

# Administración de Servidores

## Virtualización Introducción

Creado por

Iñaki Fernández de Viana y González

Huelva, enero 2021

# Sobre Nosotros

## Iñaki Fernández de Viana y González



Despacho 128. Escuela Técnica  
Superior de Ingeniería



Dpto. De Tecnologías de la Información



[i.fviana@dti.uhu.es](mailto:i.fviana@dti.uhu.es)



+34 959217378



# Índice

1. Introducción
2. Tipos de virtualización
3. Usando hipervisores
4. Proyectos OpenSource
5. Virtualización y la Nube

# Introducción

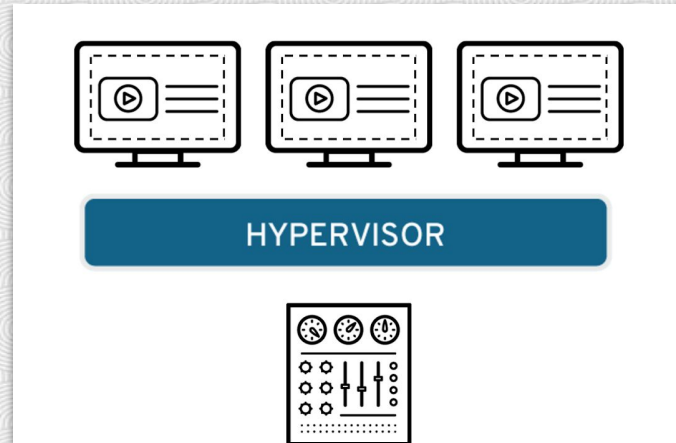
---

1

---

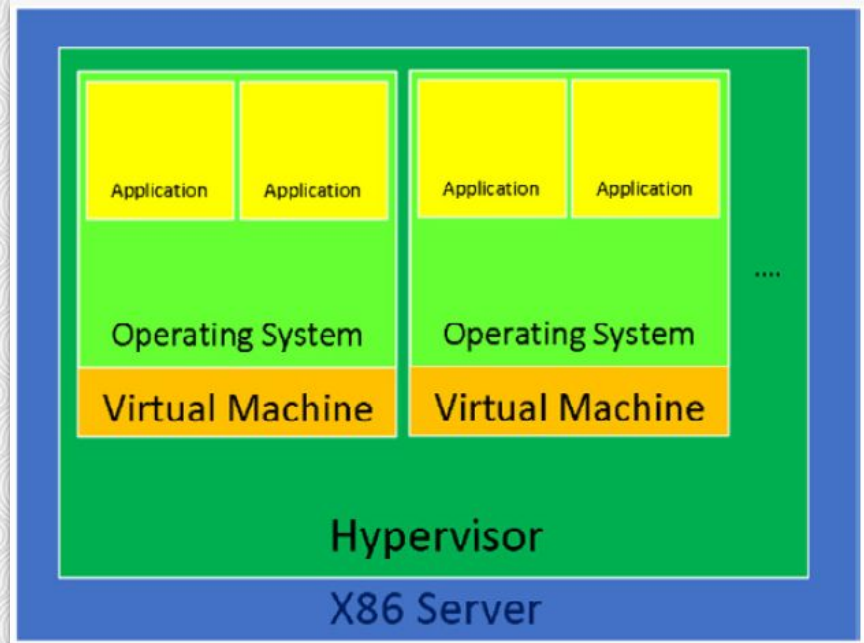
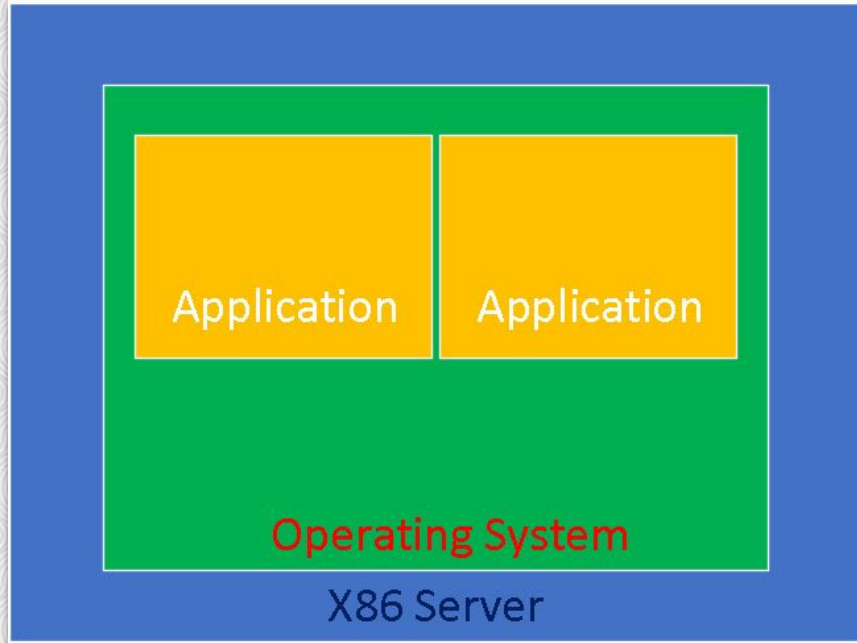
## ¿Qué es?

- ★ La virtualización es la creación de recursos virtuales y mapeo a recursos físicos
- ★ Se puede hacer usando funcionalidades *hardware* (controladoras de particionado) o *software* (**hypervisor**)





# Virtualización vs Entornos físicos



## ¿Por qué virtualizar?

- ★ Consolidación de servidores y optimización de infraestructuras.
- ★ Reducción de costes de infraestructura física.
- ★ Flexibilidad operativa mejorada y capacidad de respuesta.
- ★ Mayor disponibilidad de aplicaciones y continuidad del negocio mejorada.
- ★ Capacidad de gestión y seguridad mejorada

# Soluciones

- ★ Actualmente existe gran cantidad de soluciones de virtualización
  - Red Hat con **KVM**
  - Microsoft con **Hyper-V**
  - Vmware con **ESXi**
  - Oracle con Oracle VM (basado en **Xen**)
  - Citrix con XenServer
- ★ Todo esto ha llevado a desarrollo de distintas soluciones en la nube como **EC2**, **AWS**, **Office 365**, **Azure**, **vCloud Director** y **vRealize Automation**.



# Tipos de virtualización

---

## 2

---

# Introducción

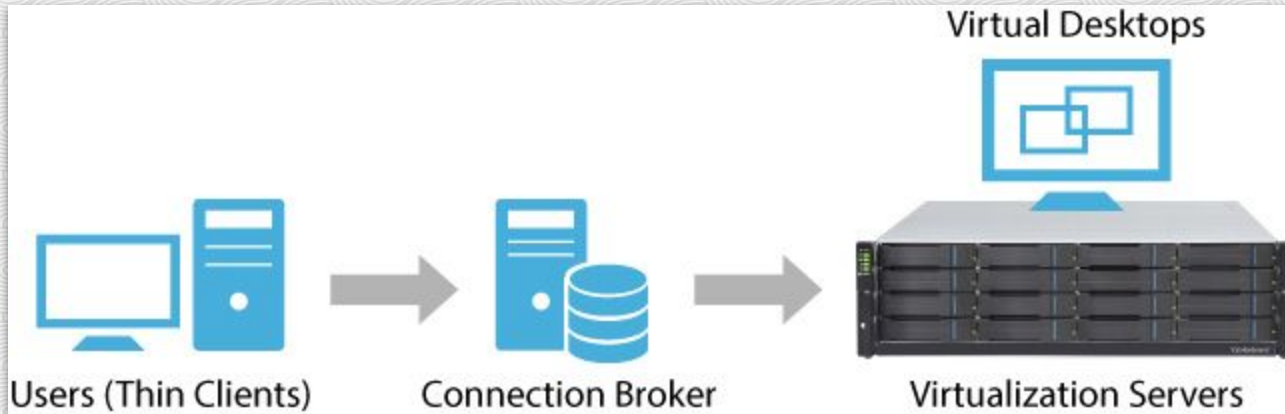
- ★ Existen distintas formas de categorizar las soluciones de virtualización existentes.
- ★ Quizás los criterios más extendidos hacen referencia al **qué** y al **cómo**.

# Qué Virtualizar

- ★ Atendiendo a lo **qué** queremos virtualizar:
  - Virtualización de escritorio (Virtual Desktop Infrastructure - VDI)
  - Virtualización de Servidor
  - Virtualización de Aplicación
  - Virtualización de Red
  - Virtualización de Almacenamiento

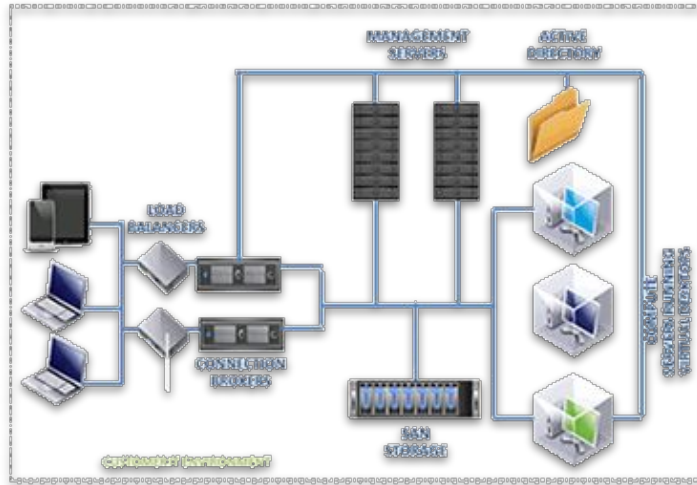
## Desktop Virtualization (VDI)

- ★ Es usado en muchas compañías porque los usuarios no requieren de un dispositivo específico para acceder a su escritorio de trabajo, ya que este está virtualizado.



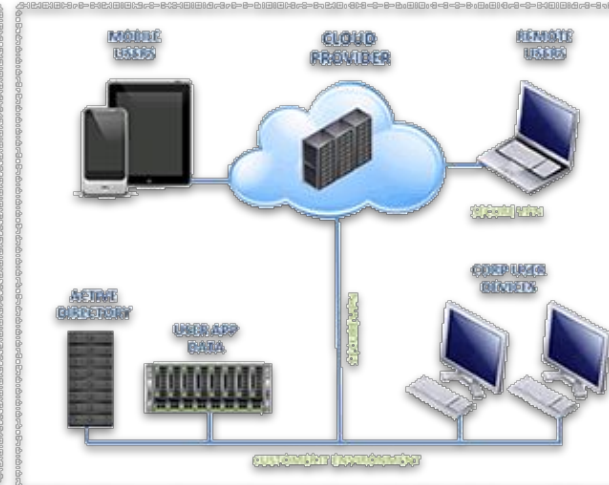
# Desktop Virtualization (VDI)

## Modelo de VDI tradicional



- Modelo basado en CAPEX
- Riesgos de implantación
- Necesario expertise en la plataforma
- Poca flexibilidad

## Modelo DaaS (cloud hosted)



- Modelo basado en OPEX
- Escalabilidad bajo demanda
- No necesario expertise en la plataforma
- Seguridad de Datacenter



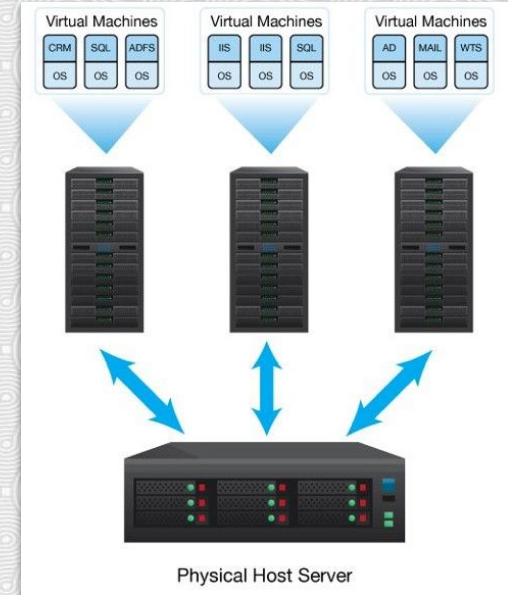
# Desktop Virtualization (VDI)

## ★ Beneficios

- Gestión y almacenamiento centralizado.
- Actualizaciones más sencillas (sólo es necesario actualizar la imagen base).
- Proceso de despliegue más sencillo (no es necesario ir ordenador por ordenador y podemos centralizar la gestión de aplicaciones)
- Más fácil de gestionar el cumplimiento de las normativas de seguridad.

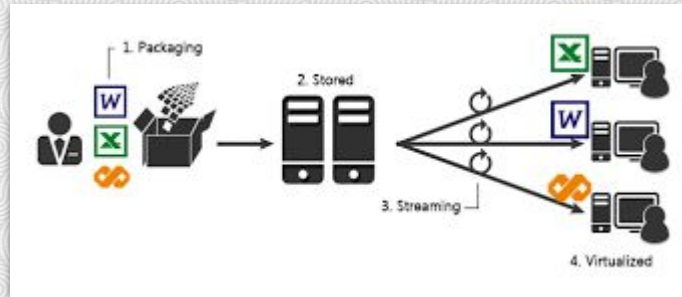
# Virtualización de Servidores

- ★ Usada por la mayoría de las empresas de IT.
- ★ Trabajamos con servidores físicos en vez de físicos
  - Facilidad backups
  - Eficiencia energética
  - Facilidad en modificación de workloads



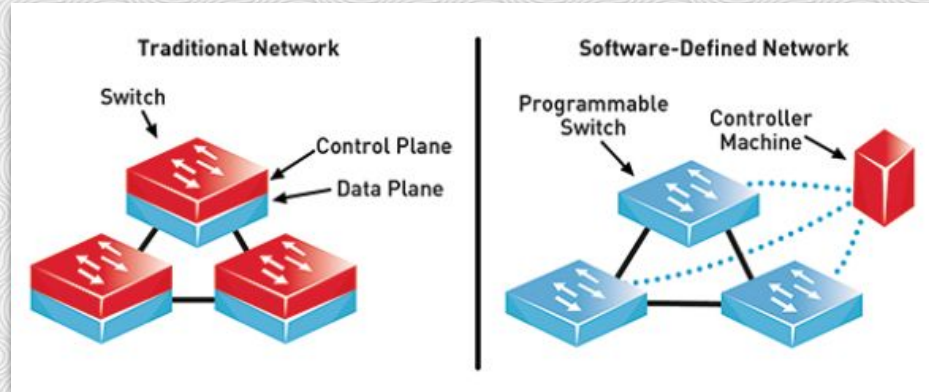
# Virtualización de aplicaciones

- ★ Se implementan aplicando
  - Algún tipo de protocolo de streaming o control remoto (**Microsoft App-V**)
  - Soluciones que permitan empaquetar las aplicaciones en volúmenes que se montan en máquinas virtuales (**VMware App Volumes**)



## Virtualización de redes

- ★ También denominada *SDN (Software-Defined Networking)*
- ★ Permite crear dispositivos de red virtuales (*switches*)
- ★ SDN es una extensión de la idea de virtualización de redes, toda la gestión y configuración de la se realiza mediante software.
- ★ Facilita la gestión de redes complejas



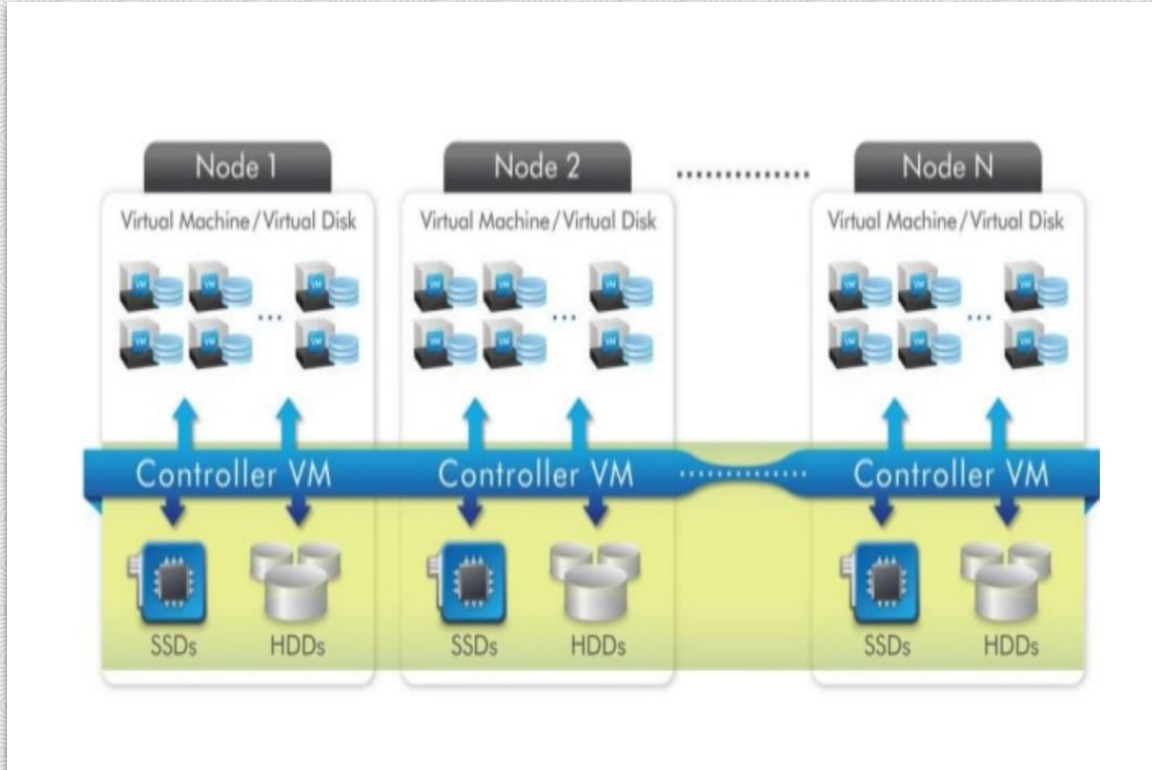


## Virtualización de Almacenamiento

- ★ También denominada *SDs (Software-Defined Storage)*
- ★ Crean dispositivos de almacenamiento virtual basados en un conjunto de dispositivos físicos que podemos gestionar de forma centralizada como si fuera un único dispositivo de almacenamiento
- ★ Se crea una capa de abstracción que aísla al recurso de los detalles internos de los dispositivos de almacenamiento
- ★ SDS desacopla el software de almacenamiento del hardware en el que se está ejecutando, se crea una capa de abstracción
- ★ Esta capa controla y gestiona el hardware.



# Virtualización de Almacenamiento

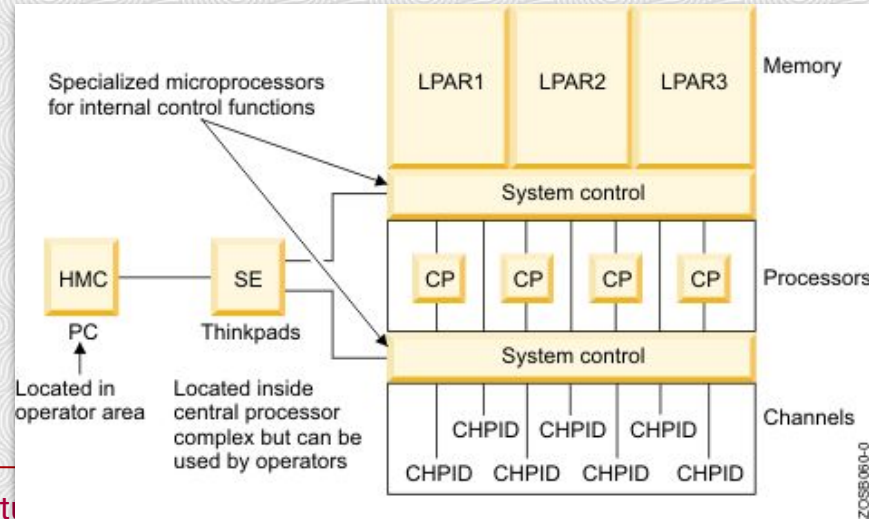


# Introducción

- ★ Atendiendo a **cómo** se realiza la virtualización
  - Partitioning
  - Full virtualization
  - Paravirtualization
  - Hybrid virtualization
  - Container-based virtualization

# Partitioning

- ★ La CPU se divide en diferentes partes y cada parte trabaja como un sistema independiente.
- ★ Estas particiones tienen su propio sistema operativo (for example, **IBM Logical Partitions (LPARs)**).



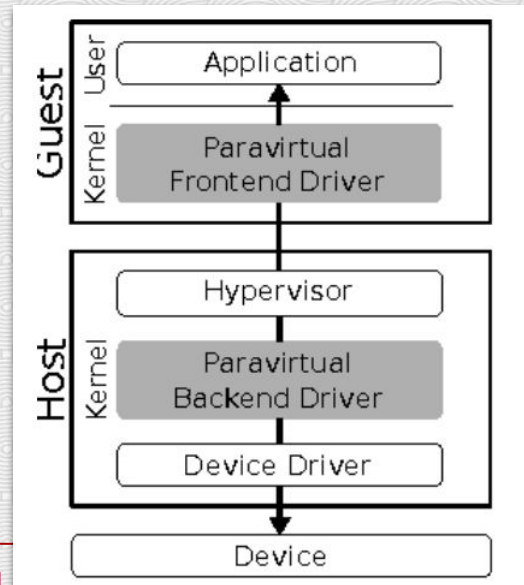
## Full virtualización

- ★ Una máquina virtual se usa para simular un hardware y el *guest OS* (SO instalado en la Máquina Virtual) no es consciente de este cambio.
- ★ Se puede hacer
  - **Por software**: se usa traducción binaria para virtualizar la ejecución de conjuntos de instrucciones sensibles, mientras que se usa software para emular hardware (sobrecargas)
  - **Por hardware**: no hay esa traducción, se interacciona directamente con características de virtualización de la propia CPU (**AMD-V**, **Intel VT**). Las instrucciones las ejecuta directamente la CPU.



## Paravirtualización

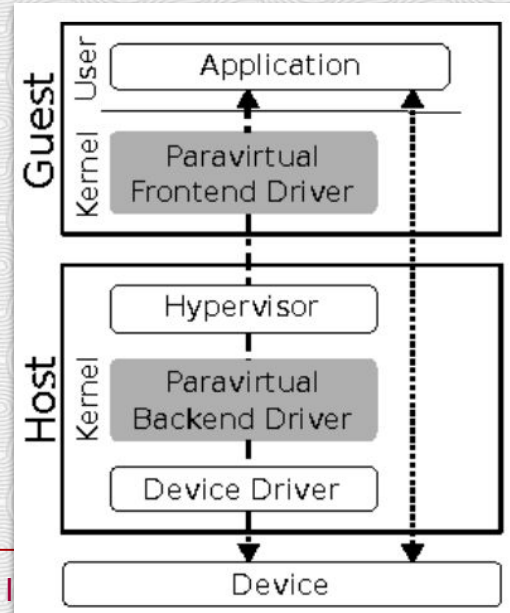
- ★ El sistema operativo invitado conoce que está siendo virtualizado y tiene que ser modificado.
- ★ No necesita activar las extensiones de virtualización de las CPU
- ★ Xen





## Hybrid virtualization

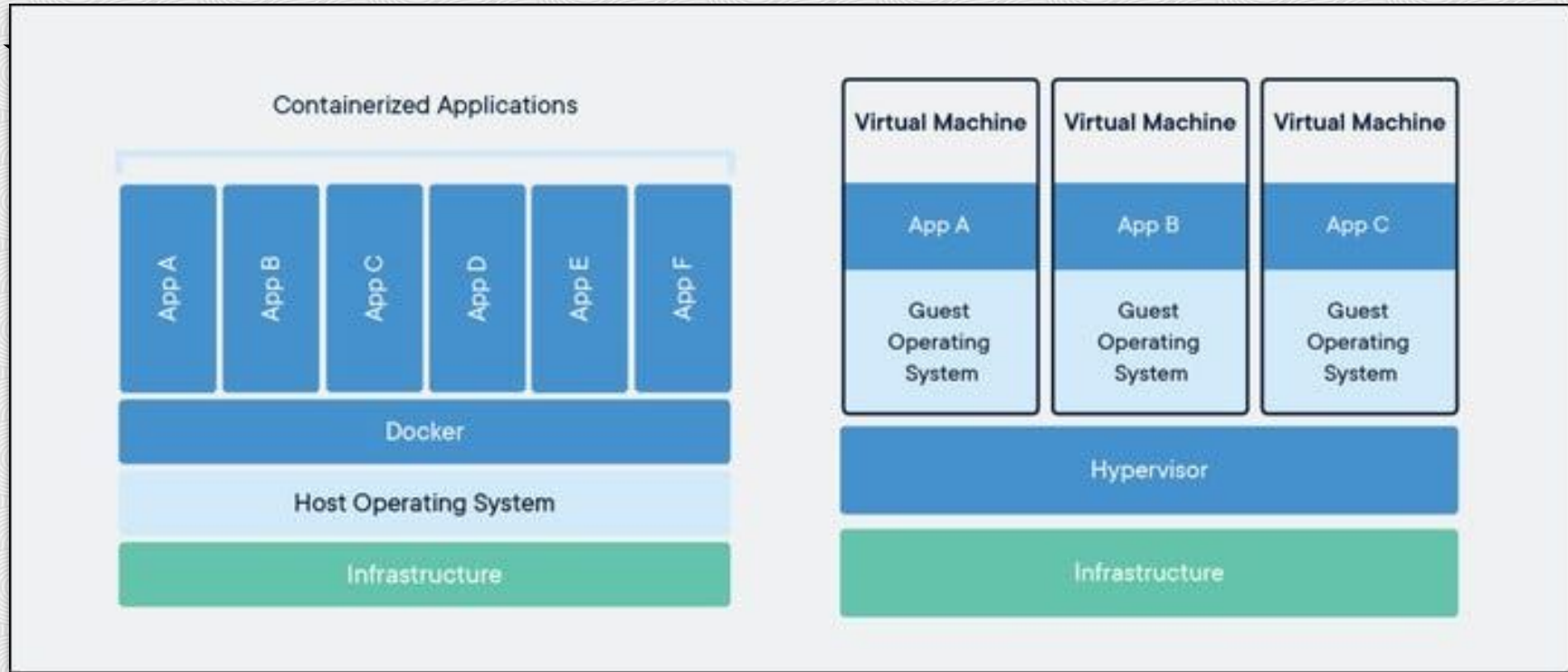
- ★ Combina la paravirtualización con la full virtualization
- ★ En concreto, permite añadir drivers de paravirtualización al Guest OS para ciertos aspectos específicos (I/O intensivo)
- ★ Xen, ESXi



## Container-based virtualization

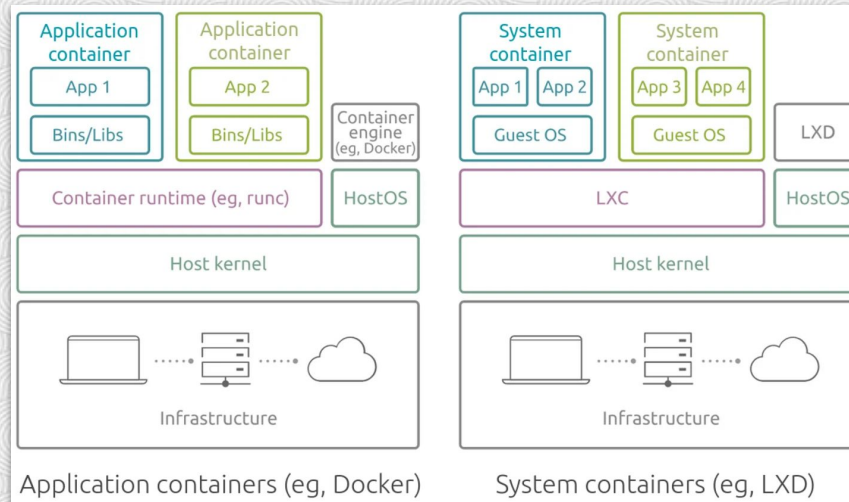
- ★ Es un tipo de virtualización que usa contenedores.
- ★ Un contener es un objeto que empaqueta una aplicación/sistema y todas sus dependencias.
- ★ Las aplicaciones son fácilmente escalables y desplegadas sin necesidad de una máquina virtual o un hypervisor.
- ★ Soluciones como **Docker** o **Podman** permiten crear contenedores para aplicaciones y **Lxd** para sistemas.

# Container-based virtualization



# Container-based virtualization

- ★ Application containers package a single process or application.
- ★ System containers simulate a full operating system and let you run multiple processes at the same time.



# Usando hipervisores

---

3

---



# Introducción

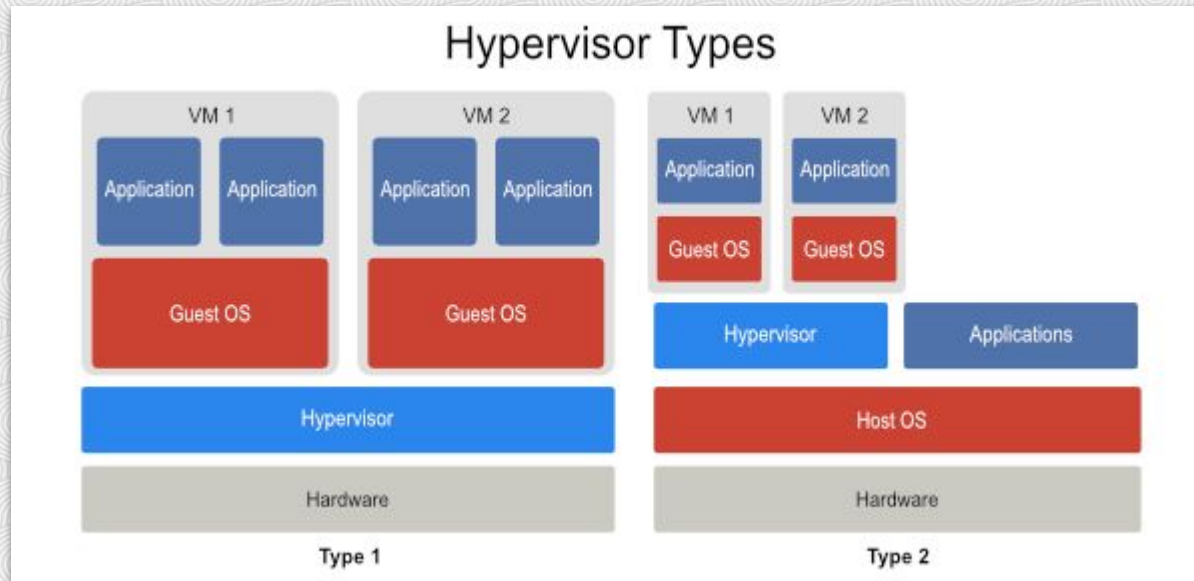
- ★ El hipervisor o *Virtual Machine Manager* (VMM) es el software responsable de monitorizar y controlar las máquinas virtuales.
- ★ Entre otras tareas se encarga:
  - Proporcionar hardware virtual y gestión eficiente
  - Gestionar el ciclo de vida de las máquinas virtuales
  - Migración de máquinas virtuales
  - Asignación de recursos en tiempo real
  - Definición de políticas para la gestión de máquinas virtuales.

# Introducción

- ★ Permiten ejecutar múltiples (y distintos) sistemas operativos sobre el mismo hardware físico.
- ★ Estos sistemas operativos solicitan recursos que el hipervisor debe gestionar atendiendo a la configuración de la máquina virtual.

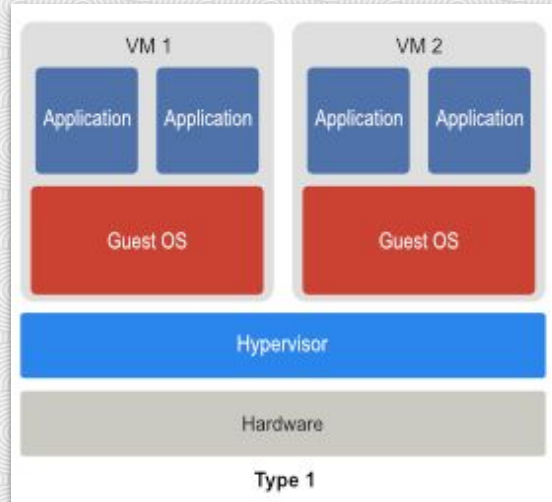
# Tipos

- ★ Fundamentalmente se dividen en tipo 1 (bare-metal, embedded, or native hypervisors) y tipo 2.



# Tipo 1

- ★ Si el hipervisor se ejecuta directamente sobre el hardware, sin necesidad de un SO (oVirt-node, VMware ESXi/vSphere, y Red Hat Enterprise Virtualization Hypervisor (RHEV-H))



## Ventajas/Inconvenientes

### ★ Ventajas:

- Fáciles de instalar y configurar
- Pequeño tamaño; optimizados para Small in size; optimizados para dar la mayor parte del hardware a la VM
- Menos overhead, sólo se ejecutan las aplicaciones necesarias
- Más seguros, las VM son independientes

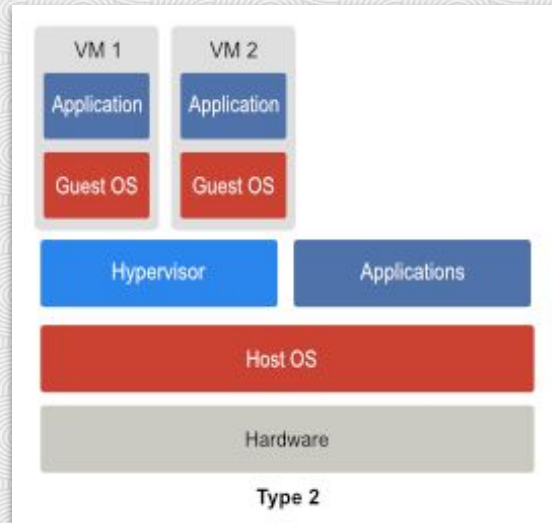
### ★ Desventajas:

- No aceptan personalizaciones: fuertes restricciones cuando necesitamos instalar aplicaciones o drivers de terceros



## Tipo 2

- ★ Si existe un sistema operativo y el hipervisor actúa en ese nivel entonces hablaremos de hipervisor de tipo 2 (**VMware Player**, **VMware Workstation**, **VMware Fusion** y **Oracle VirtualBox**).



## Ventajas/inconvenientes

- ★ Ventajas:
  - Gran soporte hardware (lo proporciona el OS host)
- ★ Inconvenientes:
  - Necesitan de cierto grado de optimización

# Proyectos OpenSource

---

3

---

# Introducción

## ★ Existen distintos proyectos de virtualización Open Source

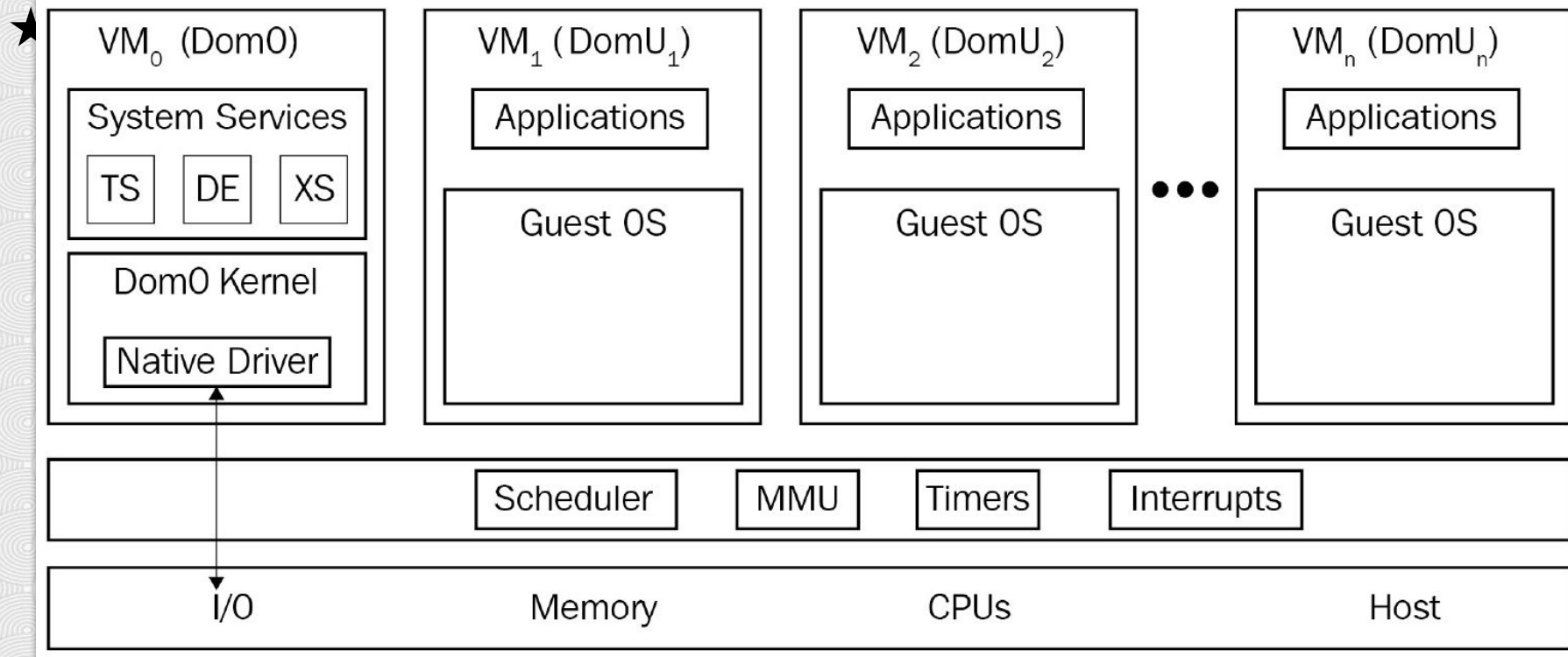
Proyecto	Tipo de virtualización
KVM	Full-virtualization
VirtualBox	Full-virtualization
Xen	Full and paravirtualization
Lguest	Paravirtualization
UML (User Mode Linux)	First

# Xen

- ★ Desarrollado originalmente por la universidad de Cambridge, tuvo su primera versión en 2003.
- ★ Está portado a distintas arquitecturas.
- ★ Xen se construye en torno al:
  - Xen hypervisor: controla la comunicación entre VM y el hardware
  - Dom0: controla el entorno de una VM (QEMU)
  - Management utilities utilidades y GUI para gestionar XEN.
  - Virtual Machines



# Xen

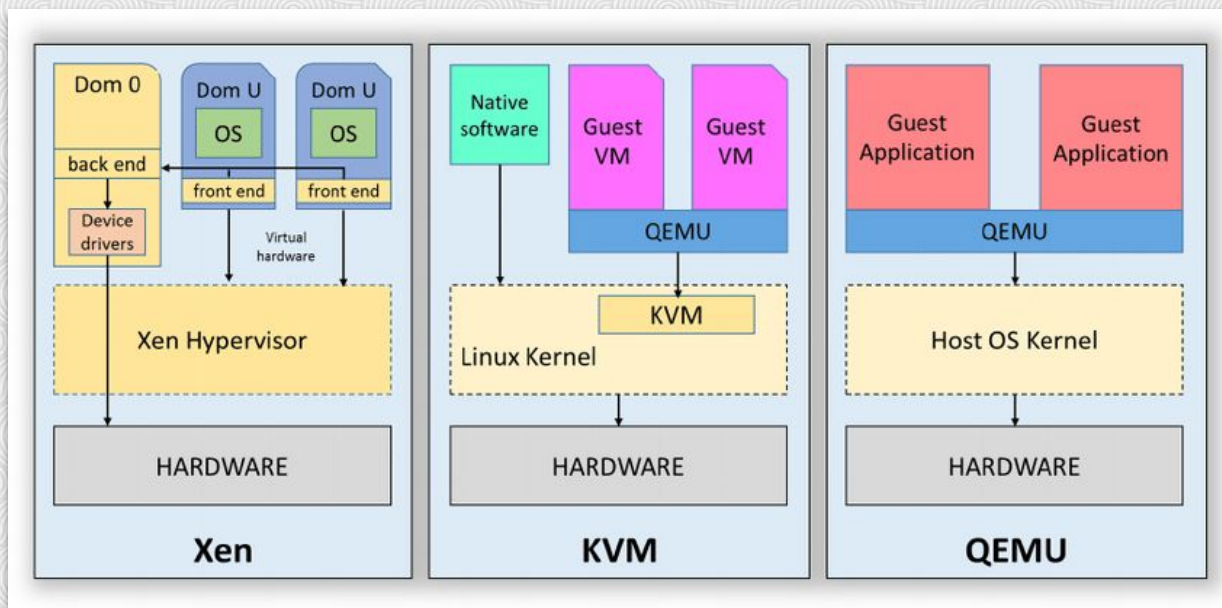


# KVM

- ★ Representa la última generación de software de virtualización
- ★ Su objetivo es crear un hipervisor moderno soportado en hardware de última generación (VT-x, AMD-V, and so on).
- ★ KVM convierte el núcleo de Linux en un hipervisor cuando instalamos el módulo para KVM
- ★ Como módulo, se beneficia de todas las mejoras que se han ido introduciendo el Linux a lo largo de los años
- ★ Para emular las entradas/salidas, KVM usa Qemu (programa en el espacio de usuario que emula hardware)

# KVM

- ★ Qemu emula al procesador ya una serie de periféricos (discos, tarjetas de red o video, PCI, USB...)



# La virtualización en la Nube

---

4

---

## Proveedores IaaS

- ★ La nube es la palabra de moda que está formando parte de todas las discusiones IT de los últimos años
- ★ Actualmente los principales proveedores de **IaaS** (**Infrastructure-as-a-Service**) son:
  - En 2006, Amazon liberó **Amazon Web Services (AWS)** y **Amazon Compute Cloud (EC2)**
  - En 2008, **Google Cloud Platform**
  - En 2010, **Microsoft Azure**
- ★ Otros proveedores: **IBM Cloud**, **VMware Cloud** on AWS, **Oracle Cloud**, **Alibaba Cloud** ...



## Proveedores IaaS

- ★ Muchos de estos proveedores usan soluciones OpenSource
  - Amazon usa Xen y KVM
  - Google cloud usa KVM
- ★ Actualmente, los principales proyectos libres que usan virtualización para crear infraestructuras IaaS son:
  - OpenStack (KVM)
  - CloudStack (Xen)
  - Eucalyptus (KVM y Xen)

## Ventajas e Inconvenientes

- ★ Se confía la infraestructura sobre la que reposa la virtualización a un tercero y se tienen que usar los medios asociados a este como: imágenes, sistemas operativos o aplicaciones específicas.
- ★ Las tareas administrativas no son exactamente las mismas.
- ★ La mayoría de los costes se desplazan al proveedor y están incluidos en las tarifas. Se calculan respecto al uso de los recursos: solamente se paga lo que se consume (CPU, memoria, disco, tráfico de red, etc.).