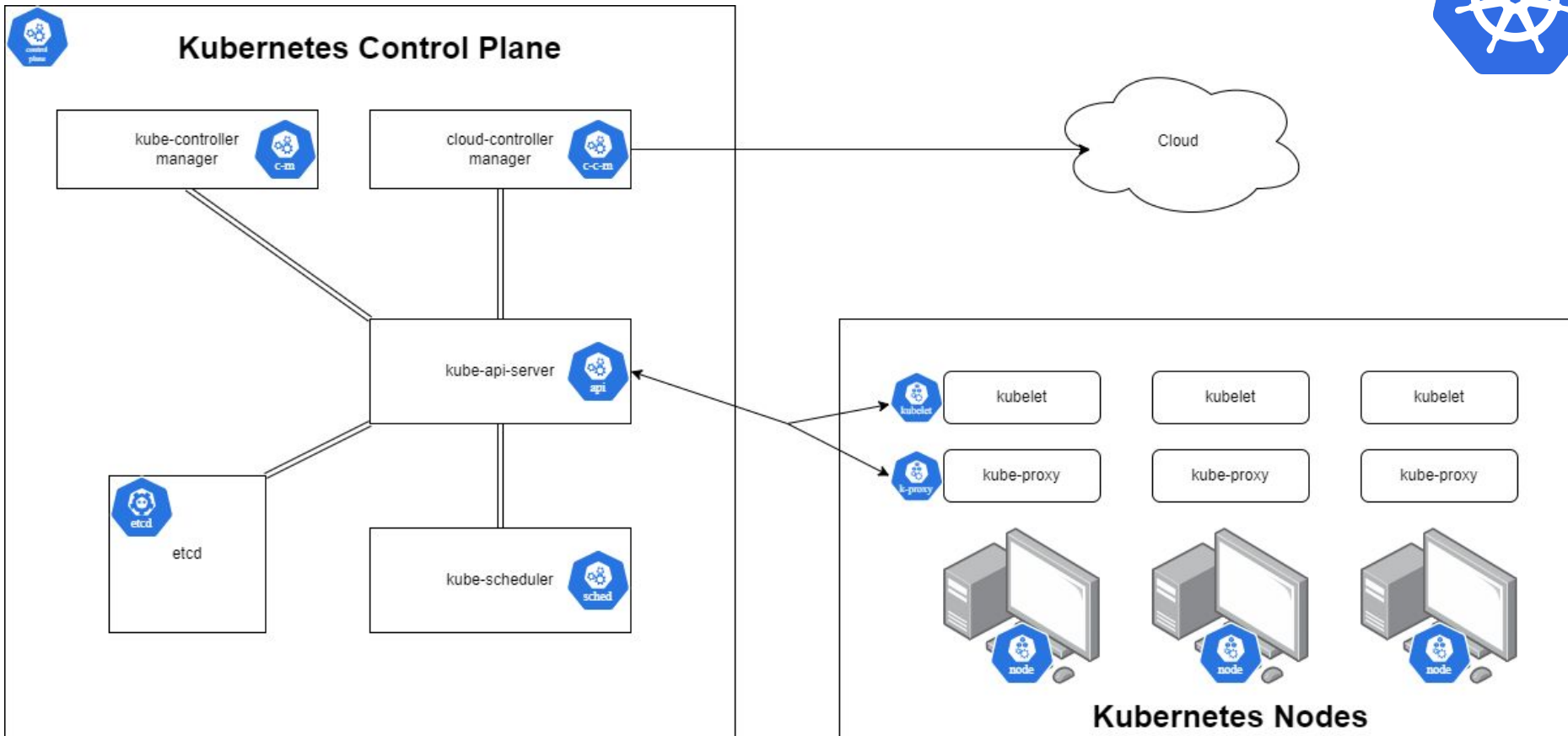




Kubernetes Architecture





Kubernetes API Server

- Exposes the Kubernetes API.
- All communication in the cluster utilizes this API.
- It accepts commands to view or change the state of the cluster.





etcd

- Highly available key-value store.
- Contains all the cluster data.
- Source of truth for the state in a Kubernetes cluster.
- It stores the deployment configuration of application, when any application is deployed.





Kubernetes scheduler

- Assigns newly created Pods to nodes.
- Determines where workloads should run.





Kubernetes controller manager

- Runs all the controller processes.
 - Controllers monitor the cluster state.
 - Controllers ensure the actual state matches the desired state.





Cloud controller manager

- Runs controllers that interact with underlying cloud providers.
- Links clusters into a cloud provider's API.





Nodes

- The worker machines in Kubernetes.
- It can be virtual or physical machine.
- It is managed by the control plane.
- It contains the services necessary to run applications.





Kubelet

- Communicates with the API server.
- Ensure that Pods and their associated containers are running.
- Reports to the control plane on health and status.





Container runtime

- Downloads images and runs containers.
- Kubernetes implements an interface so that this component is pluggable.
- Docker is well-known runtime.



Controllers

- Monitor the state of a cluster.
- Take action to ensure the actual state matches the desired state.
- Communicate with the API server to initiate these actions.
- Track Kubernetes objects and ensure that the desired state is achieved.



Kubernetes Objects

- Persistent entities in Kubernetes.
- Define the desired state for your workload.
- Use the Kubernetes API to work with them -
 - kubectl CLI
 - Client libraries
- Consist of two parts: spec and status





Namespaces and names

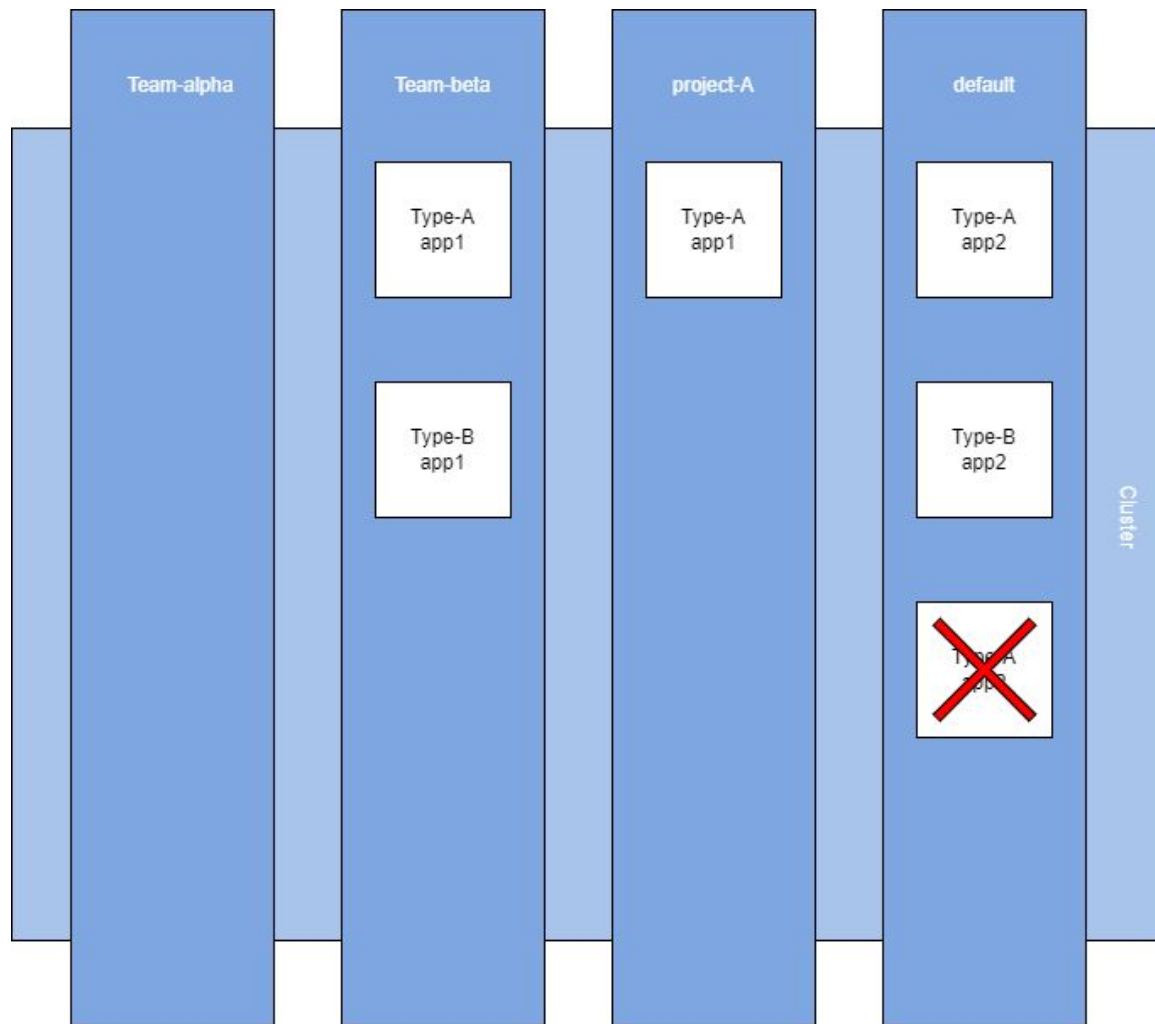
Namespaces

- **Virtualization of a physical cluster** - With namespaces, you can make one cluster appear to be several distinct clusters.
- Segregate cluster by team, project, etc.
- Necessary with large numbers of users.

Names

- Each object has a name.
- Names are unique for a resource type within a namespace.







Labels and selectors

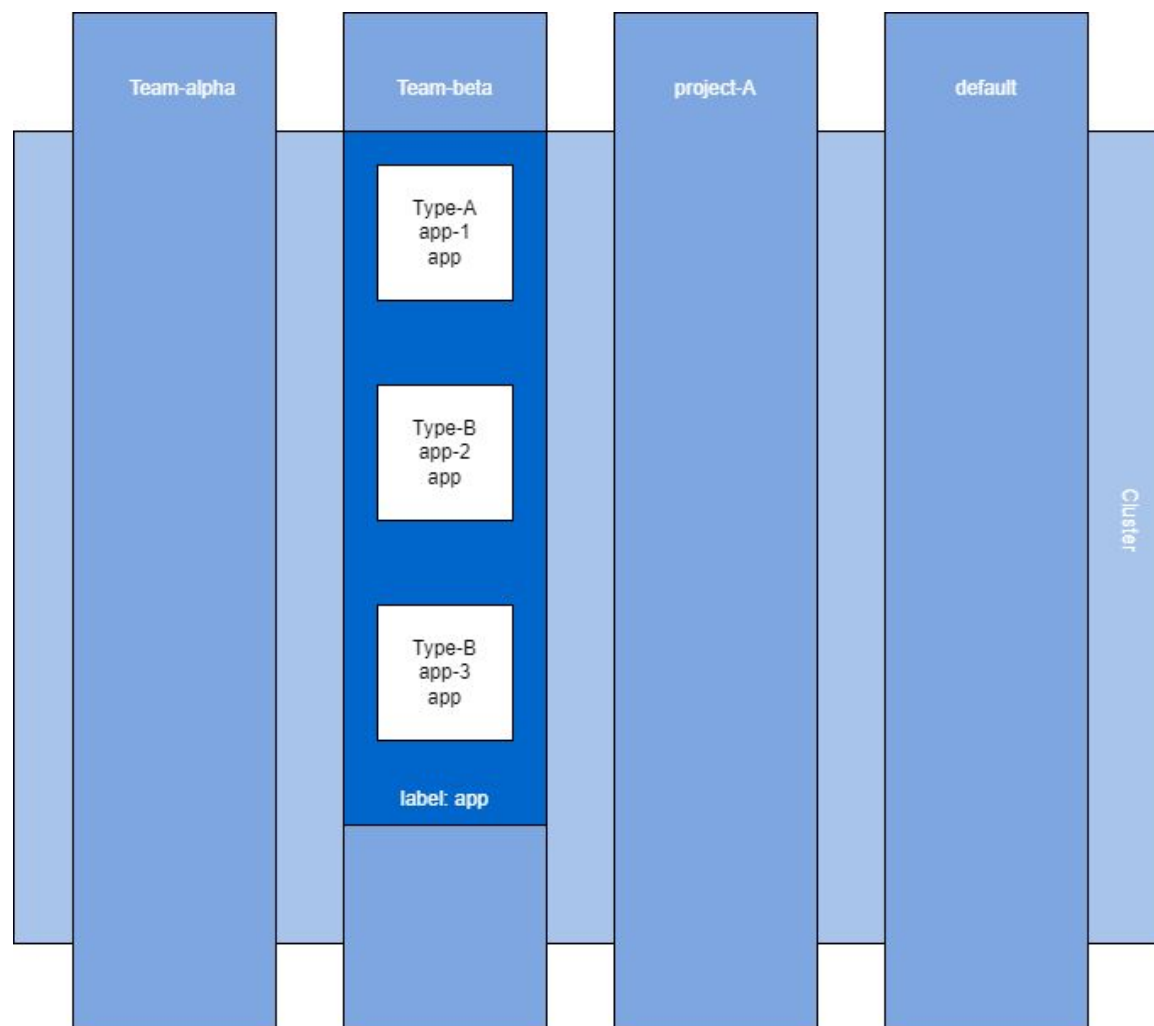
Labels

- Key/value pairs attached to objects.
- Intended for identification of objects.
- Not unique.
- Organize and group objects.

Selectors

- Identify and group a set of objects.



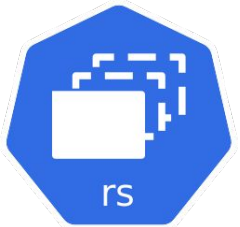




Pod

- Simplest unit in Kubernetes.
- Represents processes running in your cluster.
- Encapsulates a container (or sometimes multiple).
- Replicating a Pod serves to scale an application horizontally.





ReplicaSet

- Maintains a set of identical Pods.
- Definition consists of:
 - Number of replicas
 - Pod template
 - Selector to identify which Pods it can acquire.
- Generally encapsulated by a Deployment.





Deployment

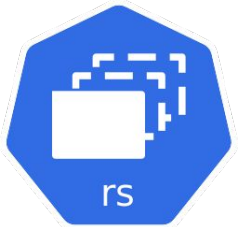
- Provides updates for Pods and ReplicaSets.
- Runs multiple replicas of your application.
- Suitable for stateless applications.
- Update triggers a rollout.



Manage Kubernetes applications

- ReplicaSet
- Autoscaling
- Rolling updates
- ConfigMaps and Secrets
- Service binding





ReplicaSet

- Maintains a set of identical Pods.
- Definition consists of:
 - Number of replicas
 - Pod template
 - Selector to identify which Pods it can acquire.
- Generally encapsulated by a Deployment.





Autoscaling

- ReplicaSet works with a set number of pods.
- Horizontal Pod Autoscaler (HPA) enables scaling up and down as needed.
- Can configure based on desired state of CPU, Memory, etc.





Rolling updates

- ReplicaSet and autoscaling are used to minimize the downtime and service interruptions.
- It is a way to roll out app changes in an automated and controlled fashion throughout your pods.
- Work with pod templates such as deployments.
- Allow for rollback if something goes wrong.





ConfigMaps & Secrets

- Used to provide configuration for deployments.
- Reusable across deployments.
- Created in a couple of different ways -
 - Using string literals.
 - Using an existing properties or “Key”=“value” file.
 - Providing a ConfigMap YAML descriptor file.
- Multiple ways to reference from pod/deployment -
 - Reference as an environment variable.
 - Mount as volume.

