RAID.



(a)



(b)

Cloud interoperability; the Intercloud

- An Intercloud → a federation of clouds that cooperate to provide a better user experience.
- Is an Intercloud feasible?
- Not likely at this time:
 - There are no standards for either storage or processing.
 - The clouds are based on different delivery models.
 - The set of services supported by these delivery models is large and open; new services are offered every few months.
 - CSPs (Cloud Service Providers) believe that they have a competitive advantage due to the uniqueness of the added value of their services.
 - Security is a major concern for cloud users and an Intercloud could only create new threats.

Energy use and ecological impact

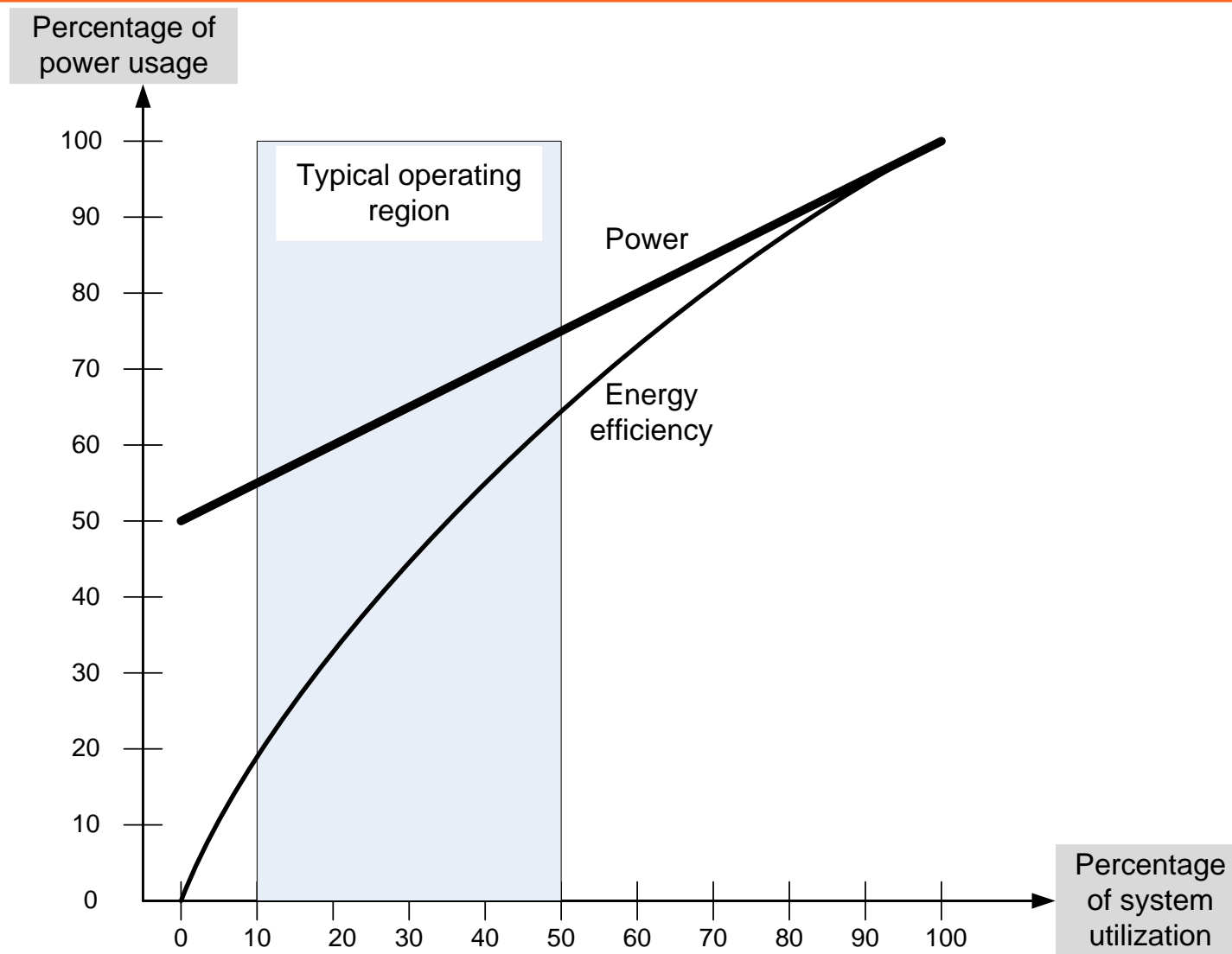
- The energy consumption of large-scale data centers and their costs for energy and for cooling are significant.
- In 2006, the 6,000 data centers in the U.S consumed 61×10^9 KWh of energy, 1.5% of all electricity consumption, at a cost of \$4.5 billion.
- The energy consumed by the data centers was expected to double from 2006 to 2011 and peak instantaneous demand to increase from 7 GW to 12 GW.
- The greenhouse gas emission due to the data centers is estimated to increase from 116×10^9 tones of CO_2 in 2007 to 257 tones in 2020 due to increased consumer demand.
- The effort to reduce energy use is focused on computing, networking, and storage activities of a data center.

Energy use and ecological impact (cont'd)

- Operating efficiency of a system is captured by the *performance per Watt of power*.
- The performance of supercomputers has increased 3.5 times faster than their operating efficiency – 7,000% versus 2,000% during the period 1998 – 2007.
- A typical Google cluster spends most of its time within the 10-50% CPU utilization range; there is a mismatch between server workload profile and server energy efficiency.

Energy-proportional systems

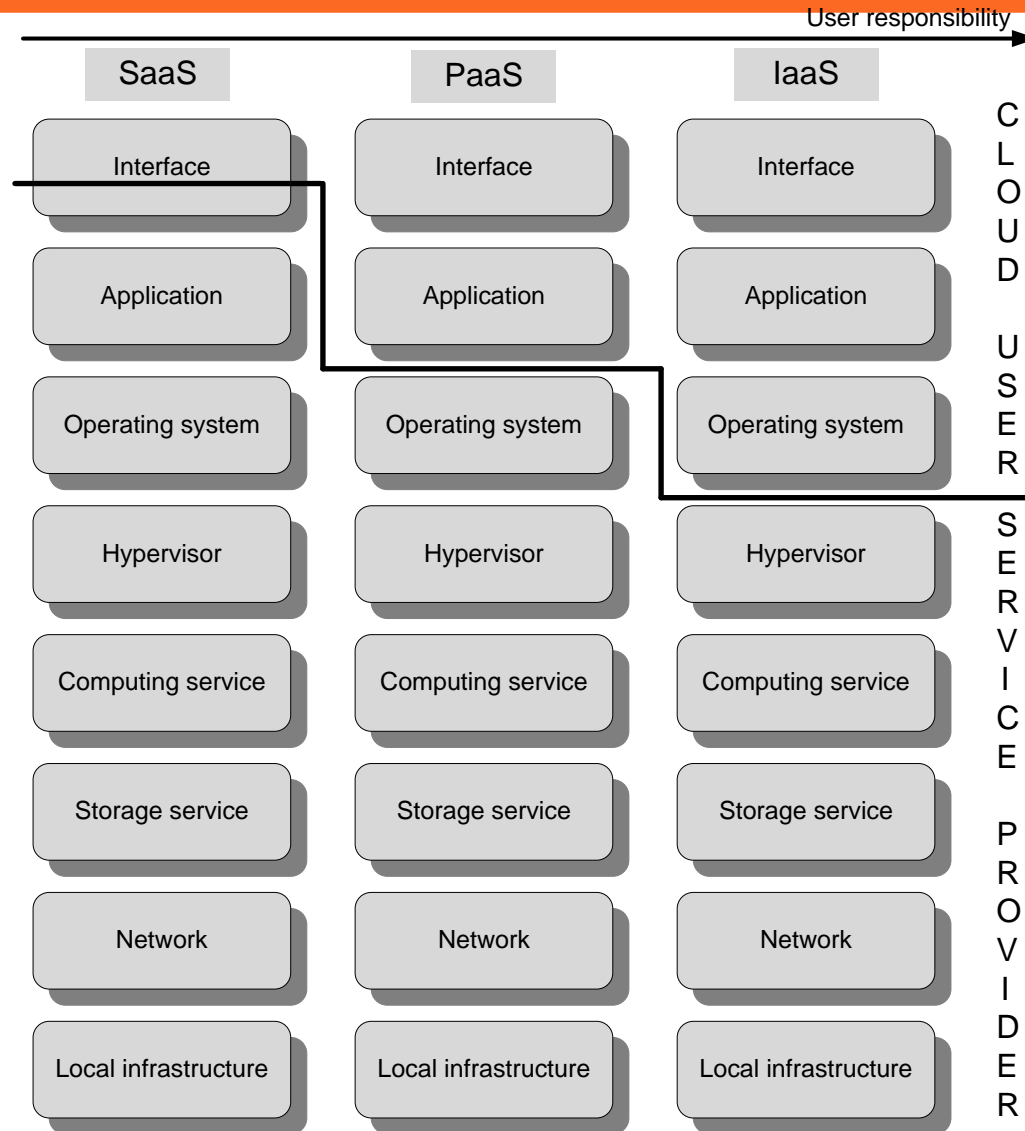
- An energy-proportional system consumes no power when idle, very little power under a light load and, gradually, more power as the load increases.
- By definition, an ideal energy-proportional system is always operating at 100% efficiency.
- Humans are a good approximation of an ideal energy proportional system; about 70 W at rest, 120 W on average on a daily basis, and can go as high as 1,000 – 2,000 W during a strenuous, short time effort.
- Even when power requirements scale linearly with the load, the energy efficiency of a computing system is not a linear function of the load; even when idle, a system may use 50% of the power corresponding to the full load.



Service Level Agreement (SLA)

- SLA - a negotiated contract between the customer and CSP; can be legally binding or informal.
Objectives:
 - Identify and define the customer's needs and constraints including the level of resources, security, timing, and QoS.
 - Provide a framework for understanding; a critical aspect of this framework is a clear definition of classes of service and the costs.
 - Simplify complex issues; clarify the boundaries between the responsibilities of clients and CSP in case of failures.
 - Reduce areas of conflict.
 - Encourage dialog in the event of disputes.
 - Eliminate unrealistic expectations.
- Specifies the services that the customer receives, rather than how the cloud service provider delivers the services.

Responsibility sharing between user and CSP



User security concerns

- Potential loss of control/ownership of data.
- Data integration, privacy enforcement, data encryption.
- Data remanence after de-provisioning.
- Multi tenant data isolation.
- Data location requirements within national borders.
- Hypervisor security.
- Audit data integrity protection.
- Verification of subscriber policies through provider controls.
- Certification/Accreditation requirements for a given cloud service.

Reasons driving decision to use public clouds

Reason	Percentage who agree
Improved system reliability and availability	50%
Pay only for what you use	50%
Hardware savings	47%
Software license saving	46%
Lower labor costs	44%
Lower maintenance costs	42%
Reduced IT support needs	40%
Ability to take advantage of the latest functionality	40%
Less pressure on internal resources	39%
Solve problems related to updating/upgrading	39%
Rapid deployment	39%
Ability to scale up resources to meet the needs	39%
Ability to focus on core competencies	38%
Take advantage of the improved economics of scale	37%
Reduced infrastructure management needs	37%
Lower energy costs	29%
Reduced space requirements	26%
Create new revenue streams	23%

Activities

- **Module 6 - Compute**
 - Lab 3 - Introduction to Amazon EC2
 - Module 6 Knowledge Check

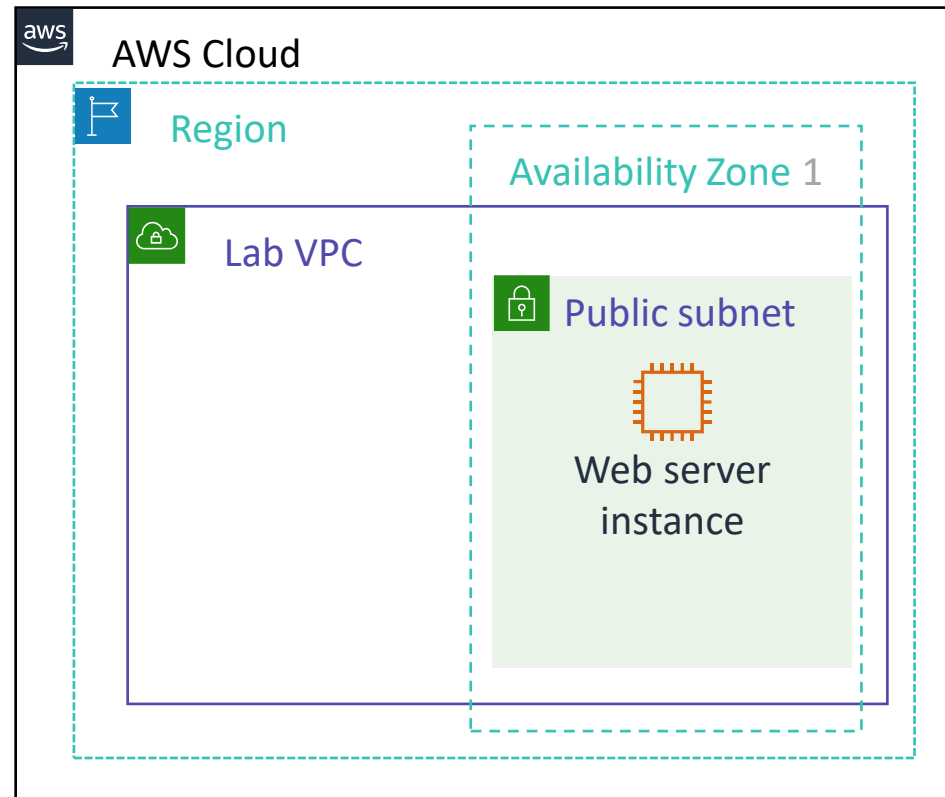
Lab 3:

Introduction to Amazon EC2



Lab 3 scenario

In this lab, you will launch and configure your first virtual machine that runs on Amazon EC2.



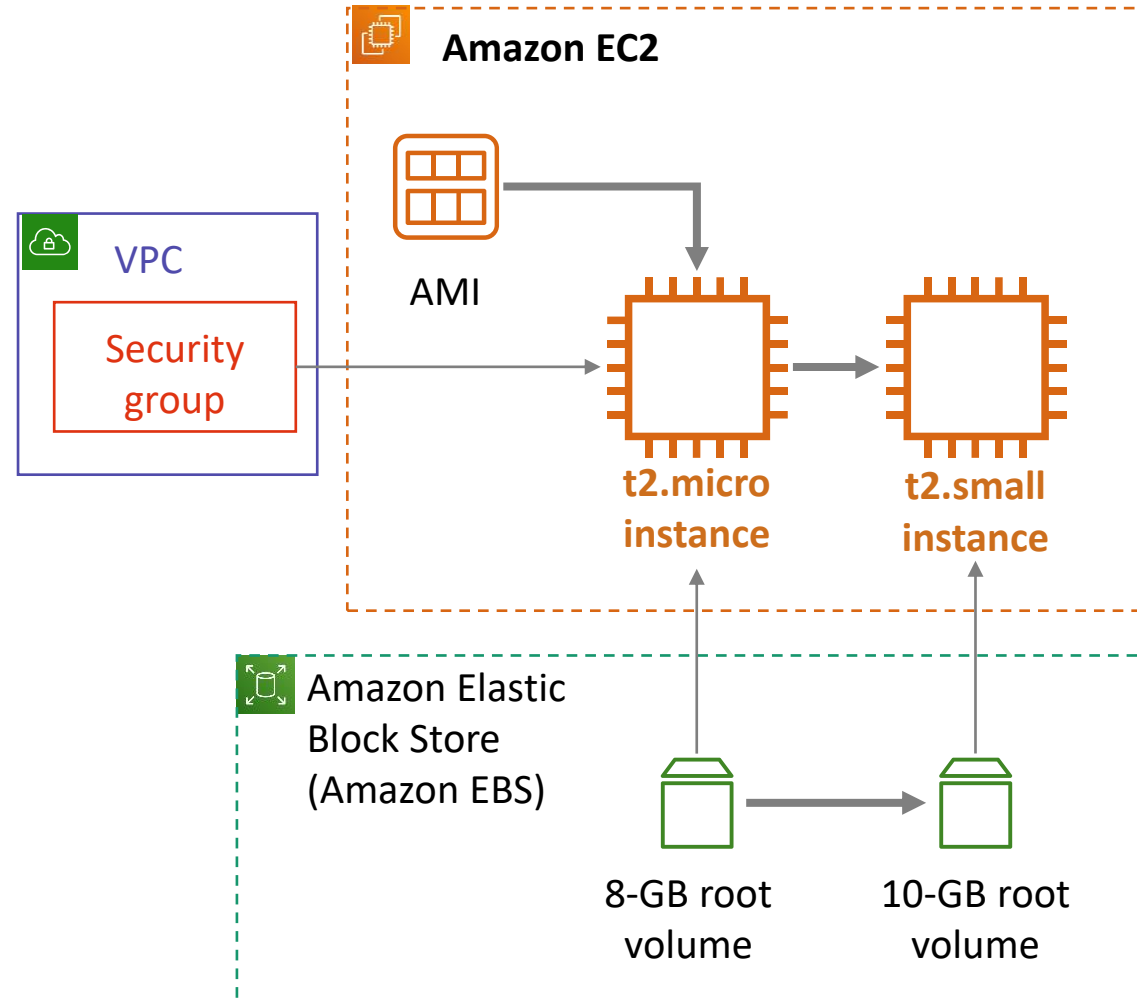
Lab 3: Tasks

- Task 1 – Launch Your Amazon EC2 Instance
- Task 2 – Monitor Your Instance
- Task 3 – Update Your Security Group and Access the Web Server
- Task 4 – Resize Your Instance: Instance Type and EBS Volume
- Task 5 – Explore EC2 Limits
- Task 6 – Test Termination Protection

Lab 3: Final product

By the end of the lab, you will have:

1. Launched an instance that is configured as a web server
2. Viewed the instance system log
3. Reconfigured a security group
4. Modified the instance type and root volume size



Complete the knowledge check

