**Content**

[1. Core concepts 2](#_Toc80040997)

[1. Pods vs Nodes 3](#_Toc80040998)

1. Core concepts

* Kubeadm is a tool built to provide kubeadm init and kubeadm join as best-practice "fast paths" for creating Kubernetes clusters.
* minikube is local Kubernetes, focusing on making it easy to learn and develop for Kubernetes.
* etdc – key value database
* kube-api server – kubectl utility is reaching to this server
* controller – a proces, which continuously monitors the state of various componens within the system. Works towards brining the whole system towards the desired functioning state. Intelligence is implemented through the various controllers.
  + node controller
  + replication controller
  + deployment c.
  + job c.
  + ....
* kube controller manager – the controllers are packaget into this one. When you install this, the different controllers get installed as well.
* kube-scheduler – responsible for scheduling pods on nodes. Only responsible for deciding which pod goes on which node. It does not place the pod on the nodes. Thats the job of the kubelet. Only decides, which pod goes where.
  + filter nodes
  + rank nodes
  + chooses the best place
* kubelet – captain on the ship, who creates the pod the ship.
  + register node
  + create PODS
  + monitor NODE & PODS, report
* kube proxy – within a kubernetes cluster, every pod can reach every other pod. This is accomplished by deploying a pod networking solution to the cluster. There are many solutions.
  + services in kubernetes (they are in memory, not a pod. They are accessible)
* control plane - The control plane manages the worker nodes and the Pods in the cluster. In production environments, the control plane usually runs across multiple computers and a cluster usually runs multiple nodes, providing fault-tolerance and high availability.
* Pods & Nodes

A Pod always runs on a Node. A Node is a worker machine in Kubernetes and may be either a virtual or a physical machine, depending on the cluster. Each Node is managed by the control plane. Kubernetes does not deploy containers directly into worker nodes. They are encapsulated in kuberentes objects, called pods. Pod 1:1 containers. Pods however can have multi containers, it only makes sense if they are not the same kind.

