

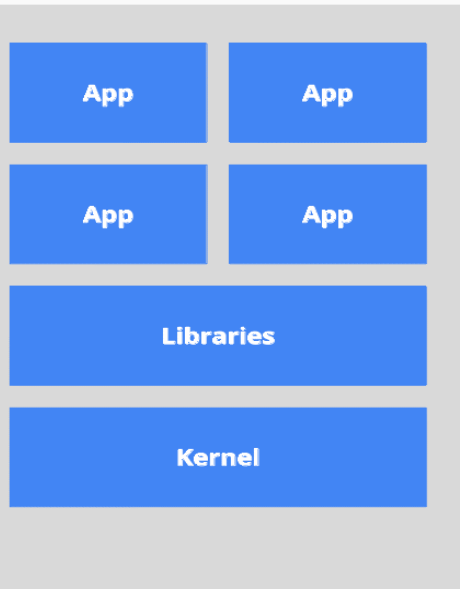
# Entendiendo qué es y cómo funciona



**kubernetes**

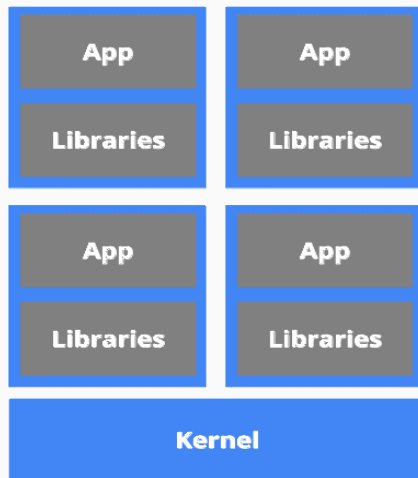
# 1. ¿Qué es Kubernetes?

**The old way:** Applications on host



*Heavyweight, non-portable  
Relies on OS package manager*

**The new way:** Deploy containers

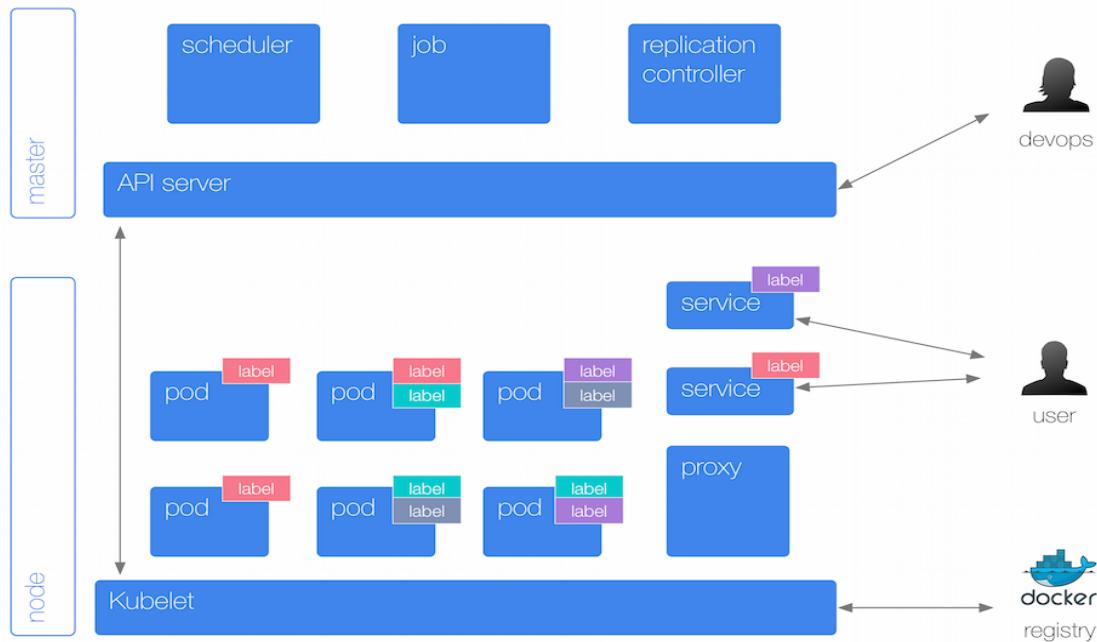


*Small and fast, portable  
Uses OS-level virtualization*

Sistema de código abierto para la orquestación de **contenedores.**

Máquinas virtuales vs contenedores.

## 2. Arquitectura de Kubernetes



Sigue una arquitectura maestro-esclavo.

**Maestro:** Encargado de planificar el resto del sistema.

**Esclavo:** Cada uno de los nodos que puede interactuar con un usuario..

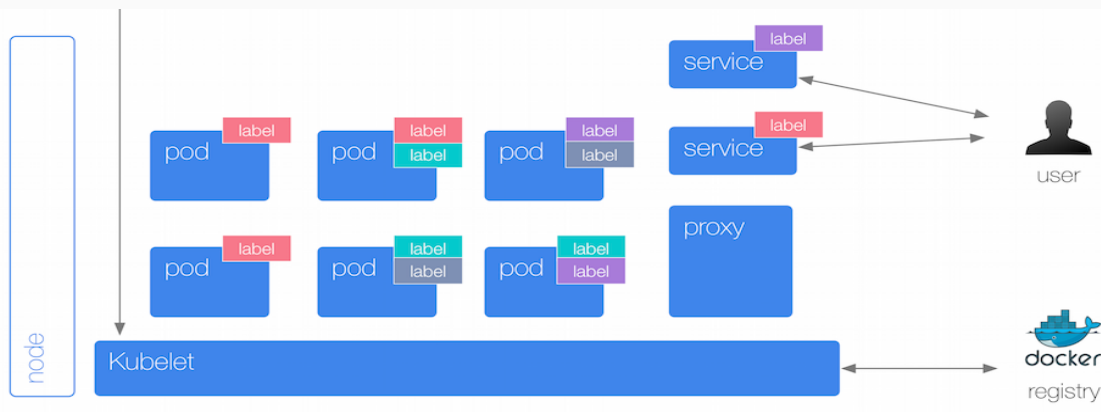
## 2.1 Nodo esclavo

**Pod:** Conjunto de uno o más contenedores que constituye la unidad mínima con la que trabaja Kubernetes.

**cAdvisor (en Kubelet):** Monitorización.

**Kubelet:** Gestiona el nodo y se comunica con el nodo maestro.

**kube-proxy:** Proxy de red y balanceador de carga interno.



## 2.2 Nodo maestro (o de control)



**kubectl**

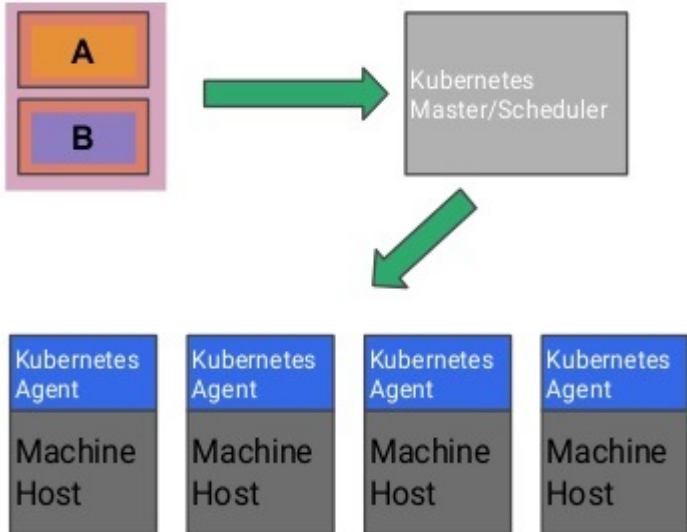
**API server /etcd**

**Trabajos (jobs)**

- **Controlador de replicación (Replication controller)**

### 3. El planificador de Kubernetes

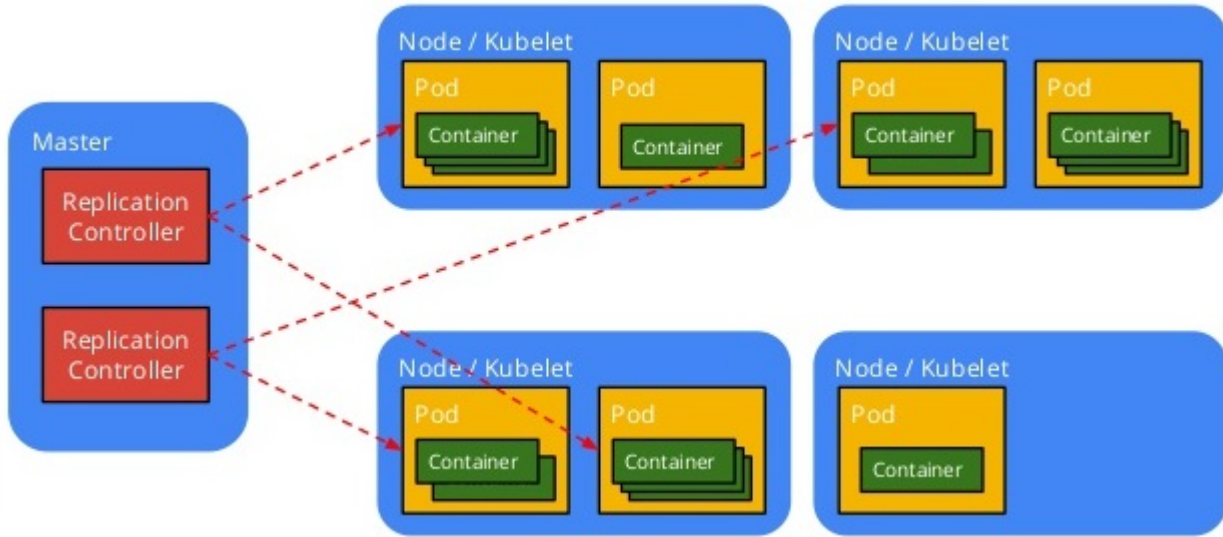
Pods



**Predicados:** Reglas que han de cumplirse para que se pueda seleccionar un nodo.

**Función de prioridad:** Valor calculado para determinar qué nodo es el mejor. Se pueden aplicar varios criterios y se toma el nodo con mayor función de prioridad.

## 4. Autoescalado en Kubernetes



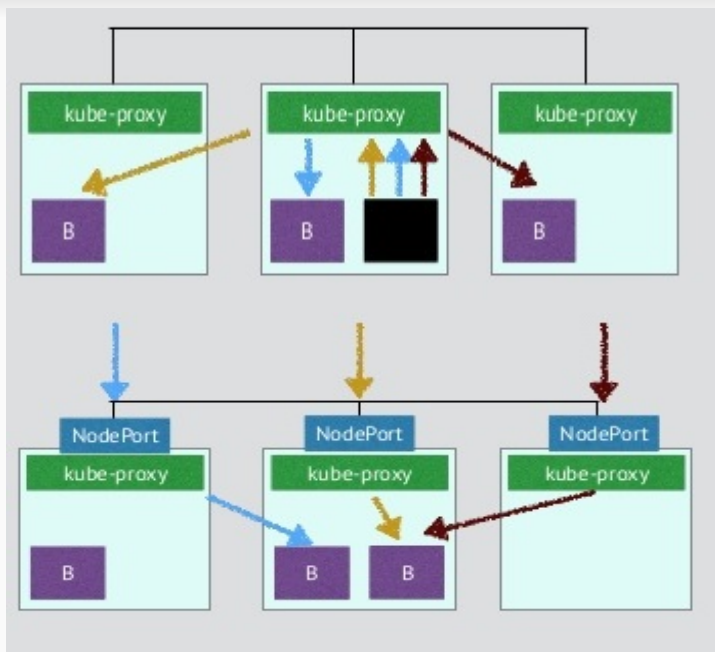
Escalado horizontal vs vertical

HORIZONTAL

**Cluster autoscaling**

**Horizontal pod autoscaling**

## 5. Balanceo de carga en Kubernetes



**Interno:**

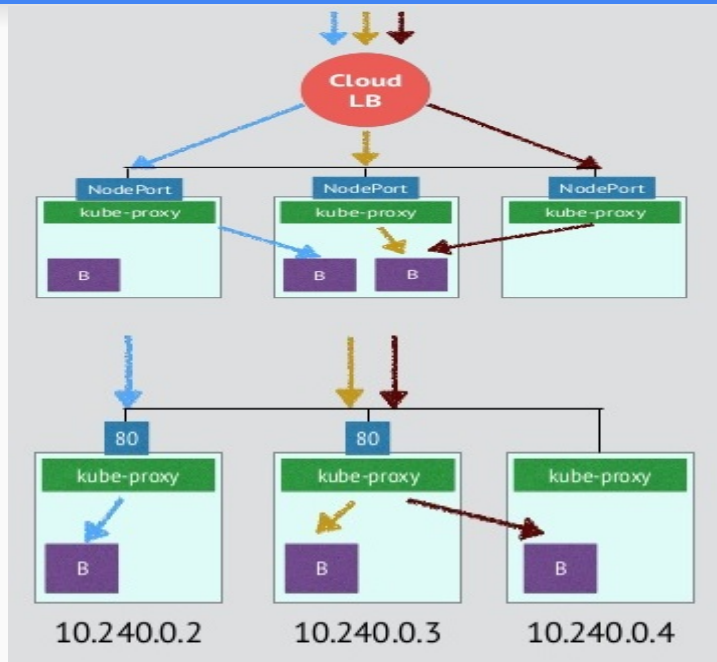
**kube-proxy**

**Externos:**

**NodePort**



# 5. Balanceo de carga en Kubernetes



Externos (continuación)

**LoadBalancer**

**External IPs**

**Ingress**