



GOBIERNO DE LA CIUDAD DE BUENOS AIRES

Ministerio de Educación

Dirección de Formación Técnico Superior

Instituto de Formación Técnico Superior N° 18

Mansilla 3643, (C1425BBW) Capital Federal

Arquitectura de sistemas en la Nube

Infraestructura y Virtualización

- *Hardware, Abstracción y Gestión del Ciclo de Vida*

- **Hardware Físico**
- Comprensión de los componentes del datacenter y tipos de almacenamiento
- **Virtualización**
- Diferencias clave entre Hypervisors, VMs y Contenedores.
- **Administración**
- Gestión automatizada del ciclo de vida de los recursos virtuales.



Componentes Principales del Datacenter

- Un Datacenter es un ecosistema diseñado para la alta disponibilidad.
- Sus componentes se agrupan en tres áreas críticas:
 - A. Infraestructura IT (Cómputo y Red)
 - B. Infraestructura de Soporte (Facilities)
 - C. Seguridad y Gestión (Security & Management)



Infraestructura IT (Cómputo y Red)

- **Racks:** Estructuras metálicas estandarizadas donde se montan los servidores.
- **Servidores:** Máquinas potentes que procesan datos. En entornos cloud
- **Networking:** Switches de alta velocidad, routers y firewalls que conectan los servidores entre sí y con el mundo exterior.



Infraestructura de Soporte (Facilities)

- **Energía:** Sistemas de alimentación ininterrumpida (UPS) y generadores diésel para evitar caídas ante cortes eléctricos.
- **Refrigeración:** Sistemas de aire acondicionado de precisión (HVAC) o refrigeración líquida para mantener el hardware a temperaturas operativas (normalmente entre 18°C y 27°C).



Seguridad Física y Lógica

- **Seguridad Perimetral:** No solo cámaras; incluye controles de acceso biométricos (huella, iris), esclusas de seguridad (mantraps) y personal de vigilancia 24/7.
- **Ciberseguridad:** Firewalls físicos, sistemas de detección de intrusos (IDS) y segmentación de redes para que un fallo en un sector no comprometa a todo el datacenter.



Monitoreo y DCIM (Data Center Infrastructure Management)

- **Gestión de Activos:** Un software especializado (DCIM) que permite ver en tiempo real el consumo eléctrico de cada rack, la temperatura exacta en cada pasillo y el estado de salud de cada servidor.
- **Redundancia N+1:** Esta área se encarga de asegurar que siempre haya un respaldo activo. Si un componente falla, el sistema de gestión debe conmutar automáticamente al respaldo sin que el usuario lo note



Conectividad y Carrier Neutrality

- **Meet-Me Rooms (MMR):** Espacios seguros donde diferentes proveedores de internet (ISPs) se conectan entre sí. Un datacenter crítico debe ser "Carrier Neutral", permitiendo que los datos entren y salgan por múltiples rutas de fibra óptica distintas para evitar quedar aislados si un proveedor tiene un problema.

Clasificación de Centros de Datos

- La clasificación más aceptada y utilizada a nivel mundial es el sistema de **Tiers** (niveles) creado por el **Uptime Institute**. Este sistema mide principalmente el **tiempo de disponibilidad (uptime)** y la **redundancia** (respaldos).



Nivel	Nombre	Disponibilidad	Características principales
Tier I	Básico	99.671%	Sin redundancia. Un solo camino de energía y refrigeración. Si algo falla o se requiere mantenimiento, el servicio se detiene.
Tier II	Componentes Redundantes	99.741%	Incluye algunos componentes de respaldo (como generadores o UPS), pero sigue teniendo un solo camino de distribución.
Tier III	Mantenimiento Concurrente	99.982%	Es el estándar de la industria. Tiene múltiples caminos de energía y enfriamiento. Se puede hacer mantenimiento sin apagar los servidores.
Tier IV	Tolerante a Fallas	99.995%	Máxima seguridad. Todo está duplicado. Si ocurre un error crítico, el sistema sigue funcionando sin interrupciones.

Clasificación por Modelo de Propiedad y Uso

- Además de los niveles técnicos, los centros de datos se dividen según quién los opera y para qué:
- **Enterprise Data Centers:** Construidos y operados por una sola empresa para su propio uso exclusivo (ej. un banco grande).
- **Colocation (Multi-tenant):** Una empresa alquila el espacio, la energía y la refrigeración en un edificio, pero lleva sus propios servidores.
- **Cloud Data Centers:** Propiedad de gigantes como Amazon (AWS), Google o Microsoft. Ofrecen servicios virtuales donde tú solo alquilas potencia de cómputo.
- **Edge Data Centers:** Centros más pequeños ubicados cerca de los usuarios finales para reducir la latencia (ideales para streaming o videojuegos).

Clasificación por Tamaño (Escala)

Micro Data Centers: Del tamaño de un refrigerador o rack, usados para oficinas pequeñas.

Hyperscale Data Centers: Instalaciones masivas con miles de servidores, diseñadas para escalar horizontalmente de forma rápida.

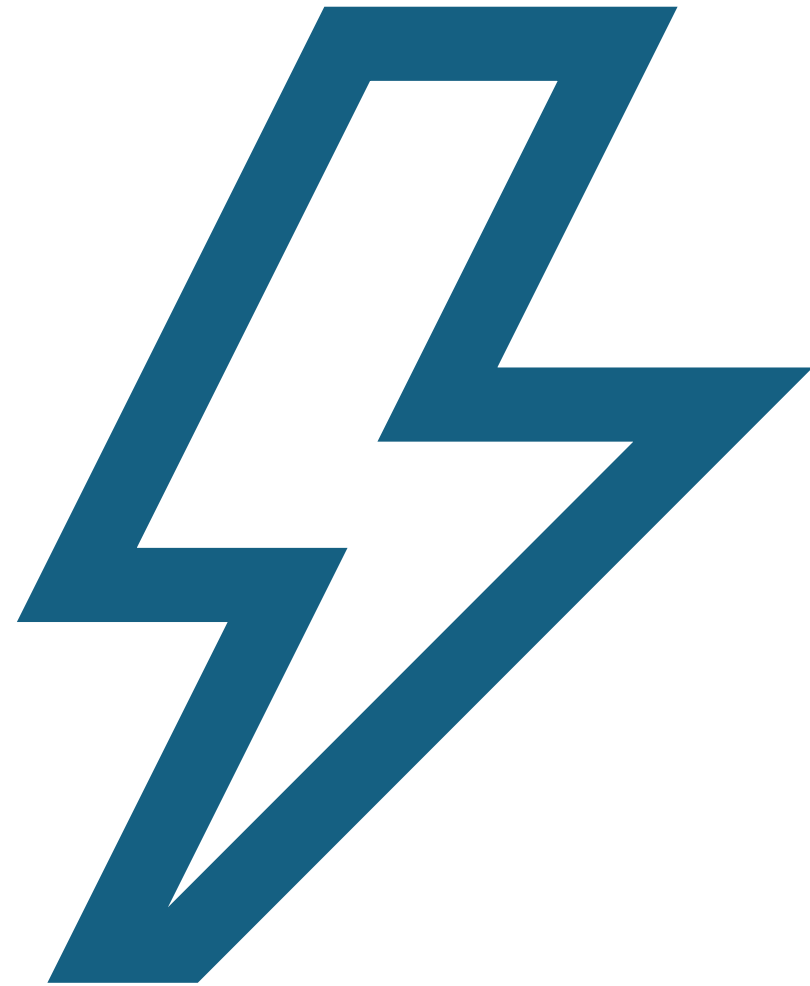
Comparación de Redundancia

En ingeniería de centros de datos, usamos la letra "**N**" para representar la capacidad básica necesaria para funcionar

Característica	Tier III	Tier IV
Arquitectura	N+1	2N o 2N+1
Caminos de energía	Múltiples (solo 1 activo)	Múltiples (todos activos)
Respuesta ante fallas	Requiere intervención o es reactivo	Respuesta automática (pasiva)
Tiempo de inactividad anual	Máximo 1.6 horas	Máximo 26 minutos
Protección contra incendios	Estándar	Sistemas autónomos por sectores

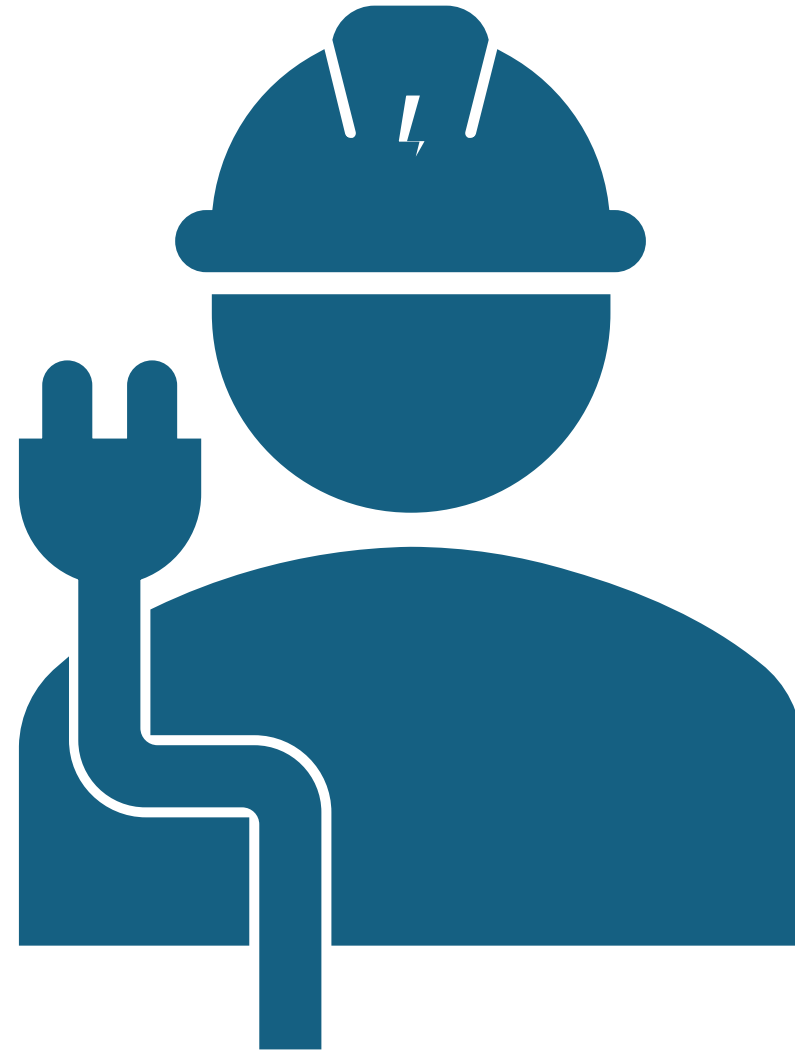
El Sistema UPS (Energía Ininterrumpida)

- La UPS es el puente crítico entre la caída de la red eléctrica externa y el encendido de los generadores (que suelen tardar de 10 a 30 segundos en estabilizarse).



Arquitecturas de Redundancia:

- **Tier III (N+1):** Si tu carga requiere 3 módulos de UPS para funcionar (N), instalas 4. Esto permite que un módulo falle o entre en mantenimiento sin afectar la operación. Sin embargo, todos suelen estar conectados al mismo bus de salida.
- **Tier IV (2N o 2 veces (N+1)):** Tienes dos sistemas de UPS completamente independientes en salas distintas. El servidor (que debe tener doble fuente de poder) recibe energía de la "UPS A" y de la "UPS B". Si explota toda la sala de la UPS A, la B sigue alimentando el equipo sin parpadear.



Tipos de almacenamiento de energía:

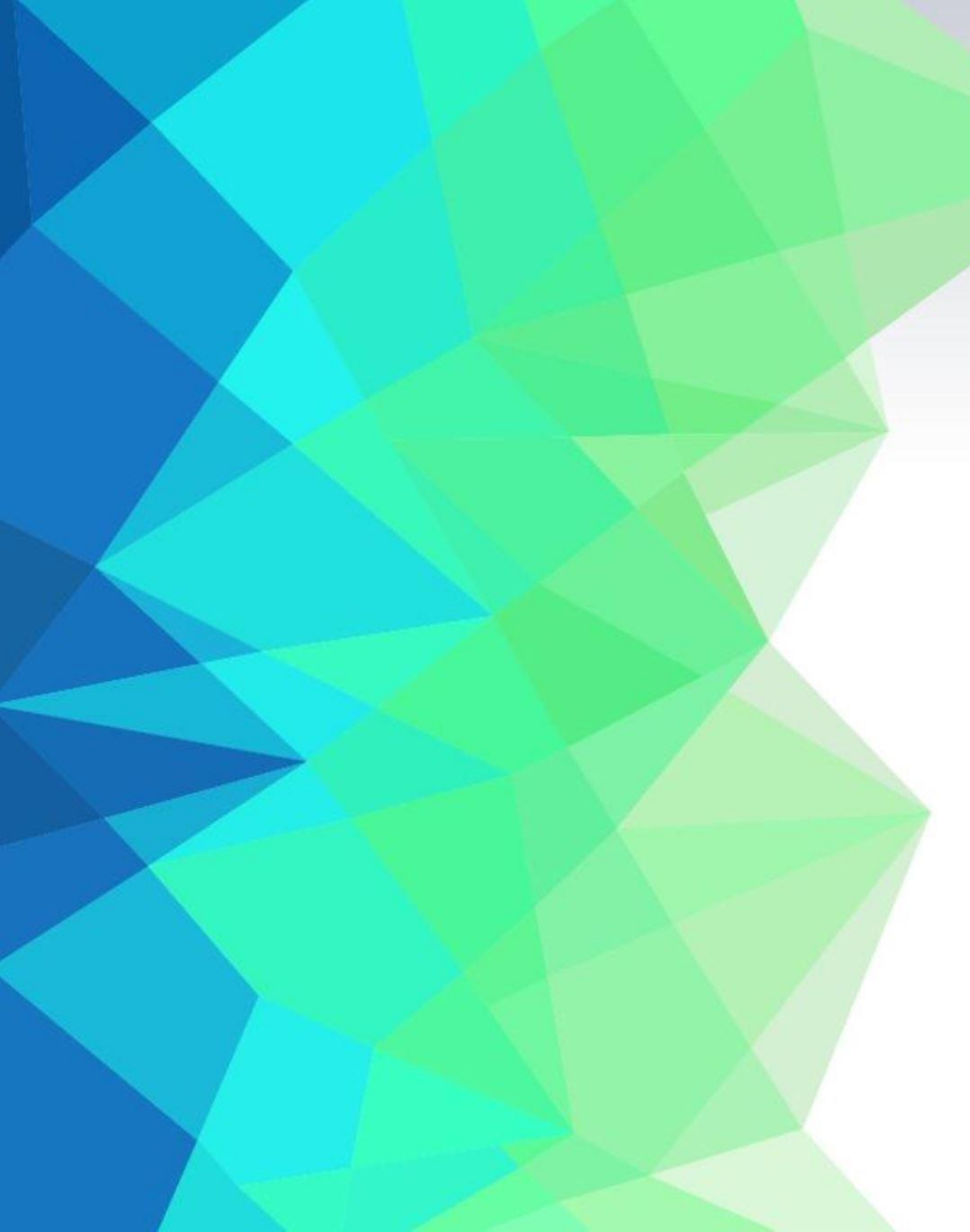
- **Baterías VRLA (Plomo-ácido):** Las tradicionales, pesadas y requieren mucho mantenimiento.
- **Baterías de Litio:** Más compactas, duran más años y cargan rápido (la tendencia actual).
- **Flywheels (Volantes de inercia):** Almacenan energía cinética en un disco que gira a miles de RPM. No usan químicos, pero solo dan energía por unos pocos segundos.



El Sistema de Refrigeración (HVAC/CRAH)

- **Estrategias de Gestión de Aire:**
 - Para evitar que el aire frío se mezcle con el caliente (lo cual es muy ineficiente), se usan los **pasillos contenidos**:
 - **Pasillo Frío (Cold Aisle):** Las caras de los racks se miran entre sí y el aire frío sube desde el suelo técnico.
 - **Pasillo Caliente (Hot Aisle):** Las partes traseras de los racks expulsan el calor hacia un retorno en el techo.
- **Métodos de enfriamiento:**
 - **CRAC (Computer Room Air Conditioner):** Un aire acondicionado de precisión que usa refrigerante directo.
 - **CRAH (Computer Room Air Handler):** Usa agua fría (chilled water) que viene de plantas externas. Es más eficiente para centros grandes.
 - **Free Cooling:** Si el clima exterior es frío, se usa aire de afuera filtrado para enfriar, ahorrando muchísima electricidad.
 - **Liquid Cooling:** El refrigerante llega directamente a una placa sobre el procesador. Es lo que se usa hoy para **Inteligencia Artificial**, donde los chips generan calor extremo.





El **PUE (Power Usage Effectiveness)** es la métrica estándar de la industria (desarrollada por *The Green Grid*) para medir la eficiencia energética de un centro de datos.

- **1. La Fórmula Fundamental**

- La relación es simple: cuánta energía entra al edificio comparado con cuánta llega realmente a los servidores.
- **PUE** = Energía Total de la Instalación / Energía de los Equipos IT
- **Energía de Equipos IT:** Servidores, almacenamiento, redes.
- **Energía Total:** IT + Refrigeración + Iluminación + Pérdidas en UPS/Transformadores.

Tabla de Referencia de Eficiencia

PUE	Nivel de Eficiencia
3.0	Muy ineficiente (común en salas de servidores viejas)
2.0	Promedio histórico de la industria
1.5	Eficiente
1.2	Muy eficiente (estándar Google/AWS/Azure)
1.0	El ideal

Ejemplos Prácticos de Cálculo

- **Escenario A:**

- Centro de Datos Tradicional (Legacy)
- Este centro tiene un sistema de refrigeración antiguo y mucha pérdida de calor.
- Carga IT: 200 kW
- Refrigeración: 150 kW
- Pérdidas en UPS y Luces: 50 kW
- Energía Total: $200 + 150 + 50 = 400$ kW

$$\text{PUE} = 400 \text{ kW} / 200 \text{ kW} = 2$$

Nota: Por cada watt que consumen los servidores, gastas otro watt entero en mantenerlos encendidos y fríos

- **Escenario B:**

- Centro de Datos Moderno (Eficiente)
- Utiliza pasillos contenidos, baterías de litio y refrigeración inteligente.
- Carga IT: 200 kW
- Refrigeración: 30 kW
- Pérdidas en UPS y Luces: 10 kW
- Energía Total: $200 + 30 + 10 = 240$ kW

$$\text{PUE} = 240 \text{ kW} / 200 \text{ kW} = 1.4$$

Nota: Solo gastas un 20% extra en infraestructura

El DCiE

- A veces se usa el **DCiE (Data Center Infrastructure Efficiency)**, que es simplemente el PUE expresado en porcentaje:

$$\text{DCiE} = (1/\text{PUE}) \times 100$$

Tipos de Almacenamiento en el Datacenter

- El almacenamiento se clasifica según **cómo se accede a los datos** y la **latencia** requerida.

Tipo de Almacenamiento	Unidad Mínima	Uso Común	Protocolos
Bloque (Block)	Bloques de datos	Bases de datos, Sistemas Operativos	iSCSI, Fibre Channel
Archivo (File)	Archivos y Carpetas	Repositorios compartidos, documentos	NFS, SMB/CIFS
Objeto (Object)	Objetos con metadatos	Backups, Big Data, imágenes web	HTTP/HTTPS (REST API)

- **Almacenamiento de Bloque (Block Storage)**
 - Es el más rápido. El almacenamiento se presenta al servidor como si fuera un disco duro físico vacío. El sistema operativo del servidor es quien decide el sistema de archivos. Es ideal para aplicaciones de **alto rendimiento**.
- **Almacenamiento de Archivo (File Storage)**
 - Se basa en una jerarquía de carpetas. Es lo que conocemos comúnmente como un NAS (Network Attached Storage). Es ideal cuando varios usuarios necesitan **colaborar sobre los mismos archivos**.
- **Almacenamiento de Objetos (Object Storage)**
 - Es la base del almacenamiento en la nube (como AWS S3 o Azure Blob). No hay carpetas; cada archivo es un "objeto" con un ID único y metadatos extensos. Es **infinitamente escalable** y más económico, aunque tiene una latencia mayor que el de bloque.



Tecnologías de Virtualización

¿Qué es un Hypervisor?

- Es la capa de software que permite la **virtualización**. Su función principal es separar el software del hardware.
- Permite que una sola computadora física (Host) ejecute múltiples máquinas virtuales (Guests) de forma simultánea, repartiendo los recursos de CPU, memoria, almacenamiento y red entre ellas.

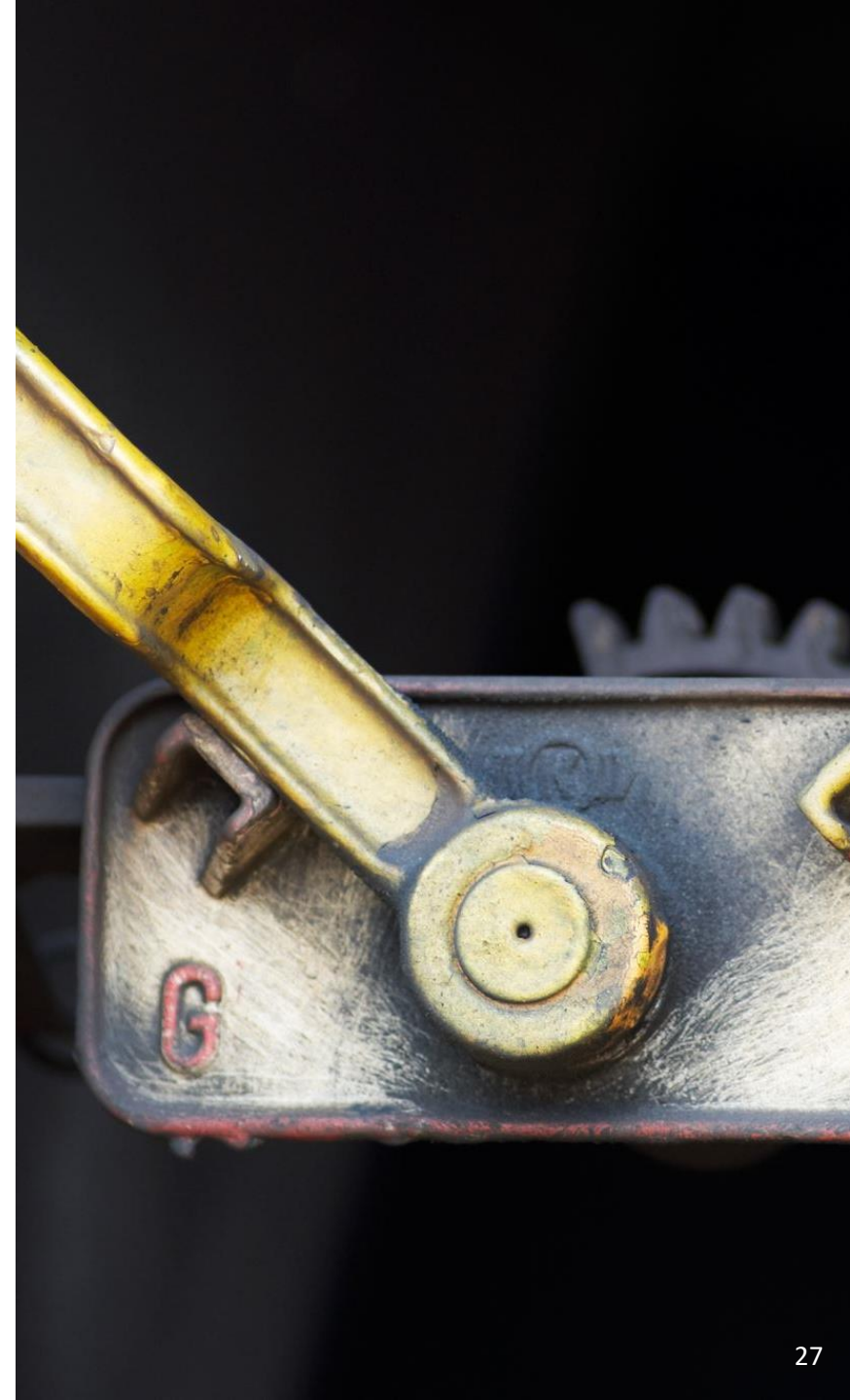


La Analogía del Edificio:

- Si el hardware es un terreno, el Hypervisor es el arquitecto que construye varios departamentos (máquinas virtuales) independientes.
- Los vecinos (Sistemas Operativos) viven en el mismo terreno, pero no se ven entre sí y cada uno cree que tiene su propia casa.

Dos tipos fundamentales

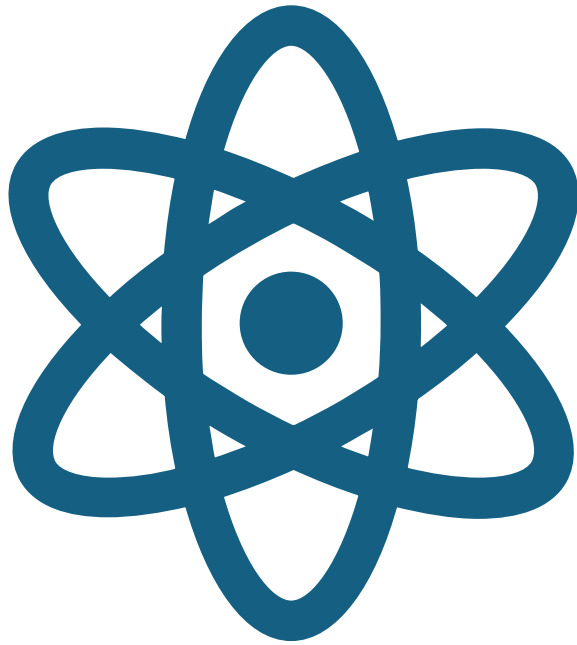
- **Tipo 1: Nativo o "Bare Metal"**
 - Se instala directamente sobre el hardware físico. No necesita un sistema operativo previo.
 - **Cómo funciona:** Toma el control total del hardware desde el arranque.
 - **Ventajas:** Máximo rendimiento, alta seguridad (menos capas de software que atacar).
 - **Uso:** Centros de datos, servidores empresariales y la **nube pública** (AWS, Azure, GCP).
 - **Ejemplos:** VMware ESXi, Microsoft Hyper-V, Xen, KVM.
- **Tipo 2: Alojado o "Hosted"**
 - Se instala como una aplicación más dentro de un sistema operativo convencional (Windows, Linux, macOS).
 - **Cómo funciona:** El Hypervisor le pide recursos al sistema operativo anfitrión, y este se los pide al hardware.
 - **Ventajas:** Muy fácil de instalar y usar para pruebas o desarrollo personal.
 - **Desventajas:** Menor rendimiento (latencia por la capa extra del SO).
 - **Ejemplos:** Oracle VirtualBox, VMware Workstation, Parallels Desktop.



Característica	Tipo 1 (Bare Metal)	Tipo 2 (Hosted)
Instalación	Sobre el hardware	Sobre un SO (Windows/Linux)
Rendimiento	Excelente / Nativo	Medio (con latencia)
Estabilidad	Muy alta	Depende del SO anfitrión
Uso común	Servidores en la Nube	Estudiantes / Desarrolladores

Funciones Clave

- **Aislamiento (Sandboxing):** Si una máquina virtual falla o es infectada por un virus, las demás siguen funcionando perfectamente. El Hypervisor garantiza que no haya "fugas" entre ellas.
- **Abstracción:** El Hypervisor le presenta a la máquina virtual un hardware genérico. Esto permite que una VM se mueva de un servidor físico a otro totalmente distinto sin problemas de drivers.
- **Programación de Recursos (Scheduling):** Decide qué VM tiene prioridad para usar el procesador en cada milisegundo, evitando que una sola máquina monopolice todo el servidor.



¿Por qué es la base de la Computación en la Nube?

- **Consolidación:** En lugar de tener 10 servidores físicos al 10% de uso, tienen 1 servidor físico con 10 máquinas virtuales al 90% de eficiencia.
- **Elasticidad:** Se pueden crear o destruir máquinas virtuales en segundos según la demanda.
- **Continuidad de Negocio:** Permite hacer "Snapshots" (fotos del estado de la máquina) o migraciones en vivo sin apagar el servicio.

VMs vs. Contenedores

- **La Diferencia Estructural**

- La distinción clave reside en qué nivel se realiza la virtualización: el **hardware** o el **sistema operativo**.
- **Máquinas Virtuales (Virtualización de Hardware)**
 - Una VM incluye la aplicación, los binarios/librerías necesarios y un **Sistema Operativo Invitado (Guest OS)** completo.
 - **Hypervisor:** Es la capa de software (como VMware, KVM o Hyper-V) que corre sobre el hardware físico y permite crear múltiples instancias aisladas.
 - **Aislamiento:** Es total. Cada VM cree que tiene su propio hardware dedicado. Si el Guest OS falla o es atacado, el impacto suele quedar confinado a esa VM.
- **Contenedores (Virtualización de SO)**
 - Los contenedores comparten el **Kernel del Sistema Operativo Host**. Solo empaquetan la aplicación y sus dependencias de usuario.
 - **Motor de Contenedores:** (Como Docker) Se encarga de asignar recursos y aislar los procesos.
 - **Eficiencia:** Al no tener que arrancar un sistema operativo completo, son increíblemente ligeros (megabytes vs. gigabytes de una VM).

- **Comparativa Técnica**

Característica	Máquinas Virtuales	Contenedores
Aislamiento	Alto (Nivel de Kernel propio)	Moderado (Comparten Kernel)
Peso	Pesado (GBs por el OS)	Ligero (MBs)
Tiempo de Inicio	Minutos	Segundos
Portabilidad	Limitada por el Hypervisor	Alta (Corre igual en cualquier motor)
Escalabilidad	Lenta y costosa en recursos	Rápida y fluida

¿Cuándo usar cada uno?

- **Elegir Máquinas Virtuales si:**
 - Necesitas ejecutar una aplicación que requiere un **Kernel de SO específico** (ej. una app de Windows en un servidor Linux).
 - Requieres un **aislamiento de seguridad estricto** (entornos multi-tenant sensibles).
 - Vas a gestionar aplicaciones legacy monolíticas que no están diseñadas para microservicios.
- **Elegir Contenedores si:**
 - Estás trabajando con una arquitectura de **microservicios**.
 - Necesitas maximizar la **densidad de aplicaciones** en un solo servidor físico.
 - Buscas agilidad en procesos de **CI/CD** (integración y despliegue continuo), donde el entorno de desarrollo debe ser idéntico al de producción.

Ciclo de Vida del Recurso



- En la nube es un concepto fundamental para cualquier arquitecto o administrador de infraestructura. No se trata solo de "crear y borrar", sino de un proceso iterativo que garantiza eficiencia, seguridad y, sobre todo, control de costos (FinOps).

1. Planificación y Diseño



- **Definición de necesidades:** ¿Qué cómputo, almacenamiento o red se requiere?
- **Selección del modelo:** ¿IaaS, PaaS o SaaS?
- **Estimación de costos:** Uso de calculadoras de precios para evitar sorpresas.
- **Cumplimiento y Seguridad:** Definir quién tendrá acceso (IAM) y en qué región se desplegará.

2. Aprovisionamiento (Provisioning)



- **Despliegue:** Puede ser manual (consola), mediante CLI o, idealmente, a través de **Infraestructura como Código (IaC)** como Terraform o CloudFormation.
- **Configuración:** Instalación de software, parches iniciales y configuración de redes (VPCs, Subnets).

3. Operación y Gestión



- **Monitoreo:** Seguimiento de métricas de rendimiento (CPU, RAM, Latencia).
- **Mantenimiento:** Aplicación de actualizaciones de seguridad y backups.
- **Escalabilidad:** Ajuste de recursos según la demanda (Auto-scaling).

4. Optimización (Fase Crítica de FinOps)



- **Right-sizing:** Ajustar el tamaño del recurso si está infrautilizado (por ejemplo, bajar el shape de una instancia)
- **Análisis de Costos:** Identificar "recursos zombies" que generan gasto sin uso.



5. Retiro y Eliminación (Decommissioning)

- **Respaldo final:** Realizar un snapshot o backup de los datos si es necesario.
- **Limpieza de dependencias:** Eliminar no solo la instancia, sino también sus discos asociados (EBS), direcciones IP estáticas y registros de DNS.
- **Auditoría:** Confirmar en la factura que el cargo ha cesado.