

Cloud Computing 101

- **on demand self services** You choose your services with no human interactions
- **Broad network access** Availability over the network
- **Resource Pooling** aggregation of HW as one (Datacenter)
- **Rapid elasticity** fast allocation/deallocation of res.
 - vertical: increase the HW
 - horizontal: redundant VMs load balanced

Stuff as a service

- **SAAS** remote access to sv (API)
- **IASS** complete control over low level res.
- **PASS** platform for developer, not caring of res.

Deployment models

- **Public Cloud**
- **Private Cloud**
- **Hybrid Cloud**

Virtualization

Running multiple OSs on same host by an **Hypervisor** (like Vbox), a mediator between OSs and physical resr.

Also HW can be virtualized, RAID: (a redundant array of inexpensive disk), take some multiple physical disks and use them as one virtualized disk.

HyperV

- **Bare-Metal**: HyperV sits on the HW, boot as an OS
- **Host-Based**: HyperV is a SW

features

- **equivalence** → exposed Vms have same res. as the local machine.
- **safety** → isolation
- **performance** → native code is sent to processor so is fast

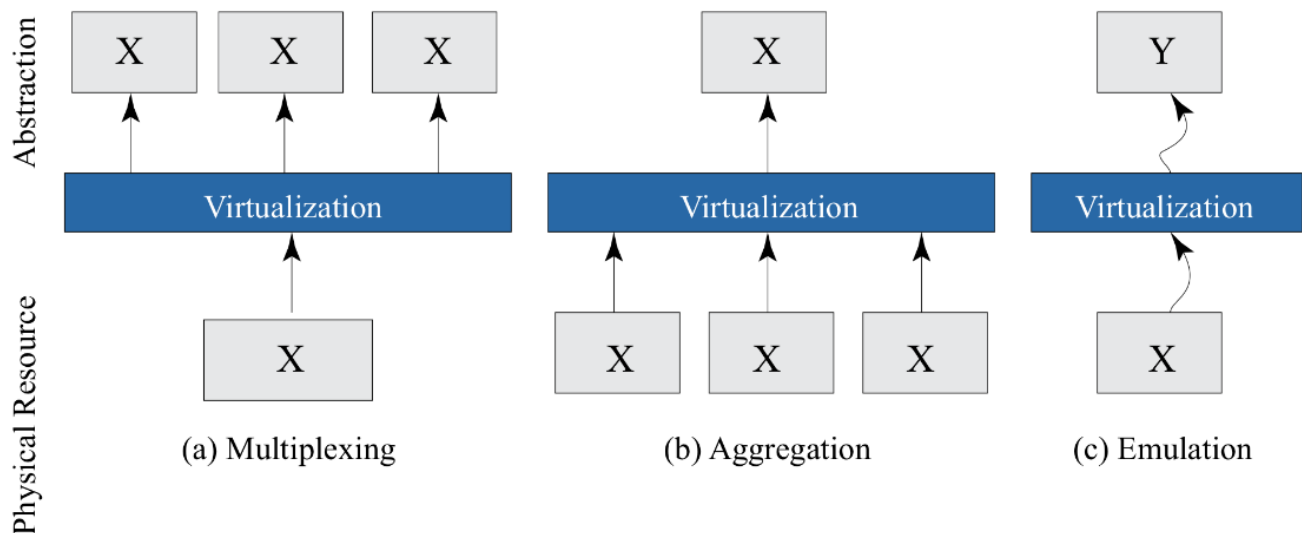
Kind of virtualization

- **Full-virtualization**: virtualization of res.
- **Para-virtualization**: part of kernel patched to interact with HyperV
- **HW assisted-virtualization**: more efficient full virtualization

Container based VMs

- Instead of an HyperV this approach share the Kernel among VMs, while the HyperV virtualized also kernels for each one.
- A container is a group of processes on a Linux host, grouped together in an isolated environment
- **Namespaces** → to assing interfaces, sockets, routing...
- **Cgroups** → assign res. to each procs (CPU, shared mem, ...)
 - Control groups (cgroups) are a kernel mechanism for grouping, tracking, and limiting the resource usage of processes.
 - Hierarchy: a set of subsystems mounted together forms a hierarchy.
 - Tasks: processes are called tasks (in cgroups terminology).
 - Cgroups : A cgroup associates a set of tasks with a set of parameters for one or more subsystems

Virtualization techniques



- **Multiplexing** use one sistem to virtualizd multiple
- **Aggregation** (see RAID)
 - Emulation** idk

Virtualize CPU, RAM, I/O

- CPU and RAM virtualization is achieved via temporal and spatial multiplexing
 - RAM → spatial multiplexing
 - CPU → temporal multiplexing
- I/O virtualization is done via emulation

Vagrant

You can use Vagrant instead on an Hypervisor for VMs management, because it use images.

SDN (Software Defined Networking)

- represent the OpenFlow project
- data plane remotely configured by the control plane
- forwarding decision flow-based instead of destination-based