What is Kubernetes?

Kubernetes is used to manage applications running in containers, which are small, isolated environments. It simplifies the reliable management of numerous apps and services, even when they are distributed across multiple servers. Kubernetes automates things like: Starting new apps when needed.

What is a Kubernetes Cluster?

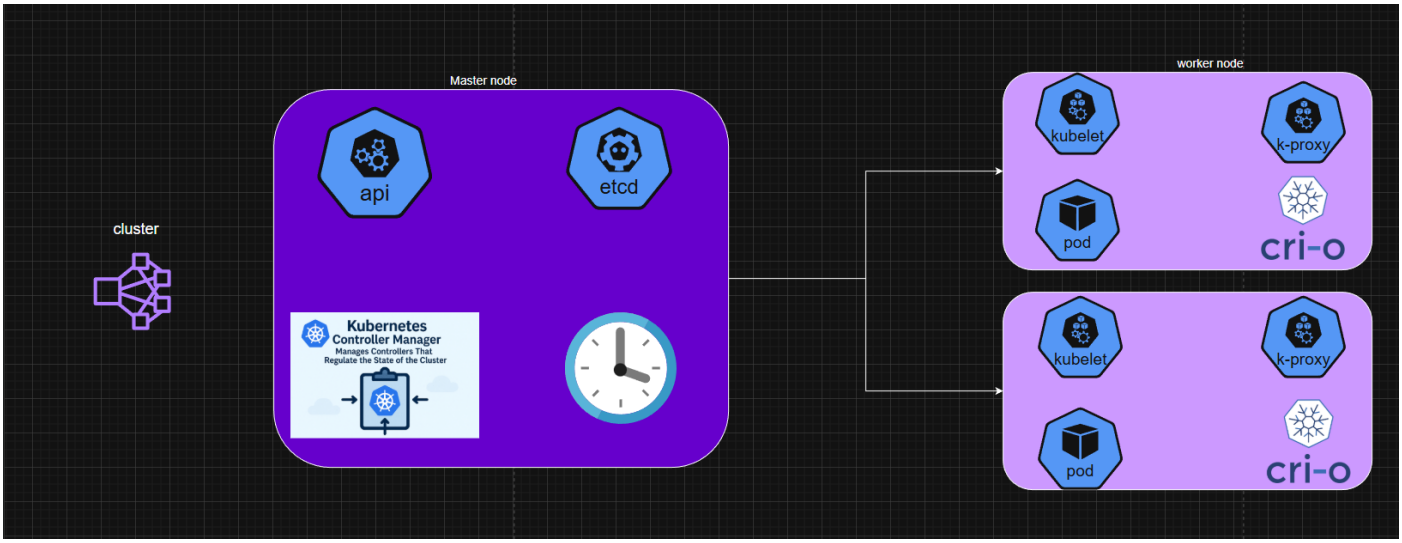
A **Cluster** is the entire ecosystem. It's a collection of **nodes** that work together as a single unit.

In the old days, if one server went down, your app went down. In a cluster, if one machine fails, the others just pick up the slack. It's basically a safety net that ensures your website or app stays live even when hardware decides to quit.

Master Node:

It doesn't actually do the "heavy lifting" of running app's code; instead, it decides *where* and *how* things should run.

- **API Server:** Every command you send goes through here.
- **Scheduler:** It looks at your available resources and decides which worker node is healthy enough to take on a new task.
- **Etcd:** The memory. It's a tiny database that stores the "truth" about the cluster—what's running, what's broken, and what the settings are.
- **Controller Manager:** If you say "I want 3 copies of this app running," and one crashes, the controller notices and starts a new one to get back to 3.



Worker Nodes:

The **Worker Nodes** are where the actual work happens. These are the machines that host your applications. If the Master Node is the architect with the blueprints, the Worker Nodes are the construction crew on the ground.

Key components in Worker Node:

- **Kubelet:** It's a tiny agent that talks back to the Master Node to make sure the containers are running as they should be.
- **Kube-proxy:** The operator. It handles the networking, making sure traffic gets to the right container.
- **Container Runtime:** The engine (usually Docker or container) that actually pulls the image and runs the software.