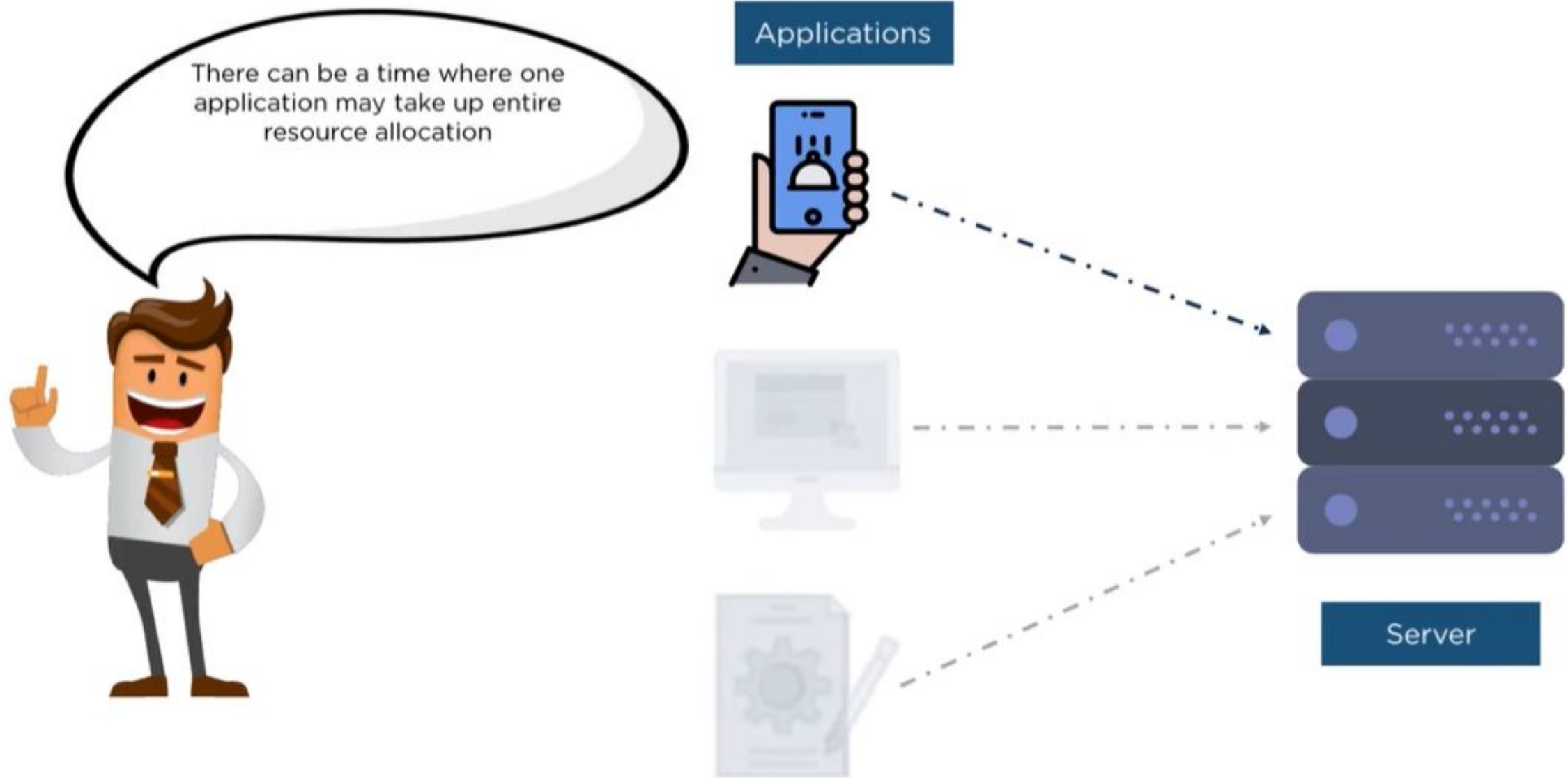


# Introduction to Kubernetes

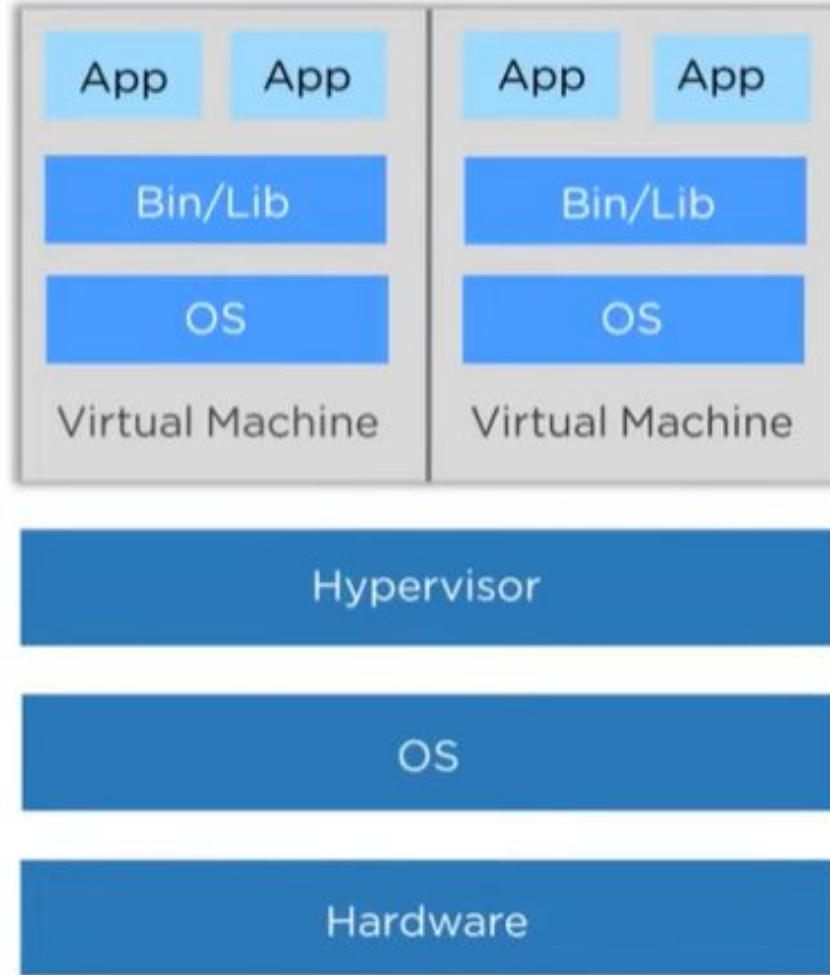


# Before Kubernetes – Traditional Deployment



# Before Kubernetes – Virtualization Deployment

Yes, but VM's have a few demerits as well



VM's are preferred as they have better resource utilization, better scalability, less cost, and much more



# Virtual Machine vs Kubernetes

Major differences are:



Not secured



Not easily portable



Time is more



Security Risk

Portability

Time Consuming



**kubernetes**



Secured



Portable



Time is less



Hypervisor

OS

Hardware



VM - It is time-consuming, as resource allocation had an extended wait time

Kubernetes - It can automatically allocate resources to the nodes that are free and hence saves much time



Container Runtime

OS

Hardware



# What is Kubernetes?

---

In practice, Kubernetes is most commonly used alongside Docker for better control and implementation of containerized applications



**kubernetes**



**docker**

**Note:** Containerized Applications means bundling an application together with all its files, libraries, and packages required for it to run reliably and efficiently on different platforms

# Did you know?

Google initially developed Kubernetes



Kubernetes was introduced as a project at Google, and it was a successor of Google Borg

It was started in 2014 because architecture of Kubernetes made it convenient for applications to run on the cloud

Cloud-Native Computing Foundation has currently maintained Kubernetes

# Benefits of Kubernetes

Portable and  
100% open-  
source

Workload  
scalability

High availability

Designed for  
deployment

Service  
discovery and  
load balancing

Kubernetes specifies how much CPU and RAM each container needs. Once resources are defined, managing the resources and making decisions for them becomes easy

Storage  
orchestration

Self healing

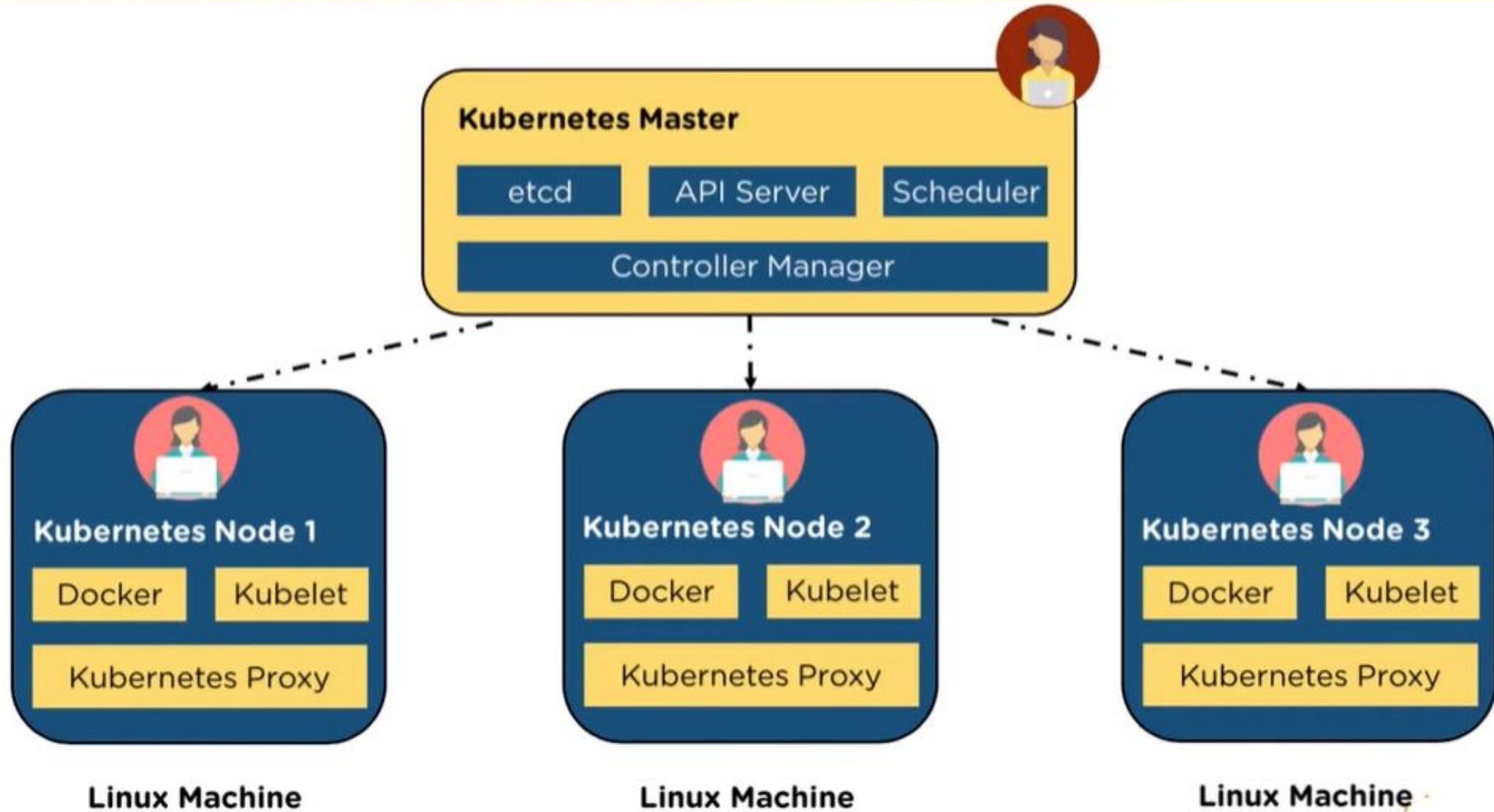
Automated  
rollouts and  
rollbacks

Automatic  
bin packing

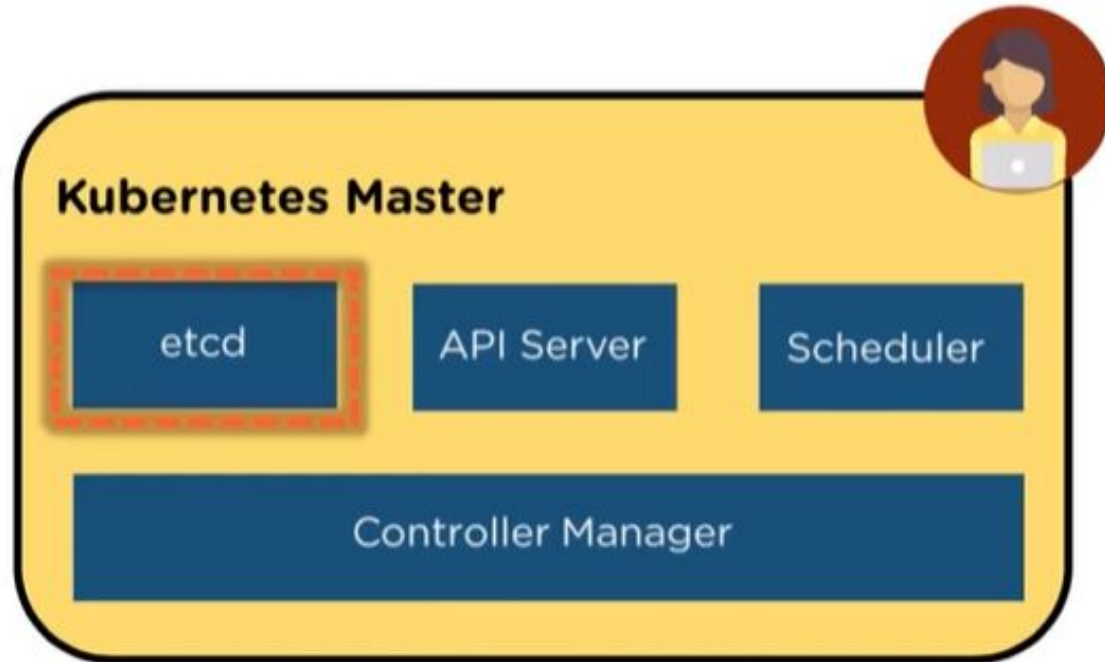




# Kubernetes Cluster Architecture



# Kubernetes Master Components

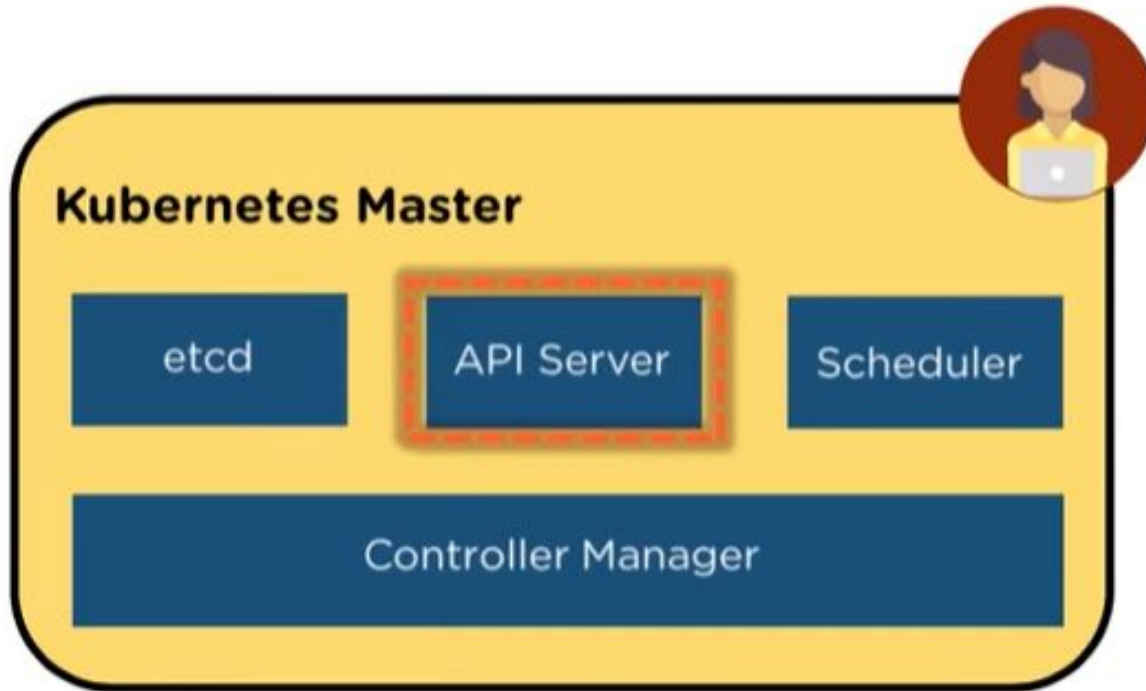


**Note** - A node is a working machine in Kubernetes; it can be a VM, cloud, and so on

## etcd

- Stores the configuration information required by nodes in the cluster
- Key-value is available and distributed across multiple nodes
- Accessible only by API Server as it may contain some sensitive information

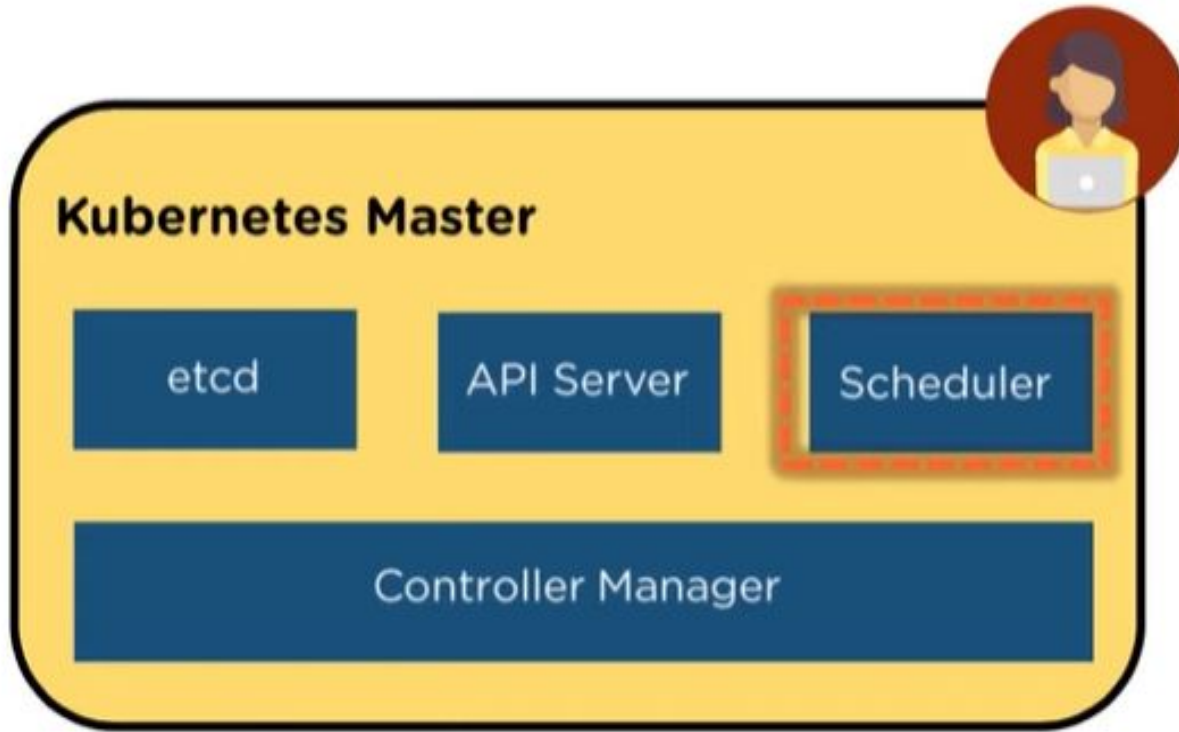
# Kubernetes Master Components



## API Server

- It is a means to provide operations on the cluster
- It implements an interface so that different tools and libraries can communicate effectively
- This component interacts with the worker nodes and gives them the required information

# Kubernetes Master Components

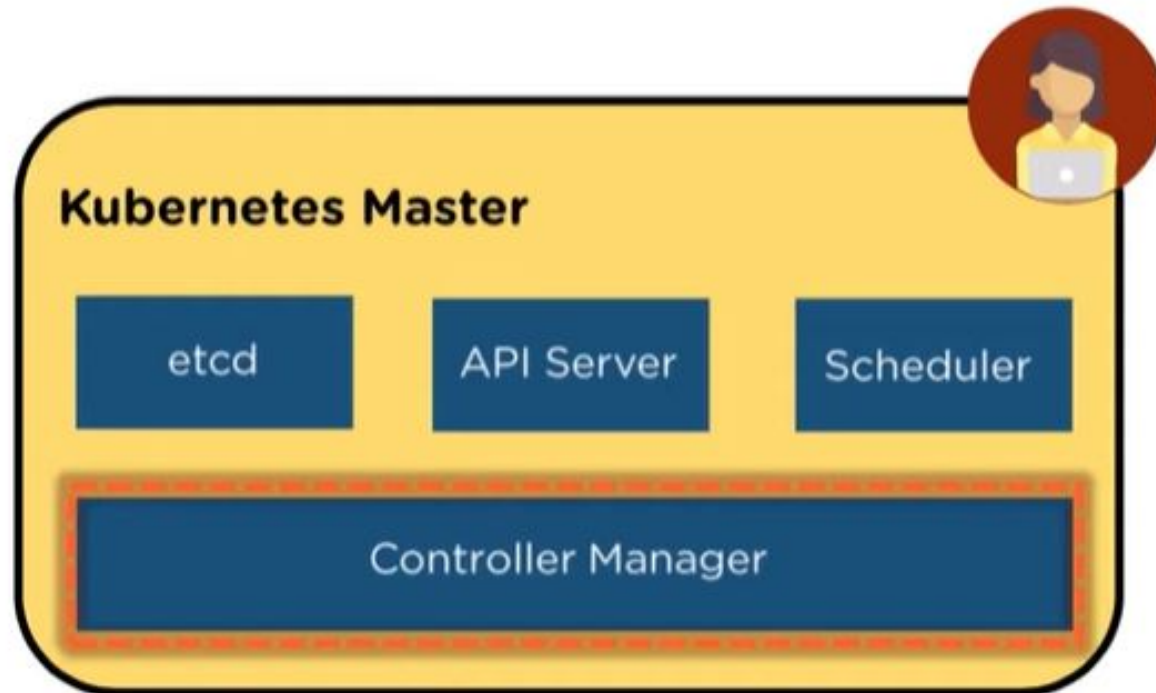


## Scheduler

- It is a key component of Kubernetes Master
- It is a service that is responsible for the dispersing workload in the master node
- It also tracks the utilization of workload on cluster nodes and then places the workload onto the resources that are available



# Kubernetes Master Components



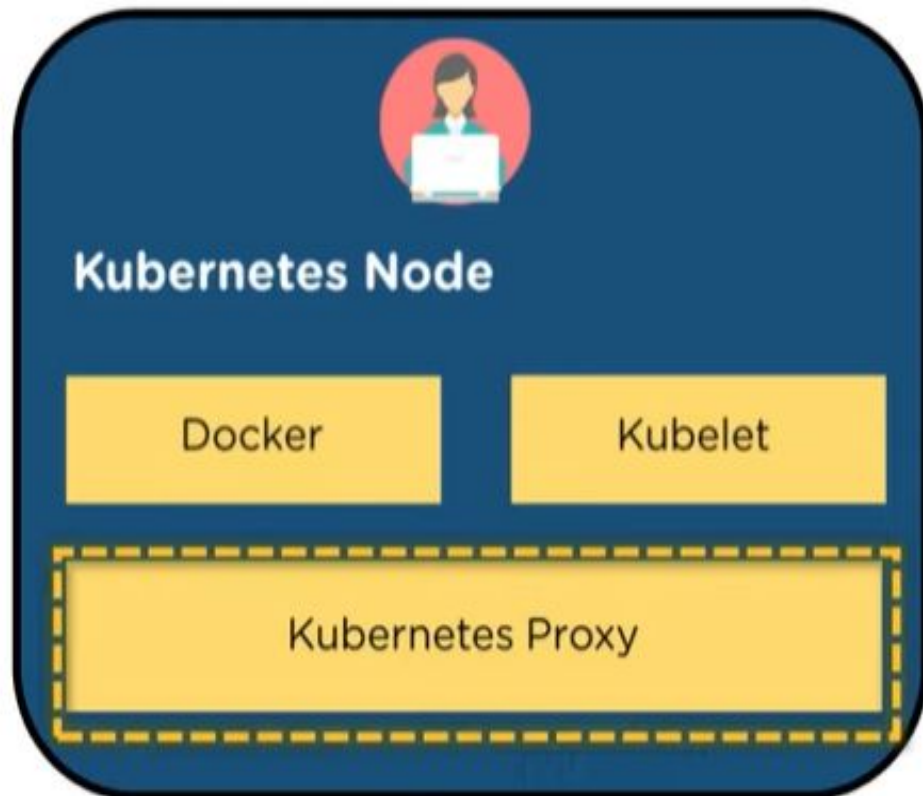
## Controller Manager

- It is a daemon (server) that runs in a continuous loop and is responsible for gathering information and sending it to the API Server
- This component is responsible for changing the current state of cluster to its desired state



# Kubernetes Worker Components

---



## Kubernetes Proxy

- It is a proxy service that runs on every node and helps in making services available to the external host. It helps in forwarding request to assigned containers
- It performs primitive load balancing. It manages pods on nodes, volumes, secrets, creation of new containers, and health check-ups

# Kubernetes Cluster Structure (RECAP)

---

## Kubernetes Master

### Kube- apiServer

Exposes Kubernetes API

### etcd

Distributed key value accessible to all

### Controller Manager

Multiple kinds of controllers to handle nodes

### Scheduler

Workload utilization and allocation of pods

## Kubernetes Node

### Kubelet Service

Manages pods on nodes, volume, secrets, creating new containers, etc.

### Kube Proxy Service

Manages networking parts for nodes

### (3) What is Kubernetes? - YouTube