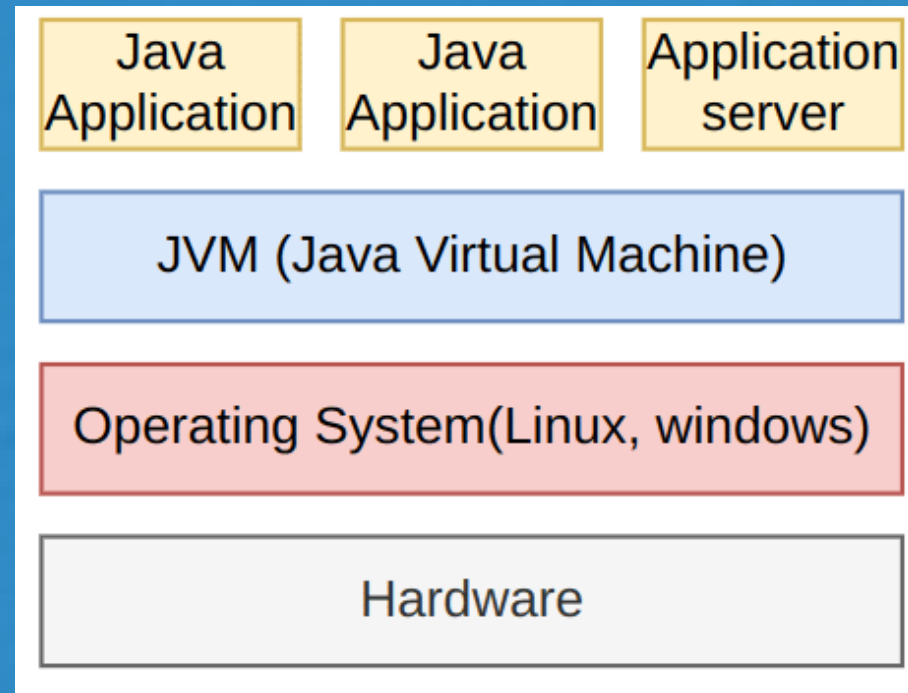
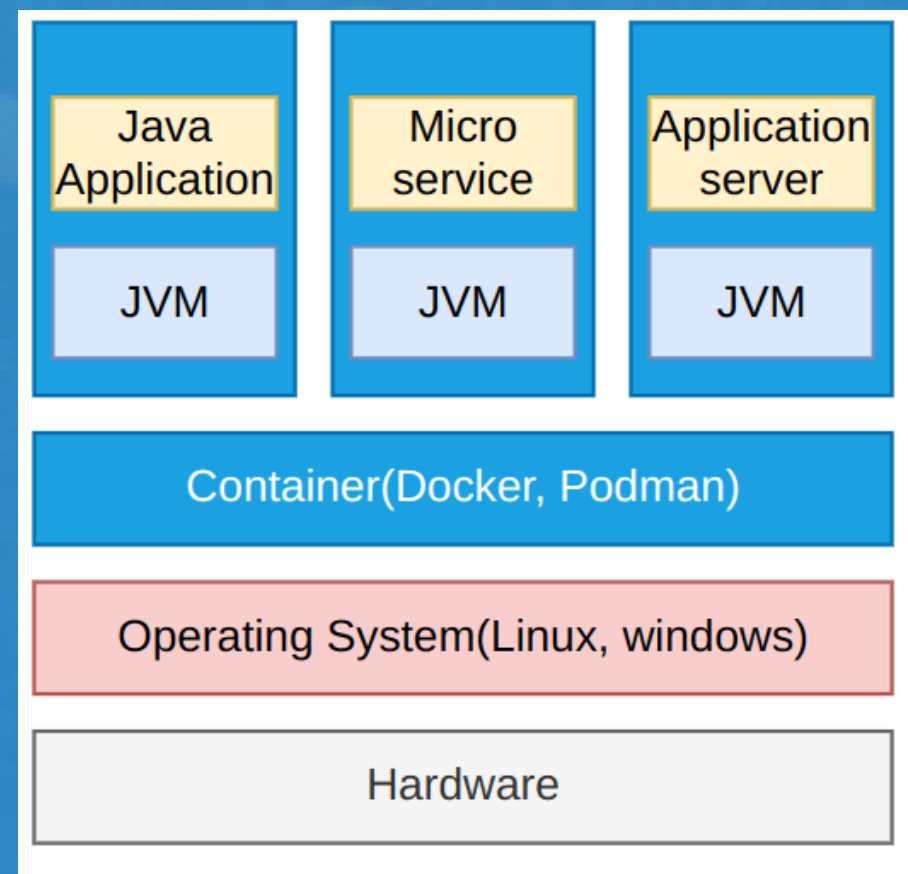


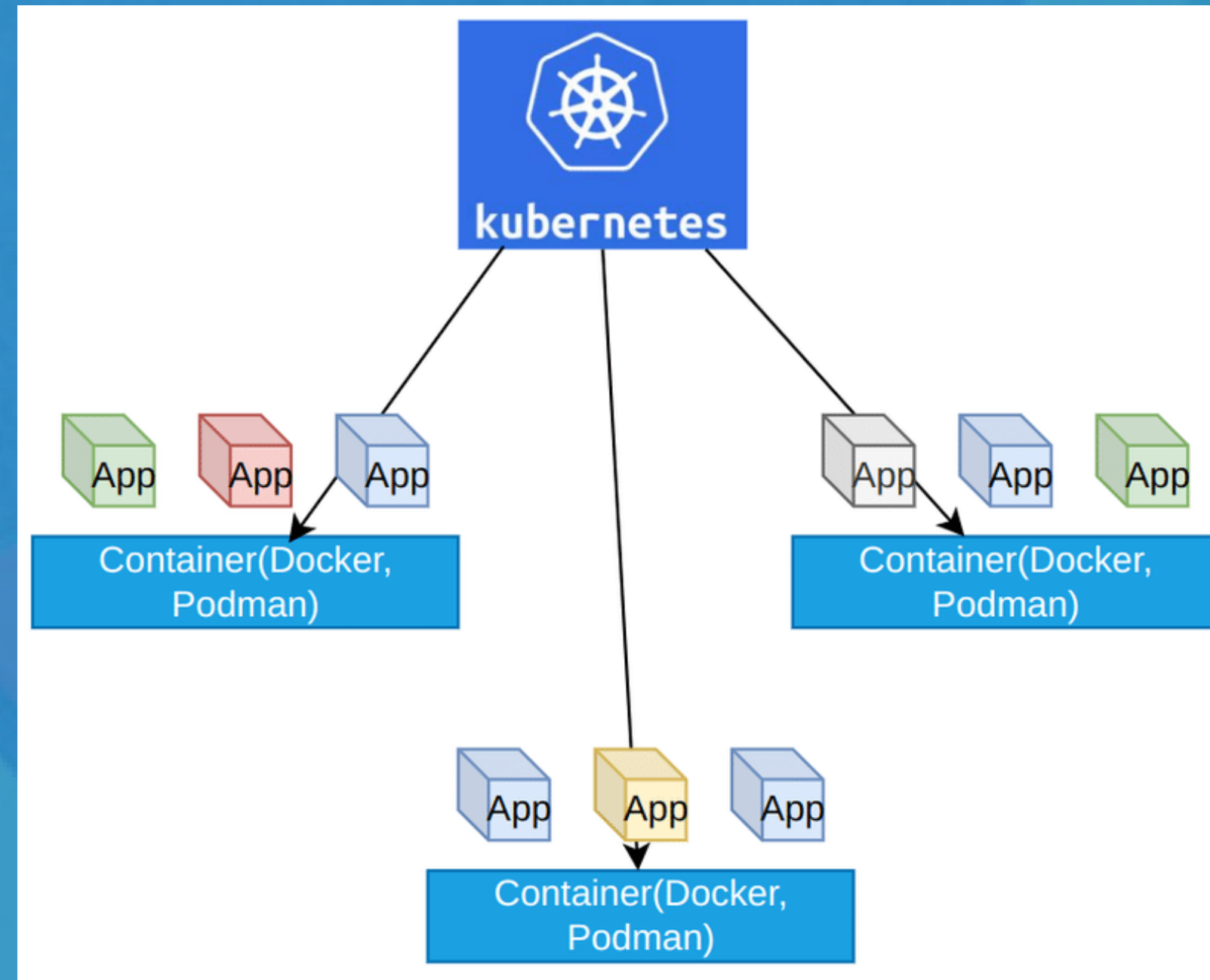
1- Application Server



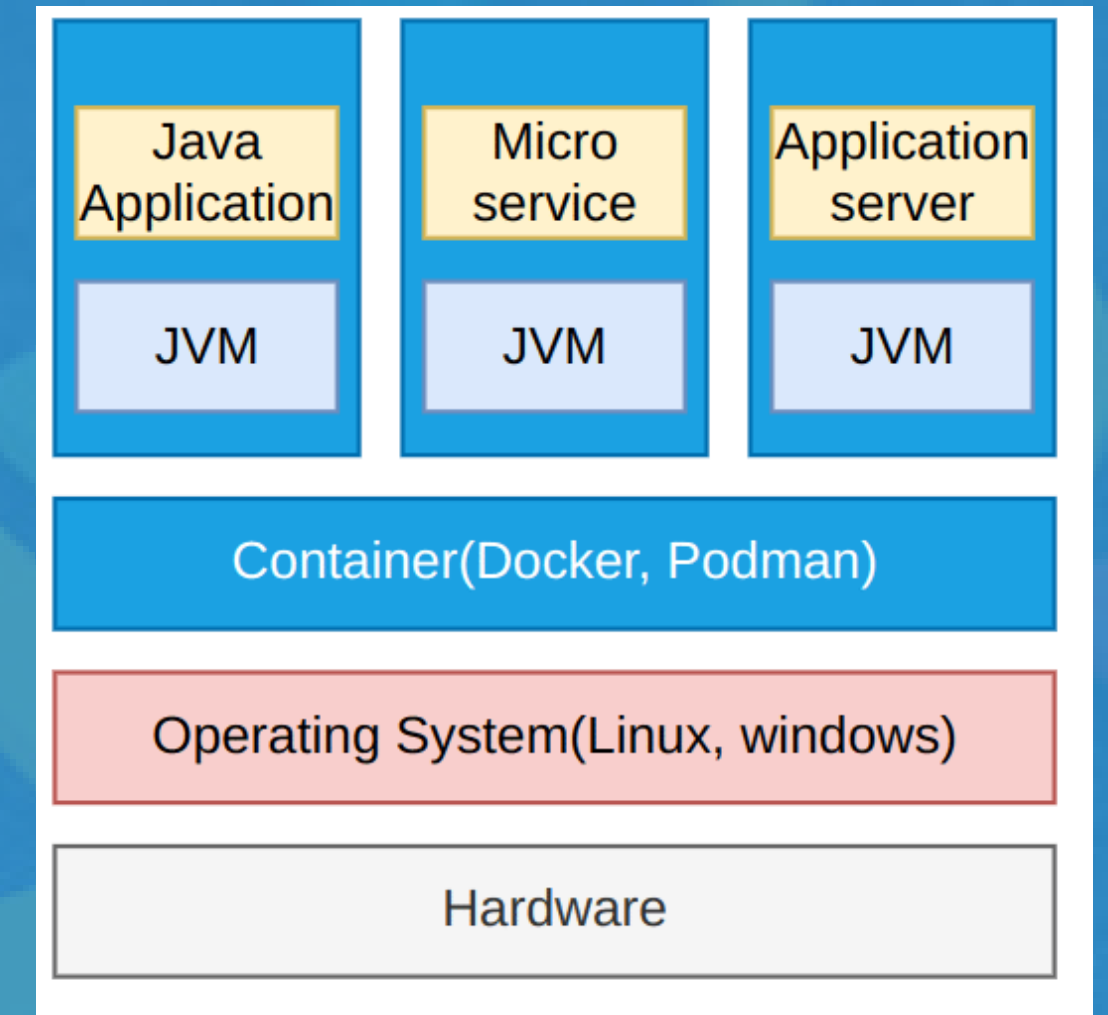
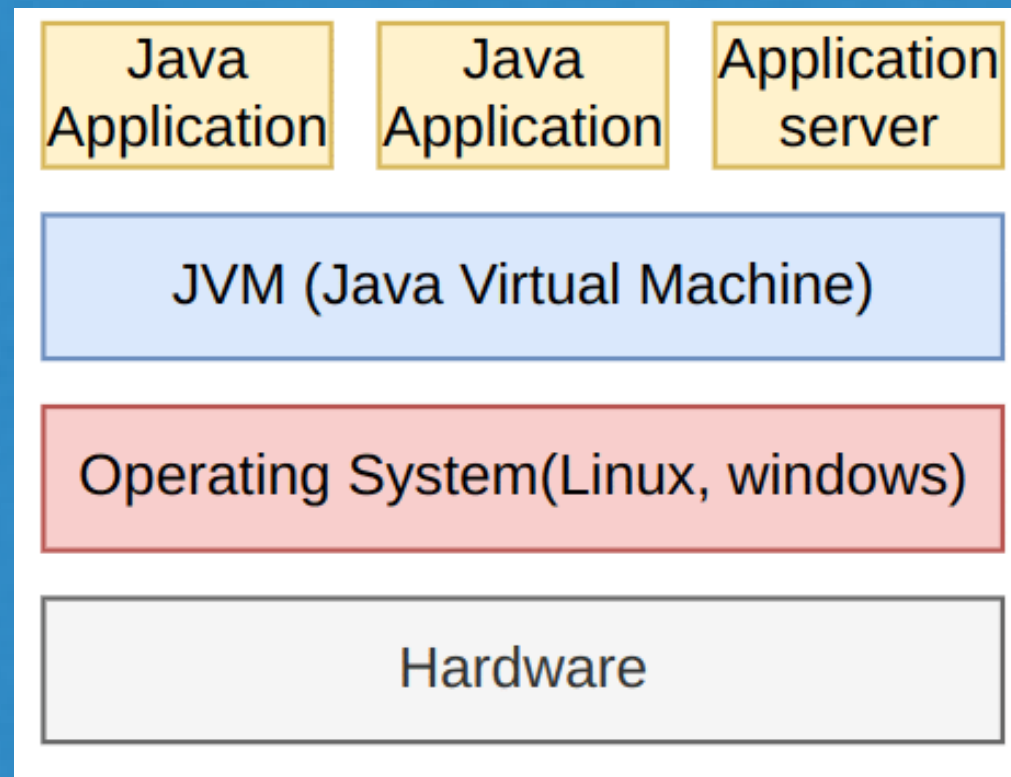
2- containerized APP



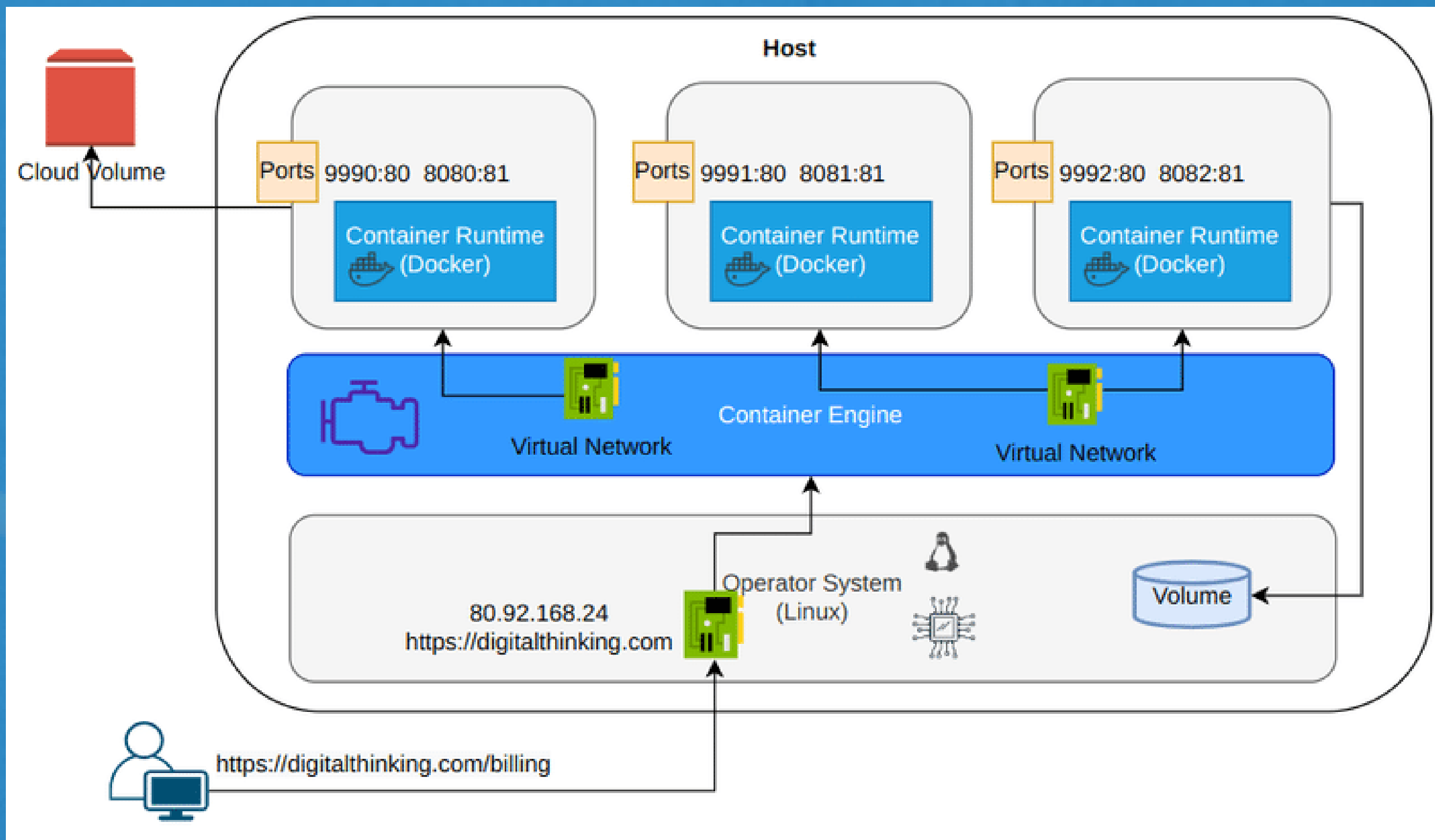
3- Containerized app in cluster Kuberentes orquestrator



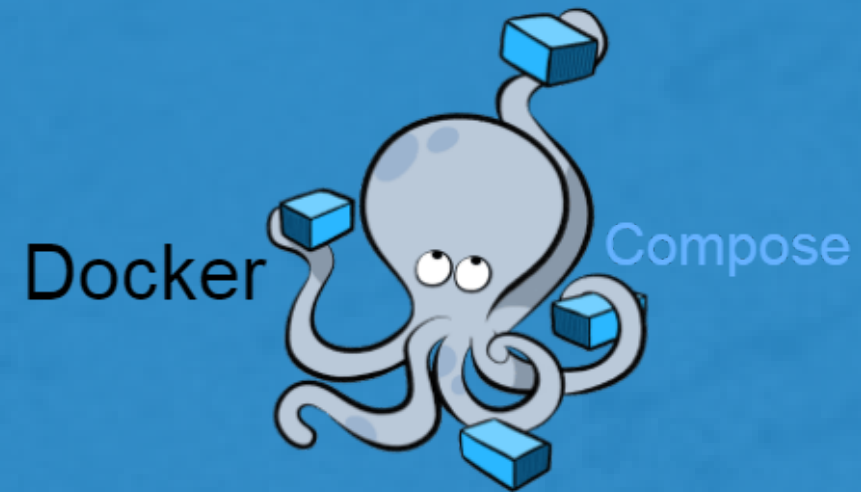
Contenedores



- **Contenedores:** Unidad de software que empaqueta el código y todas las dependencias necesarias de una aplicación.
- **Imágenes:** Paquete ligero y ejecutable de software con todo lo necesario para la aplicación.
- **Docker Engine:** Motor de ejecución de contenedores
- **Docker Hub:** Repositorio por defecto para las imágenes de docker
- **Podman:** Alternativa Opensource a Docker
- **Docker Compose:** Orquestador ligero de contendores
- **Docker Swarm:** Orquestador de contenedores permite manejar un cluster
- **Kubernetes:** Sistema para la administración de clusters y Orquestador Empresarial de contenedores.



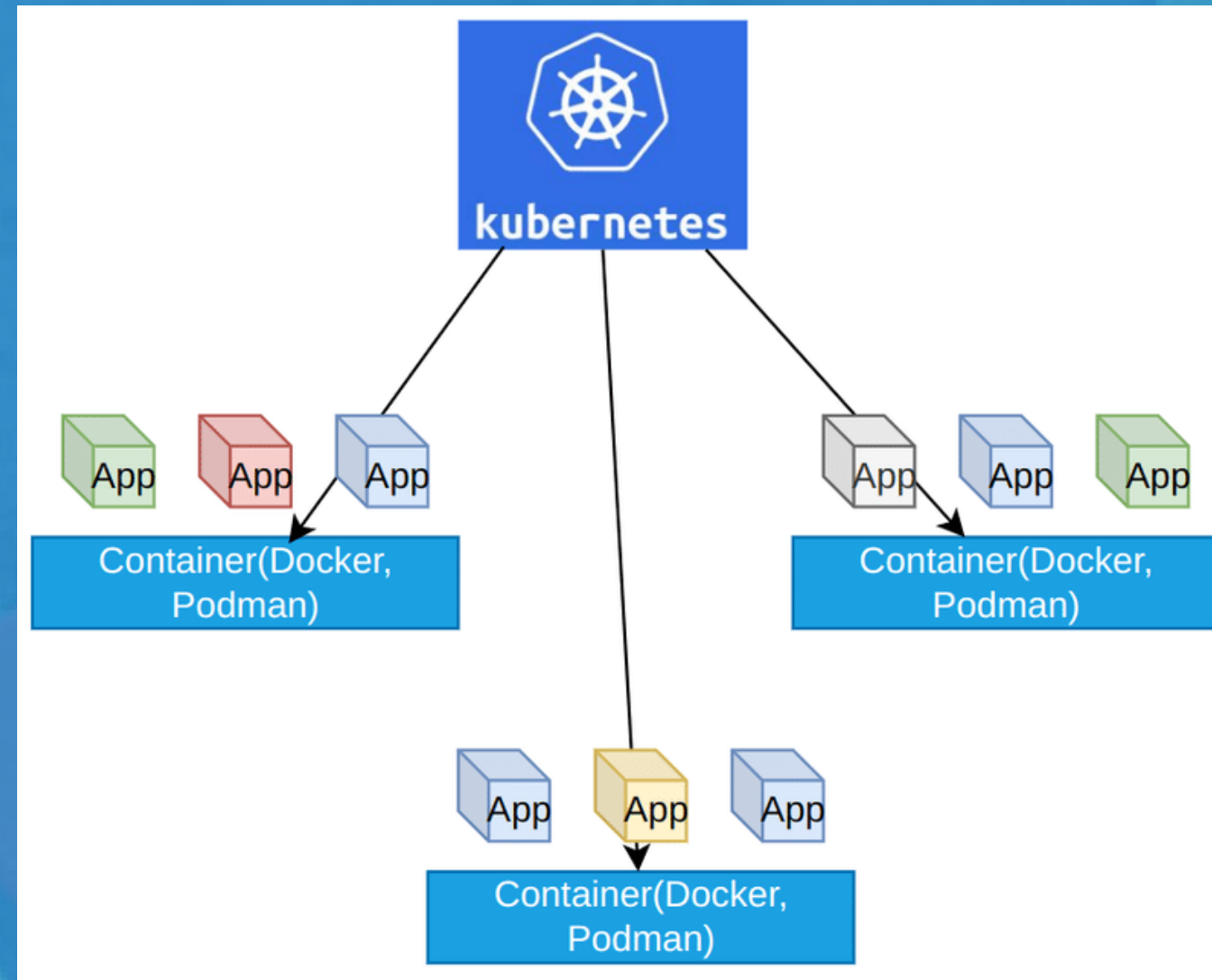
1- Docker Compose



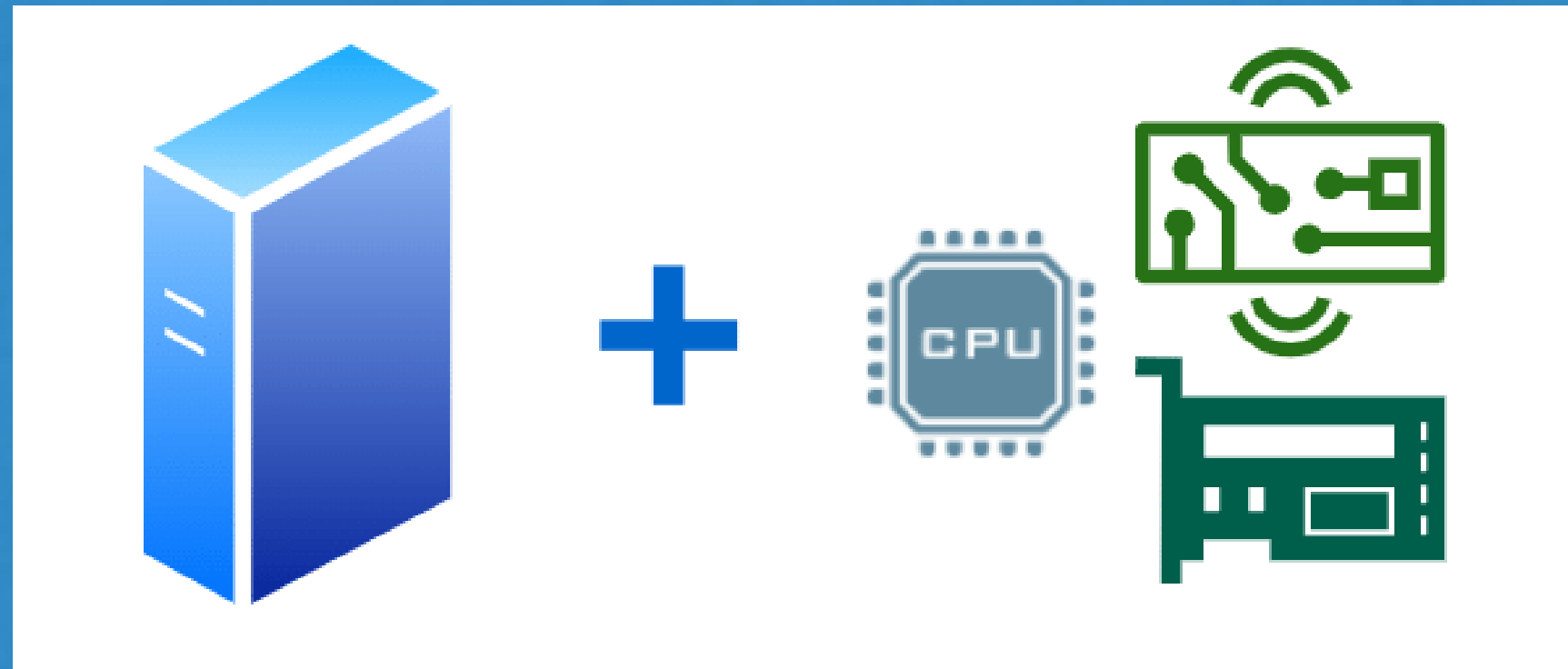
2- Docker Swarm



3- Kubernetes



1- vertical scaling

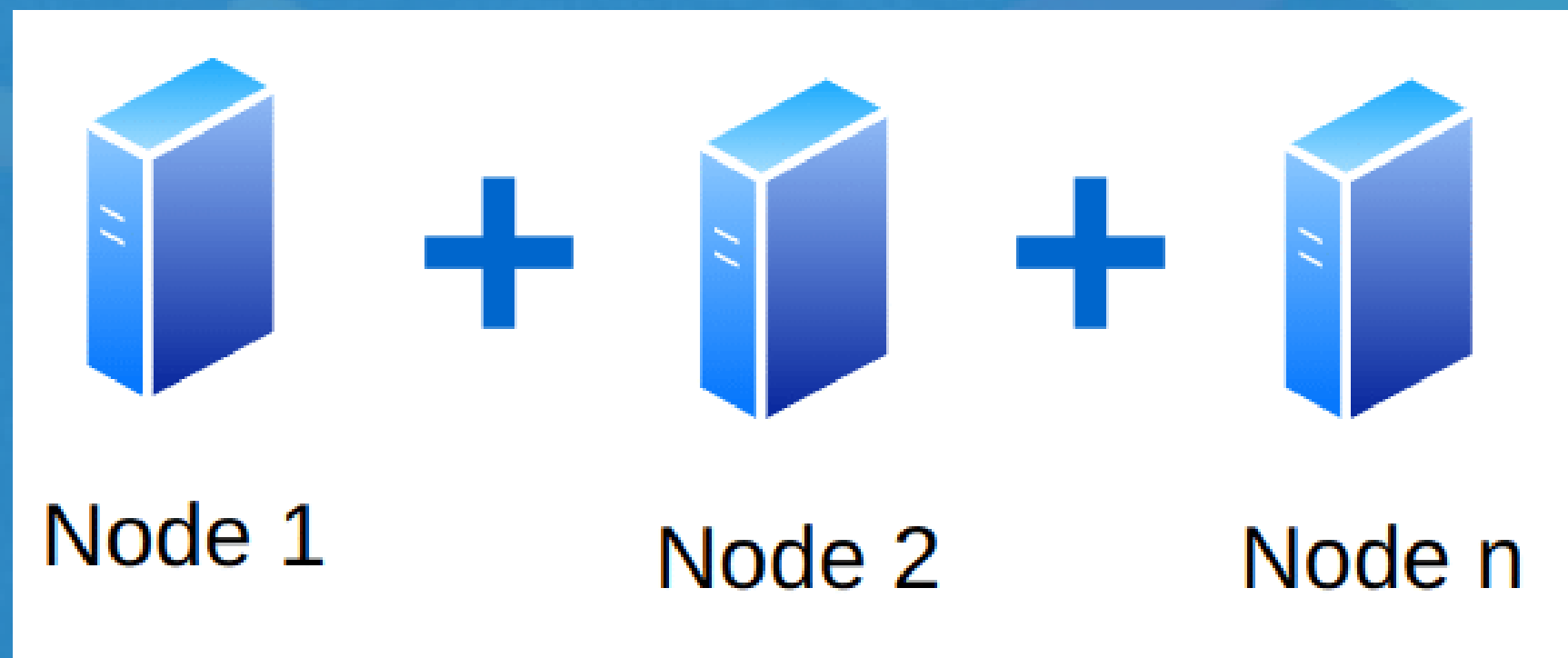


Capacidad de los sistemas para adaptarse al crecimiento.
Por demanda y complejidad.

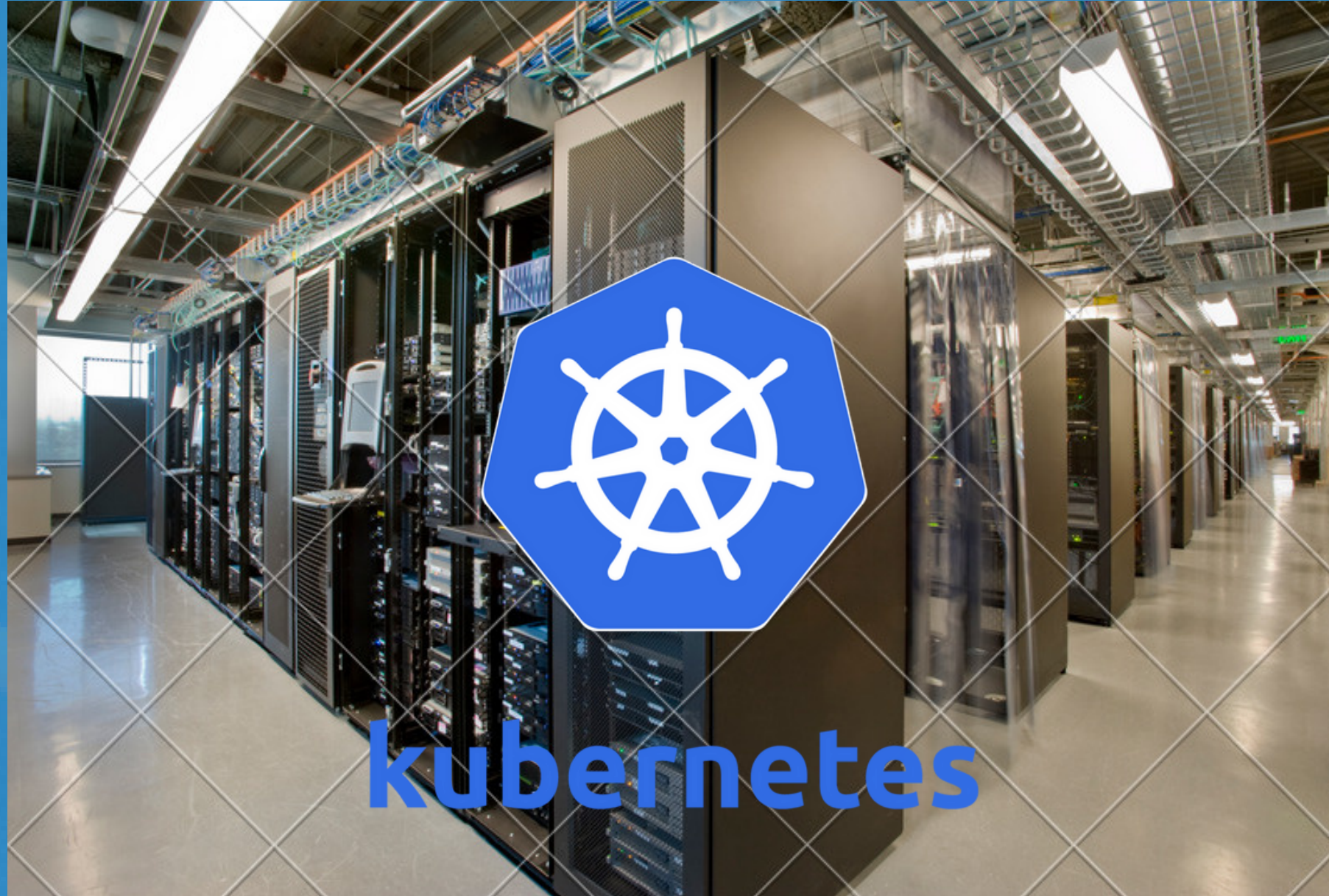
Sistema diseñado inicialmente para una carga de menos de 100 usuarios y con el tiempo su demanda de usuario aumenta a 500, 8000, 60000 etc. A medida que aumenta los usuarios y/o la complejidad del sistema se requieren más recursos (RAM, procesamiento, Almacenamiento etc)

Vertical: Agregar más recursos al mismo nodo y aumentar el poder de cómputo.
Horizontal: Agregar más nodos que se adapten a la carga de trabajo.

2- horizontal scaling



Cluster de Kubernetes

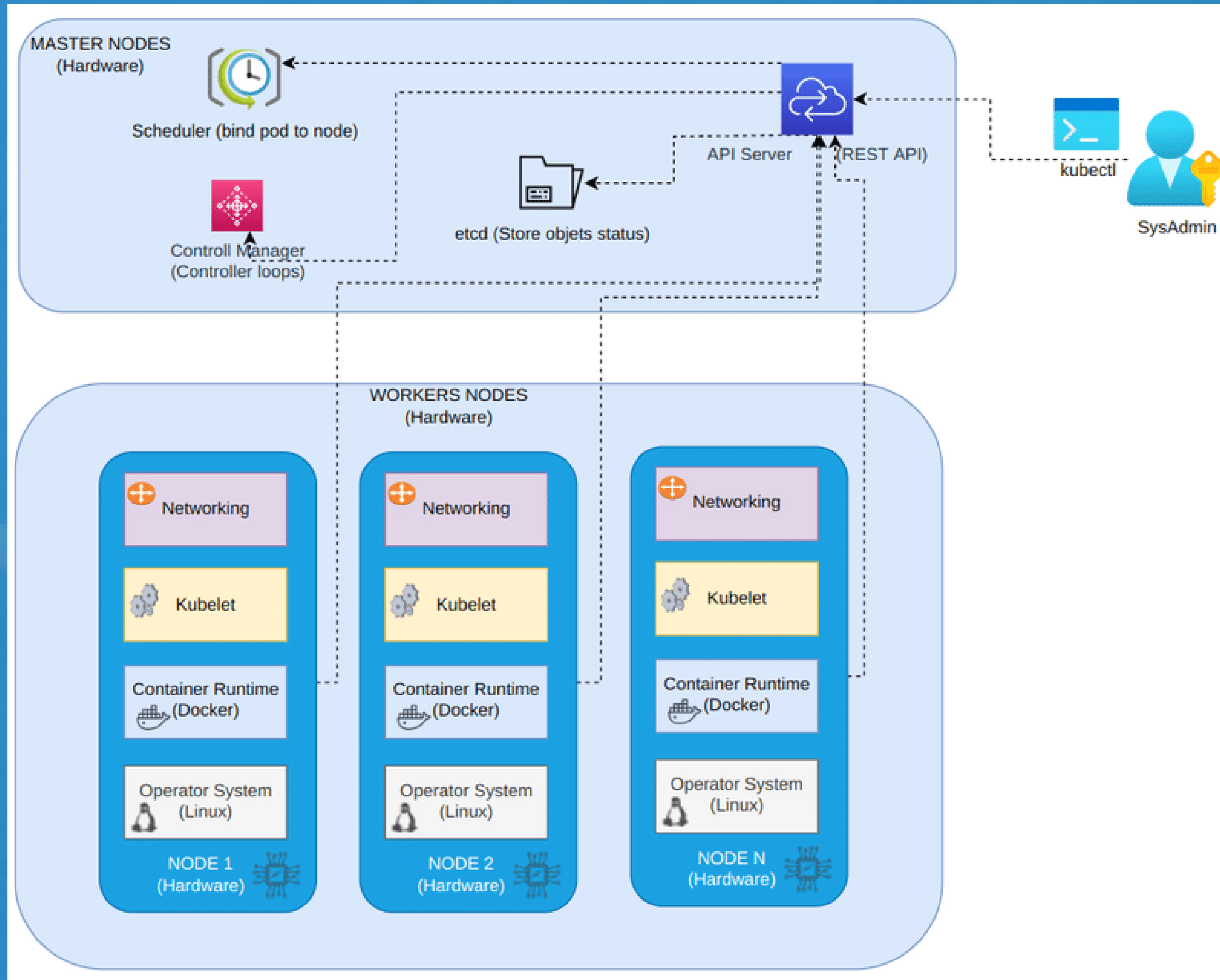


Es un sistema opensource para automatizar el despliegue, escalado y administración de aplicaciones en contenedores. También conocido como k8s (pronounced Kate's).

- Opensource
- Auto-Sanado
- Escalado horizontal
- Balanceo de carga y discovery
- Licencia Apache v2
- Creado por google en 2015
- Escrito en Go
- Ultima version estable 1.24(2022)
- Mantenido por Cloud Native Computing Foundation (parte de linux foundation).

Inspirado en el sistema Borg de google, administrador de clusters capaz de operar cientos de miles de trabajos de miles de Aplicaciones diferentes en varios clusters cada uno con hasta decenas de máquinas.

Arquitectura interna de un cluster de Kubernetes



Un cluster de kubernetes se compone de varias máquinas llamadas NODOS, que se agrupan en nodos maestros y nodos workers.

Los nodos master, controlan el cluster y los nodos workers alojan los pods que son los componentes de carga de la aplicación.

Api server: Provee la interacción para las herramientas de administración kubectl or the Kubernetes dashboard.

etcd: Almacenamiento mantiene la configuración y el estado del cluster.

Scheduler: All crear o escalar la aplicaciones selecciona el nodo para los pods y los ejecuta.

Controller-manager: Supervisa controladores más pequeños que ejecutan tareas de replicar pods y manejar operaciones de los nodos.

1- Tipos de instalaciones de Kubernetes

On-premise

- **All in one:** Se instala todo en un único nodo usando mini kube par propósitos educativos y pruebas.
- **Single master and multiworker:** Un nodo para el control panel y uno o más nodos controlados por el master
- **Single master, single etcd and multiworker:** Un nodo para el control panel, un nodo para almacenar la configuración y el estado y uno o más nodos controlados por el master.
- **Multi master, and multiworker:** Múltiples nodos para el control panel en alta disponibilidad y uno o más nodos controlados por el master en HA.

Multi master,multi etcd and multiworker: Múltiples nodos para el control panel y múltiples nodos para el almacenamiento etcd en alta disponibilidad y uno o más nodos controlados por el master en HA.

Gestionados

- AKS: Azure kubernetes service
- EKS: Amazon Elastic Kubernetes Service
- GKE: Google Kubernetes Engine
- IBM Cloud

1- Principales objetos de Kubernetes

En kubernetes todo se conoce como un objeto y estos objetos se definen en los ficheros .yaml.

Las definiciones se guardan y ejecutan en el cluster mediante el API Server.

La definición de objetos en kubernetes también se conoce como infraestructura como código.

Pods: Unidad más pequeña que se puede desplegar y gestionar en kubernetes. Es un grupo de uno o más contenedores que comparten almacenamiento y red y especificaciones de cómo ejecutarse. Son efímeros.

Deployments: Describe el estado deseado de una implementación, ejecuta múltiples réplicas de la aplicación, reemplaza las que están defectuosas o las que no responden.

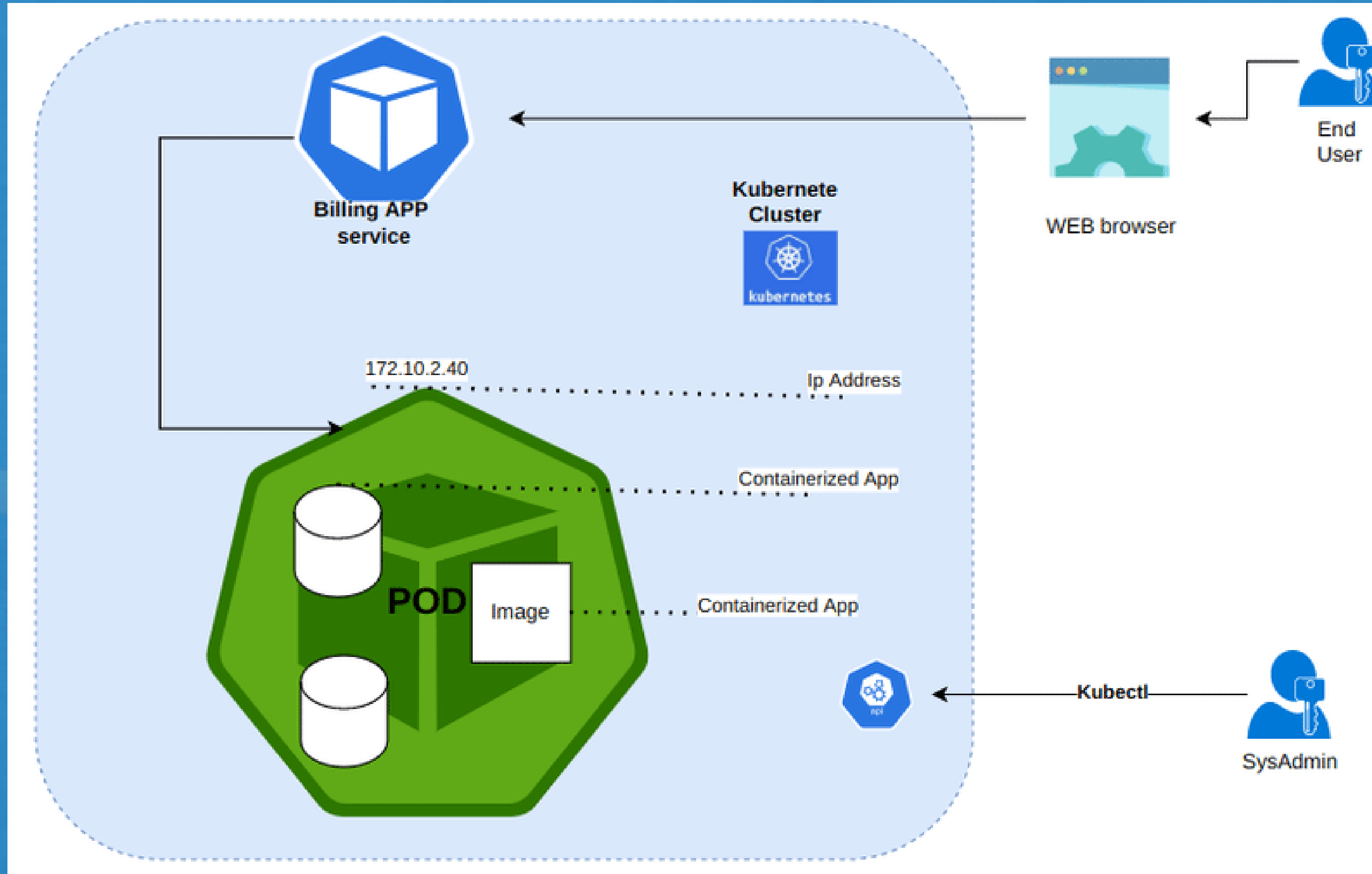
Services: Definición de cómo exponer una aplicación que se ejecuta en un conjunto de pods como un servicio de red (por defecto se usa round-robin para balanceo de carga).

Config Map: Permite desacoplar la configuración para hacer las imágenes más portables, almacenan variables de entorno, argumentos para línea de comandos, o configuración de volúmenes que pueden consumir los pods (no encriptación).

Labels: Pares de clave-valor ("environment" : "qa") para organizar, seleccionar, consultar y monitorear objetos de forma más eficiente, ideales para UI y CLIs.

Selectores: mecanismo para hacer consultas a las etiquetas. `kubectl get pods -l 'environment in (production),tier in (frontend)'`

1- Desplegar una plicación en Kubernetes



Billing Aplicación base

Datos de la factura

Número
Cliente
Detalle
Monto
Crear

Listado de facturas

Número Cliente Detalle Monto

