

Comparativa técnica entre servicios de cómputo

Este documento presenta una comparativa técnica entre tres servicios de cómputo en la nube: máquinas virtuales (VMs), contenedores y cómputo sin servidor (serverless). La comparación se realiza en base a criterios como nivel de control, escalabilidad, costos, tiempo de arranque, mantenimiento, ventajas, limitaciones y casos de uso recomendados.

| Criterio | VMs (Máquinas Virtuales) | Contenedores | Serverless |
|---------------------|--|--|---|
| Nivel de control | Total: SO, middleware, configuraciones | Medio: entorno y librerías, no SO | Bajo: solo código y eventos |
| Escalabilidad | Manual o autoscaling (más lento) | Automática con Kubernetes | Totalmente automática e instantánea |
| Costos | Por tiempo activo (aunque no se use) | Por recursos reservados/consumidos | Por número de ejecuciones y duración |
| Tiempo de arranque | Minutos | Segundos | Milisegundos |
| Mantenimiento | Alto: parches, seguridad, SO | Moderado: gestión de imágenes/clusters | Nulo: lo gestiona el proveedor |
| Facilidad de uso | Requiere administración de sistemas | Requiere conocimientos DevOps/CI-CD | Muy fácil, enfocado en el código |
| Ventajas | Flexibilidad, compatibilidad con legacy | Portabilidad, rapidez, eficiencia | Pago exacto por uso, escalado sin esfuerzo |
| Limitaciones | Costoso y poco ágil | Gestión compleja de clusters | No apto para cargas largas o persistentes |
| Caso ideal | Migrar ERP/CRM legacy, sistemas críticos | Microservicios, pipelines de datos | Procesamiento de imágenes, APIs, eventos |
| Caso no recomendado | Apps con tráfico variable y bajo presupuesto | Aplicaciones monolíticas grandes | Aplicaciones de larga duración o alta I/O constante |

Conclusión

- VMs: adecuadas cuando se necesita máximo control o migrar sistemas legacy sin rediseño.
- Contenedores: recomendados para microservicios y entornos modernos con despliegue continuo.
- Serverless: ideal para funciones puntuales, tráfico impredecible y bajo costo.