**Weekly Questions**

**Week 1**

**Question 1: The evolution of Cloud Computing has been compared to the evolution of electricity supply as a utility. Describe specific problems that Cloud Computing solves as compared to businesses running their own data centres.**

* Analogy to 'classical' utilities (electricity, water, ...)
* No up-front investment (pay-as-you-go model)
* Low price due to economies of scale
* Elasticity - can quickly scale up/down as demand varies
* Flexibility in operations
* Increase in Transparency
* Enhanced Risk management
* Improved Security

For Detailed Explanation, refer to this <https://www.orchestrate.com/blog/five-business-problems-that-cloud-computing-can-solve/>

**Question 2: Describe the different categories of services (XaaS) cloud computing can provide with specific examples of each service.**

Xaas is a collective term that refers to the delivery of anything as a service. It encompasses the many products, tools and technologies that vendors deliver to users as a service over a network typically the internet -- as an alternative to providing them locally or on-site to an enterprise.

The three main ones are SaaS, platform as a service (PaaS) and infrastructure as a service (IaaS).

1. **SaaS** is a cloud-based software distribution model in which a cloud provider hosts applications and provides them to users over the internet as a service. It is one of the three main cloud computing deployment models and a commonly referenced one.

Examples:

**SaaS** includes a range of applications, such as [Google Apps](https://www.techtarget.com/searchmobilecomputing/definition/Google-Apps-for-Business), [Microsoft Office 365](https://www.techtarget.com/searchenterprisedesktop/definition/Microsoft-Office-365-suite) and [Salesforce](https://www.techtarget.com/searchcustomerexperience/definition/Salesforcecom).

2. **PaaS** providers deliver hardware and software tools to users over the internet.

Examples:

PaaS offerings, such as Amazon Web Services (AWS) Elastic Beanstalk, Apache Stratos, Google App Engine and Salesforce's Heroku and Salesforce Platform

3. **IaaS** is a model where a cloud provider provides IT infrastructure, such as storage, server and networking resources, over the internet to customers on a subscription basis .

Examples :

AWS Elastic Compute Cloud, Google Compute Engine and Microsoft Azure.

**Week 2**

**Question 1: Describe the steps which you would take on AWS and the decisions that would need to be made to create, configure and run a Virtual Machine Instance**

NOTE: Write steps (headings) from Lab 1 & Lab 2

Lab 1 – Setting up the workspace

* Have an AWS account
* Create access key and make note of Access Key and Secret Access Key
* Install and set up Virtual Machine such as VirtualBoxVM
* Install Ubuntu and set this up inside the VM
* Install Python, AWSCLI, Boto3

Lab 2 – Preparation and Creation of VM Instance (EC2) using AWSCLI

(NOTE: Can choose any methods such as AWSCLI or Python Boto Script)

* Create a security group
* Authorize inbound traffic for ssh
* Create a key pair to allow ssh to the EC2 instance
* Create the instance and note the instance id
* Get the public IP address from the instance id
* Connect to the instance

**Question 2: Describe EBS and what features it offers.**

* AWS Elastic Block Store (EBS) is Amazon’s block-level storage solution used with the EC2 cloud service to store persistent data.
* EBS volumes that are attached to an instance are exposed as storage volumes that persist independently from the life of the instance.
* EBS volumes behave like raw, unformatted block devices.
* •EBS storage is allocated in volumes (virtual disk)
* •Can be attached to an instance (but only one at a time). You can mount these volumes as devices on your instances.
* •A single instance can access multiple volumes
* You create an EBS volume in a specific Availability Zone, and then attach it to an instance in that same Availability Zone.
* •Replicated across multiple servers - Data is not lost if a single server fails
* To make a volume available outside of the Availability Zone, you can create a snapshot and restore that snapshot to a new volume anywhere in that Region.
* Amazon EBS provides the following volume types: General Purpose SSD, Provisioned IOPS SSD, Throughput Optimized HDD, and Cold HDD.
* You can dynamically change the configuration of a volume attached to an instance.
* You can create your EBS volumes as encrypted volumes, in order to meet a wide range of data-at-rest encryption requirements for regulated/audited data and applications.
* You can create point-in-time snapshots of EBS volumes, which are persisted to Amazon S3. Snapshots protect data for long-term durability, and they can be used as the starting point for new EBS volumes.

**Question 3: What is CLI and Boto? What are advantages of using CLI? How does Boto function helps in AWS operation?**

CLI

* The AWS Command Line Interface (AWS CLI) is an open source tool that enables you to interact with AWS services using commands in your command-line shell. With minimal configuration, the AWS CLI enables you to start running commands that implement functionality equivalent to that provided by the browser-based AWS Management Console from the command prompt in your terminal program.

Boto

* Boto is the Amazon Web Services (AWS) SDK for Python, which allows Python developers to write software that makes use of Amazon services like S3 and EC2. Boto provides an easy to use, object-oriented API as well as low-level direct service access.

Advantages of using CLI:

* One of the main benefits is the ability to save substantial time. The savings comes in the form of easier installs, support of all services from one tool, moving beyond GUIs, and automating processes and commands with shell scripting.
* **Easier to install -** installation is smooth, quick, simple, and standardized.
* **Supports all Amazon Web Services**
* **Saves time -** Most users find it faster and easier to use the AWS CLI once they reach a certain level of proficiency.
* **Scripting - t**o automate control of all Amazon’s web services with scripts is possibly the biggest benefit. Partial task automation can free developers from needing to log into the AWS Management Console. Shell scripts make it easy to fully automate cloud infrastructure.

How Boto function helps in AWS operation:

* **High-level Service-Specific Features:** Boto3 includes many service-specific features, such as automatic multi-part transfers for Amazon S3 and simplified query conditions for Amazon DynamoDB.
* The Boto library includes modules for every service Amazon offers. Thus, besides EC, it includes the basic services, such as Secure Virtual Network, Autoscaling, DNS, and Load Balancing. For simple web applications with greater scalability and availability requirements, these tools will take you quite a long way. Additionally, Amazon offers a wide range of possibilities for storing data, including the Dynamo distributed database, the Elastic Block Store, a relational database, and a large-scale system for a “data warehouse.”
* The use of language specific libraries that can be easily incorporated and used in simple scripts comes in handy, when a person has to perform an operation that requires a lot of manual effort.
* Python is a kind of programming language that can easily be learnt and used. It is a boon that such an easy language can be used to solve problems of high complexities and cloud automation

**Week 3**

**Question 1: Describe what virtualisation is and describe the characteristic attributes of the different types of virtualisation (Language, Operating System and Hardware).**

Virtualization:

* A machine that’s implemented in software, rather than hardware
* A self-contained environment that acts like a computer
* 4 properties:
  + Partitioning – can run multiple OS on one physical machine and can divide resources between VMS
  + Isolation – provide fault and security isolation at the hardware level and preserve performance with advanced resource controls
  + Encapsulation - Save the entire state of a virtual machine to files and Move and copy virtual machines as easily as moving and copying files.
  + Hardware Independence - Provision or migrate any virtual machine to any physical server.

NOTE: For each of the three types, there is a diagram in the lecture so listen to the lecture and write!!!

Language

OS

* The most basic form of server virtualization is OS-level virtualization. There are limitations when using this method, but it can be implemented and maintained with fewer resources than para- or full- virtualization. When using OS-level virtualization, there is no need for a hypervisor. The duty of managing resources and separating virtual machines is instead handled by the physical server’s operating system. The drawback to this solution is that each virtual machine will have to run the same operating system, because the OS is acting as a hypervisor.

Hardware

* A [Type 1 hypervisor](https://searchservervirtualization.techtarget.com/definition/bare-metal-hypervisor) runs directly on the host machine's physical hardware, and it's referred to as [a bare-metal hypervisor](https://searchservervirtualization.techtarget.com/tip/Compare-the-top-5-bare-metal-hypervisors). The Type 1 hypervisor doesn't have to load an underlying OS. With direct access to the underlying hardware and no other software -- such as OSes and device drivers -- to contend with for virtualization, [Type 1 hypervisors are regarded as the most efficient](https://searchservervirtualization.techtarget.com/tip/Compare-the-top-5-bare-metal-hypervisors) and best-performing hypervisors available for enterprise computing.
* Hypervisors that run directly on physical hardware are also highly secure. Virtualization mitigates the risk of attacks that target security flaws and vulnerabilities in OSes because each guest has its own OS. This ensures an attack on a guest VM is logically isolated to that VM and can't spread to others running on the same hardware.
* The typical Type 1 hypervisor can scale to virtualize workloads across several terabytes of RAM and hundreds of CPU cores.
* In addition, Type 1 hypervisors often provide support for software-defined storage and networking, which creates additional security and portability for virtualized workloads.

**Question 2: Describe what containers are with reference to Docker and discuss their similarities and differences from operating system virtualisation perspective as provided by VirtualBox.**

Docker

* •Containers are cut down VMs used to execute code in a isolated environment
* •All Docker containers use the same underlying OS but present various parts of the OS as if they were dedicated to the running system
* Docker is an open platform for developing, shipping, and running applications. Docker enables you to separate your applications from your infrastructure so you can deliver software quickly.
* Docker provides the ability to package and run an application in a loosely isolated environment called a container. The isolation and security allows you to run many containers simultaneously on a given host. Containers are lightweight and contain everything needed to run the application, so you do not need to rely on what is currently installed on the host. You can easily share containers while you work, and be sure that everyone you share with gets the same container that works in the same way.
* **Fast, consistent delivery of your applications**
* **Responsive deployment and scaling**

Similarities and differences from OS

<https://cloudacademy.com/blog/docker-vs-virtual-machines-differences-you-should-know/>

**Question 3: There was an evolution of operating system during the half century from 1960 to 2010. Identify the virtualisation milestones in this above evolution and explain them briefly.**

* •First VM architected by IBM in 1972 VM/370 to provide full VM of mainframe machine
* •1997 Virtual PC for Mac by Connectix
* •1999 VMware’s VMware Virtual Platform
* •2003 Open Source hypervisor Xen
* •2005 VMware Player – free VM player
* •2007 VirtualBox

Week 4

Week 5