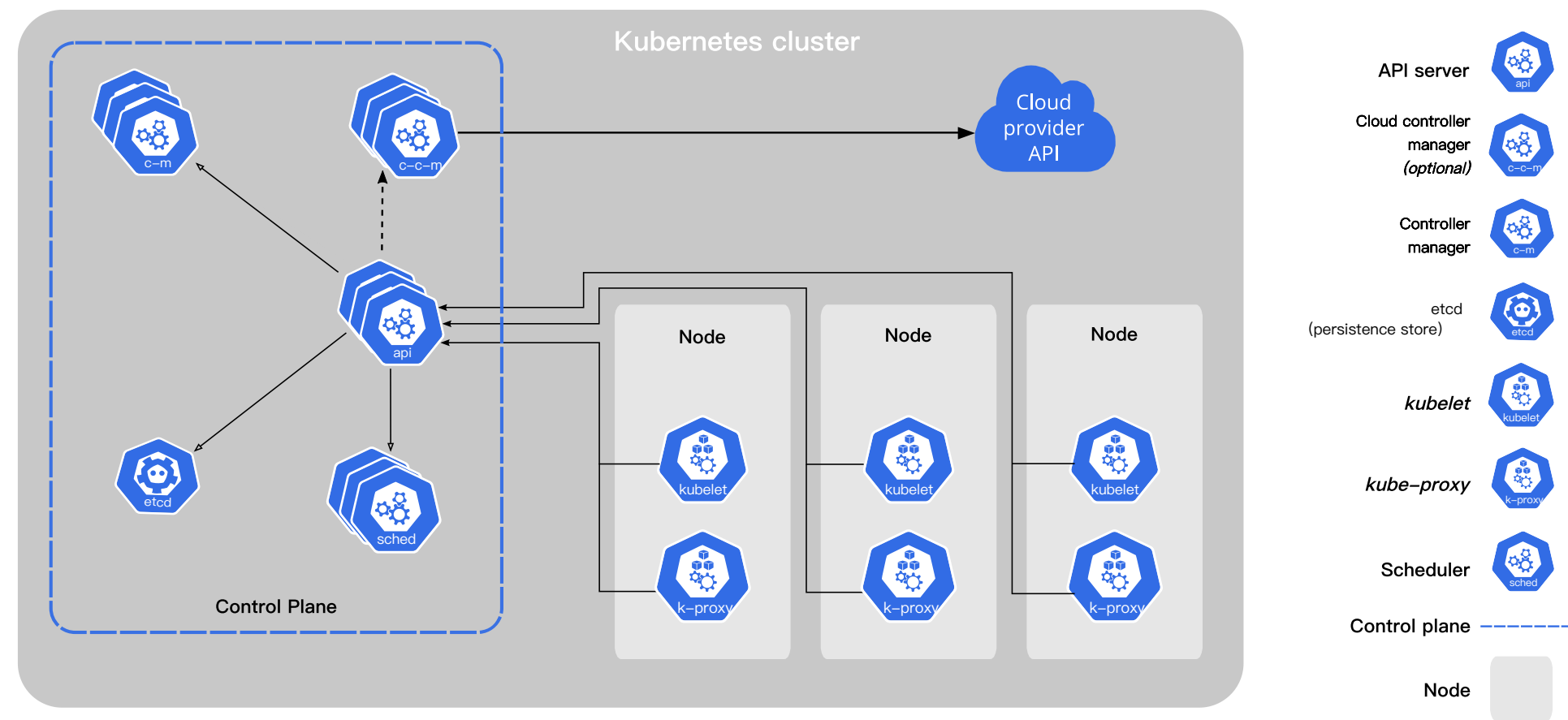


Kubernetes Components

An overview of the key components that make up a Kubernetes cluster.

This page provides a high-level overview of the essential components that make up a Kubernetes cluster.



The components of a Kubernetes cluster

Core Components

A Kubernetes cluster consists of a control plane and one or more worker nodes. Here's a brief overview of the main components:

Control Plane Components

Manage the overall state of the cluster:

[kube-apiserver](#)

The core component server that exposes the Kubernetes HTTP API.

[etcd](#)

Consistent and highly-available key value store for all API server data.

[kube-scheduler](#)

Looks for Pods not yet bound to a node, and assigns each Pod to a suitable node.

[kube-controller-manager](#)

Runs controllers to implement Kubernetes API behavior.

[cloud-controller-manager](#) (optional)

Integrates with underlying cloud provider(s).

Node Components

Run on every node, maintaining running pods and providing the Kubernetes runtime environment:

[kubelet](#)

Ensures that Pods are running, including their containers.

[kube-proxy](#) (optional)

Maintains network rules on nodes to implement Services.

[Container runtime](#)

Software responsible for running containers. Read [Container Runtimes](#) to learn more.

This item links to a third party project or product that is not part of Kubernetes itself. [More information](#)

Your cluster may require additional software on each node; for example, you might also run [systemd](#) on a Linux node to supervise local components.

Addons

Addons extend the functionality of Kubernetes. A few important examples include:

[DNS](#)

For cluster-wide DNS resolution.

[Web UI](#) (Dashboard)

For cluster management via a web interface.

[Container Resource Monitoring](#)

For collecting and storing container metrics.

[Cluster-level Logging](#)

For saving container logs to a central log store.

Flexibility in Architecture

Kubernetes allows for flexibility in how these components are deployed and managed. The architecture can be adapted to various needs, from small development environments to large-scale production deployments.

For more detailed information about each component and various ways to configure your cluster architecture, see the [Cluster Architecture](#) page.

Items on this page refer to third party products or projects that provide functionality required by Kubernetes. The Kubernetes project authors aren't responsible for those third-party products or projects. See the [CNCF website guidelines](#) for more details.

You should read the [content guide](#) before proposing a change that adds an extra third-party link.

Feedback

Was this page helpful?

Yes

No