## Lecture 1

#### Virtualization

Multiple virtual machines (VMs) can run inside a physical machine (PM). A user of the VM is given the illusion that they have the full harware to themselves (but in reality many VMs are sharing the same underlying hardware).Containers is another virtualization technology, they are like lightweight VMs.

Virtualization is a building block for cloud computing. Virtualization enables multiple clients share the cloud’s compute resources. Multiple users on VMs/containers can share the same cloud server. Without virtualization, the cloud wouldn’t be a profitable idea.

#### Why cloud computing?

Bare-metal refers to a computer system or server that is used without any virtualization layer pre-installed. In other words, the term "bare-metal" implies that the hardware is directly accessible and utilized without the abstraction layers that virtual machines or containerization technologies provide.

Why run applications on cloud and not on bare metal servers?

* Multiplexing gains: multiple VMs can share the system resources. It stems from the fact that the machines are not utilized at 100% capacity by each user; there’s some spare capacity that can be shared.
* Lower overhead of maintenance: hardware/software maintained by providers (if using public cloud).
* Flexibility: VMs can move to another physical machine if one fails.
* Pay as per usage: no need to invest in servers if only lightly used.

Disadvantages of running applications on cloud:

* Performance: longer delay to access server via internet.
* Higher cost if heavily used

#### Virtualization terminology

System virtualization: running one full system (OS and applications) over another OS.

Process virtualization: gives a single process the illusion that it is running on a different architecture; it lets a single process run on a different architecture from the underlying machine.

Hypervisor or Virtual Machine Monitor (VMM): a piece of software that allows multiple VMs to run on a PM.

A group of blue and orange rectangular boxes with text

Description automatically generated with medium confidenceThe hypervisor allows the VMs to share the underlying PM.

The Guest OS runs inside the VM, and the Host OS runs on the PM.

**Type 1 Hypervisor**: runs directly on hardware (no need for host OS); it takes on the role of the host OS as well. It is VMM plus an OS built-in together.

**A close-up of several types of computer hardware

Description automatically generatedType 2 Hypervisor**: runs as an application on top of the host OS; it does not have an OS functionality.

#### Challenges to virtualization

A group of blue and orange rectangular boxes with text

Description automatically generatedThe Guest OS expects complete control over hardware, but the VMM must multiplex multiple guests on the same hardware.

Some ways to design VMMs:

* Hardware assisted virtualization (e.g. KVM/QEMU): modern CPUs have support for virtualization and VMMs are built over this support
* Full virtualization (e.g. VMWare): original technique to run unmodified OS (as Guest OS) over original hardware with no virtualization support (the hardware provides no support for virtualization).
* Paravirtualization (e.g. Xen): OS source code is modified to be compatible with virtualization.

When the underlying hardware does not support virtualization, we have to use the full virtualization technique in order to run OSs as Guest Oss.

The virtualization applies to all hardware resources: CPU, memory, I/O devices, etc (they all need to be shared). All of them need to be virtualized.

## Lecture 2