DATE : 18.04.2025

DT/NT : DT

LESSON : KUBERNETES

SUBJECT: VOLUMES

BATCH : B 303

AWS-DEVOPS

TECHPRO EDUCATION

techproeducation.com +1 (585) 304 29 59

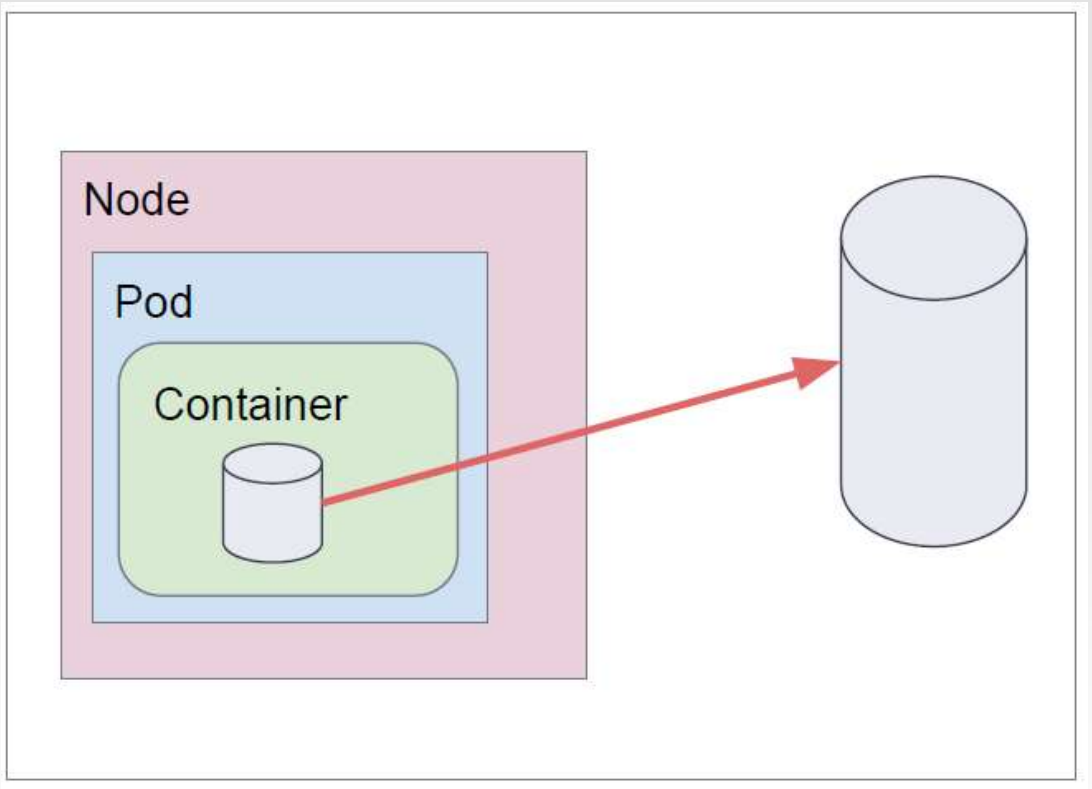
# Table of Contents

# Volumes

- on-disk files in a Container are ephemeral.
- All data stored inside a container is deleted if the container crashes.
- When a Container crashes, kubelet will restart it, but the files will be lost which means that it will not have any of the old data.
- To overcome this problem, Kubernetes uses **Volumes**. A Volume is essentially a directory backed by a storage medium. The storage medium, content and access mode are determined by the Volume Type.

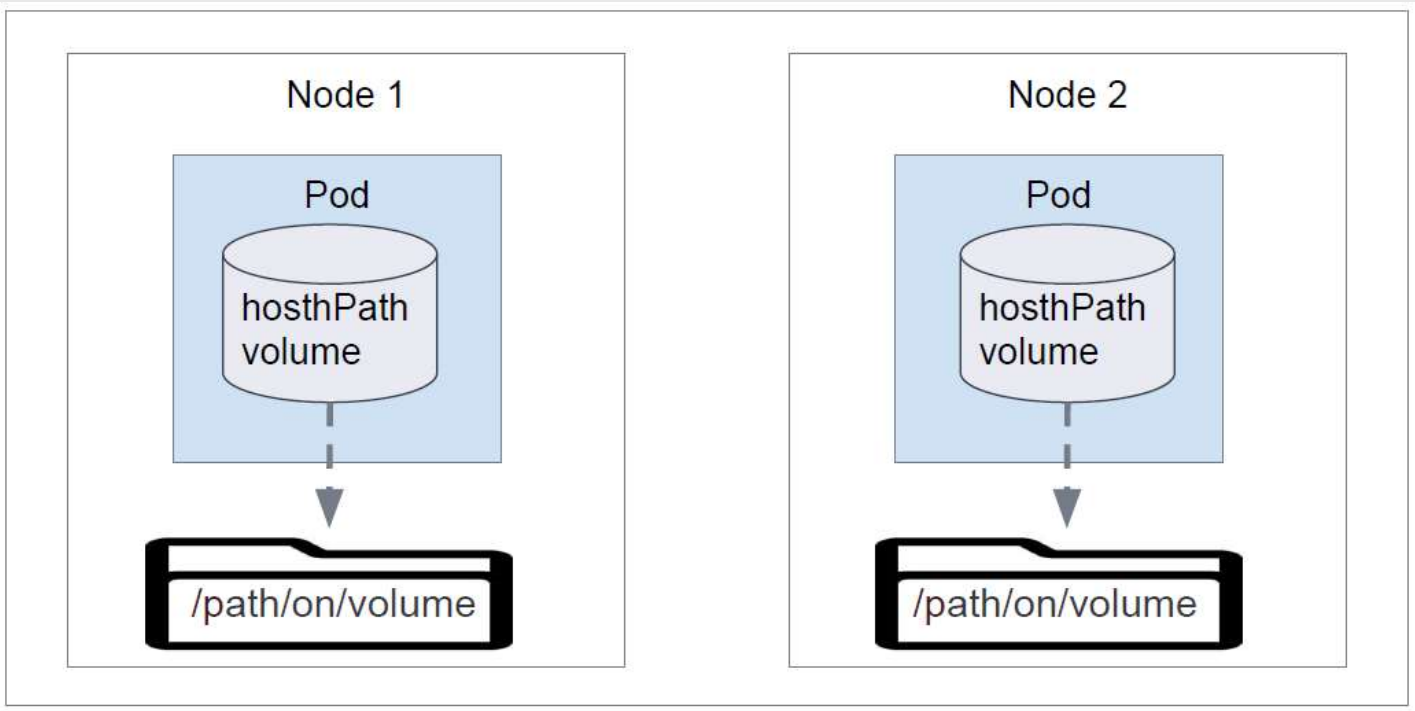
# Volumes

A **volume** can be thought of as a directory which is accessible to the containers in a pod.



# Volume Types

**hostPath:** A hostPath volume mounts a file or directory from the host node's filesystem into your Pod.



# Volume Types

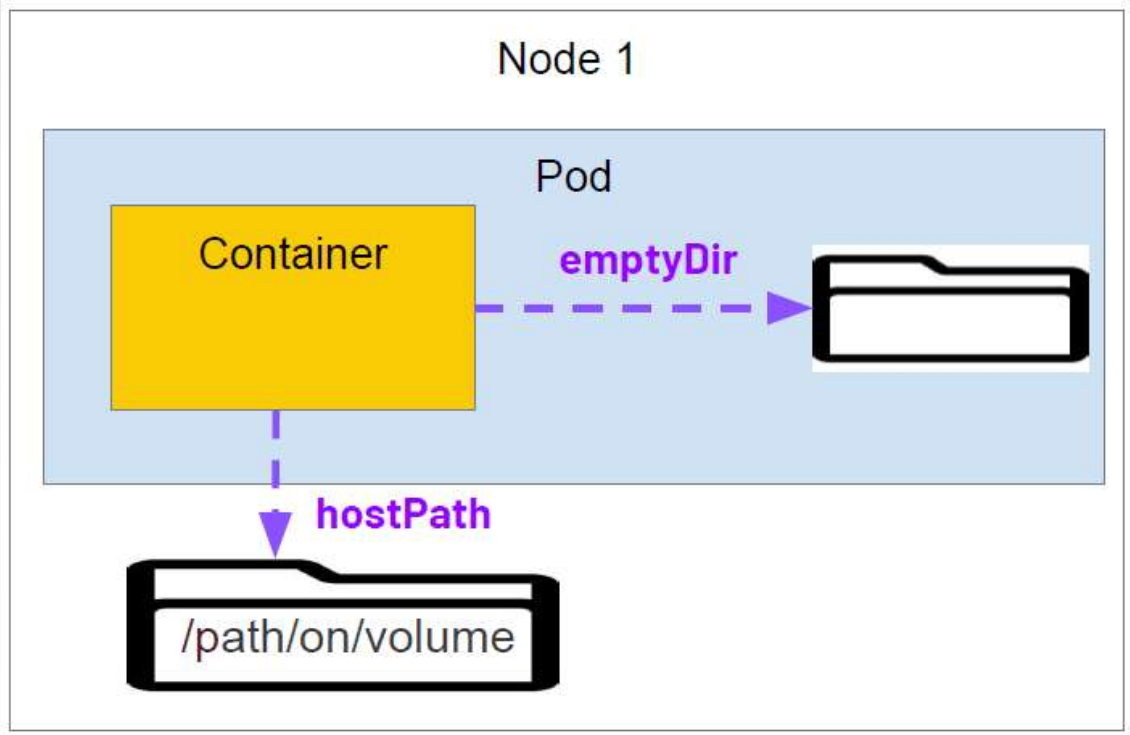
- **emptyDir:** An emptyDir volume is first created when a Pod is assigned to a Node and exists as long as that Pod is running on that node.

Some uses for an emptyDir are:

- checkpointing a long computation for recovery from crashes
- as a cache

# Volume Types

- **hostPath vs emptyDir**



# Volume Types

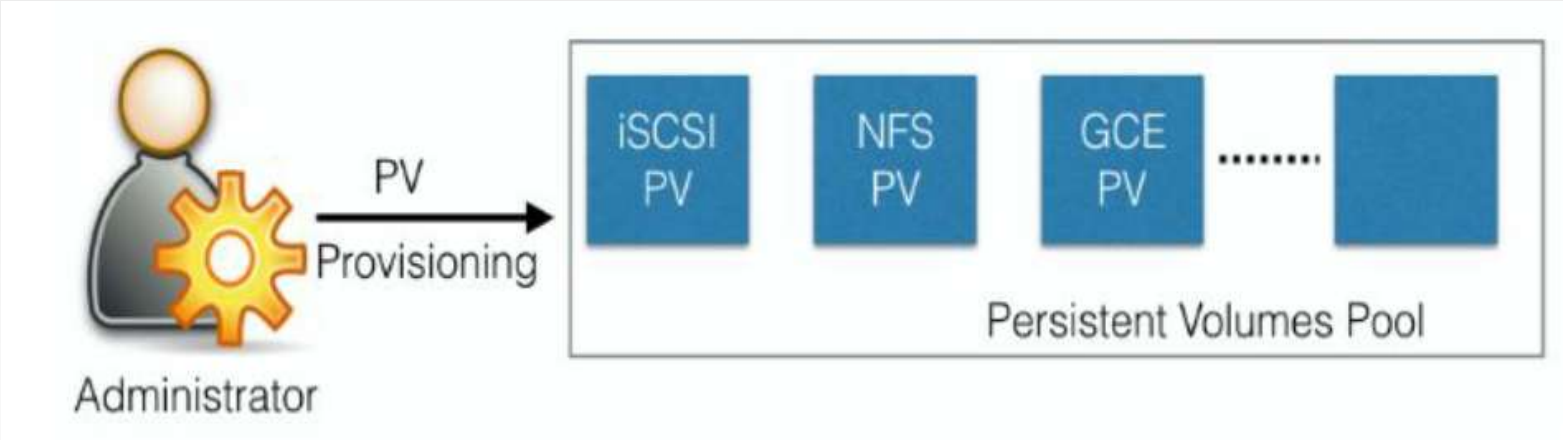
- **awsElasticBlockStore:** An awsElasticBlockStore volume mounts an Amazon Web Services (AWS) EBS Volume into your Pod.
- **azureDisk:** An azureDisk is used to mount a Microsoft Azure Data  Disk into a Pod.
- **gcePersistentDisk:** A gcePersistentDisk volume mounts a Google  Compute Engine (GCE) persistent disk (PD) into your Pod.

# Volume Types

- **Secret:** A secret volume is used to pass sensitive information, such as passwords, to Pods.
- **configMap:** The configMap resource provides a way to inject configuration data, or shell commands and arguments into a Pod.
- **persistentVolumeClaim:** A persistentVolumeClaim volume is used to mount a PersistentVolume into a Pod.

# PersistentVolumes

A **PersistentVolume (PV)** is a piece of storage in the cluster that has been provisioned by an administrator or dynamically provisioned using Storage Classes.



# PersistentVolumeClaims

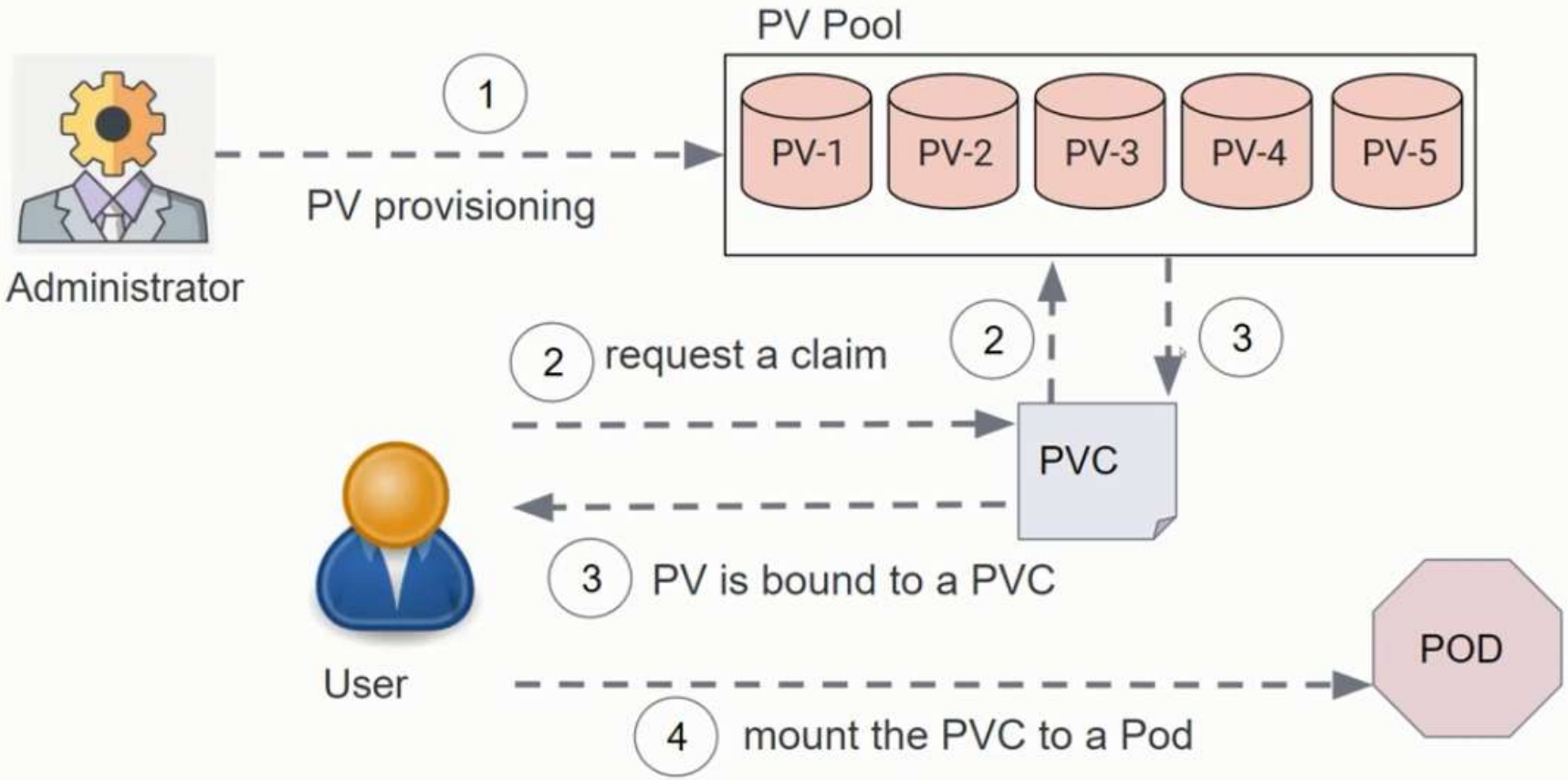
A **PersistentVolumeClaim (PVC)** is a request for storage by a user. Users request for PersistentVolume resources based on type, access mode, and size. There are four access modes:

- **ReadWriteOnce** (read-write by a single node)
- **ReadOnlyMany** (read-only by many nodes)
- **ReadWriteMany** (read-write by many nodes).
- **ReadWriteOncePod** (read-write only one pod in the cluster)

Once a suitable **PersistentVolume** is found, it is bound to a

**PersistentVolumeClaim**.

# PersistentVolumeClaims



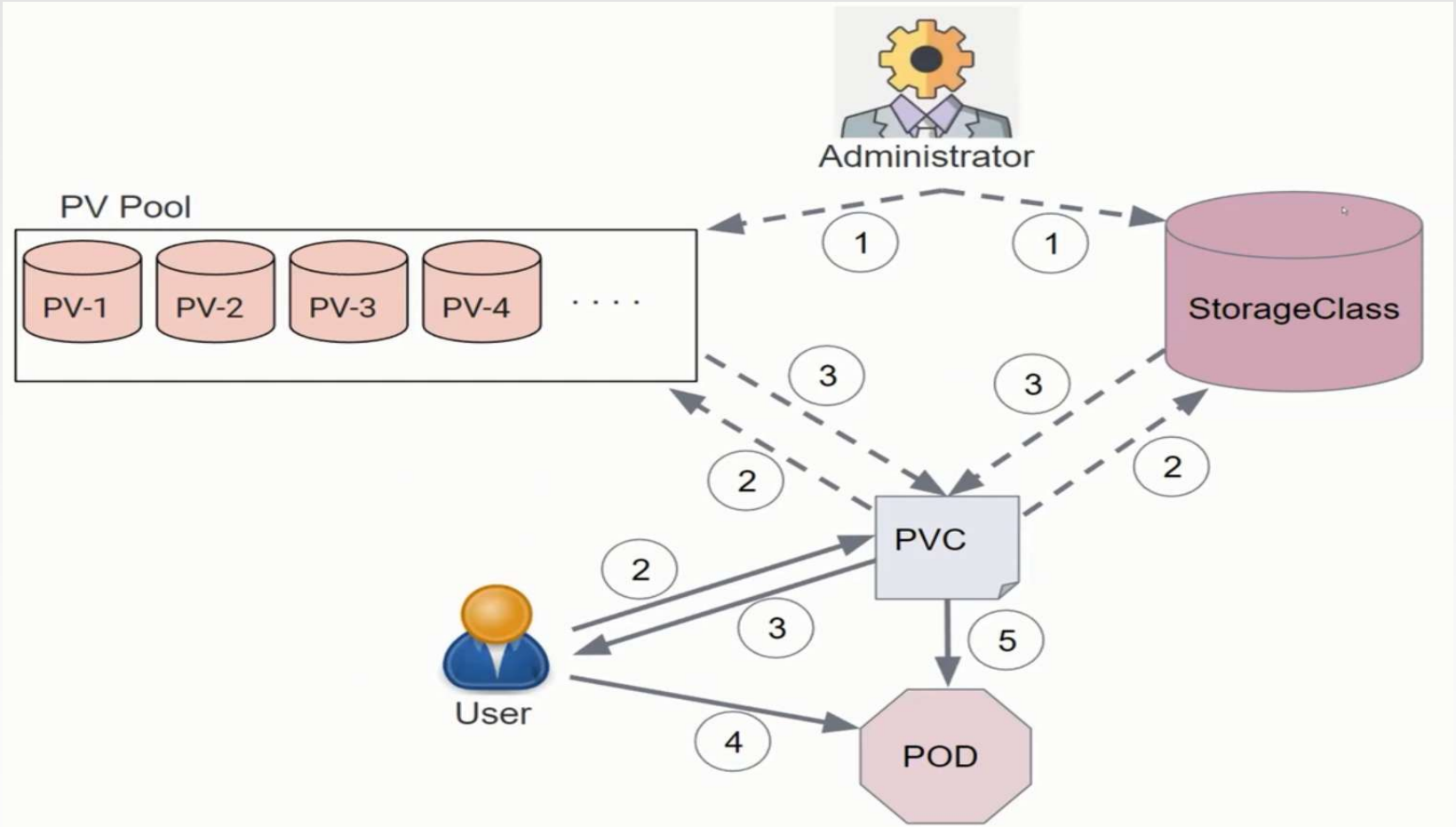
# The interaction between PVs and PVCs

**Static**

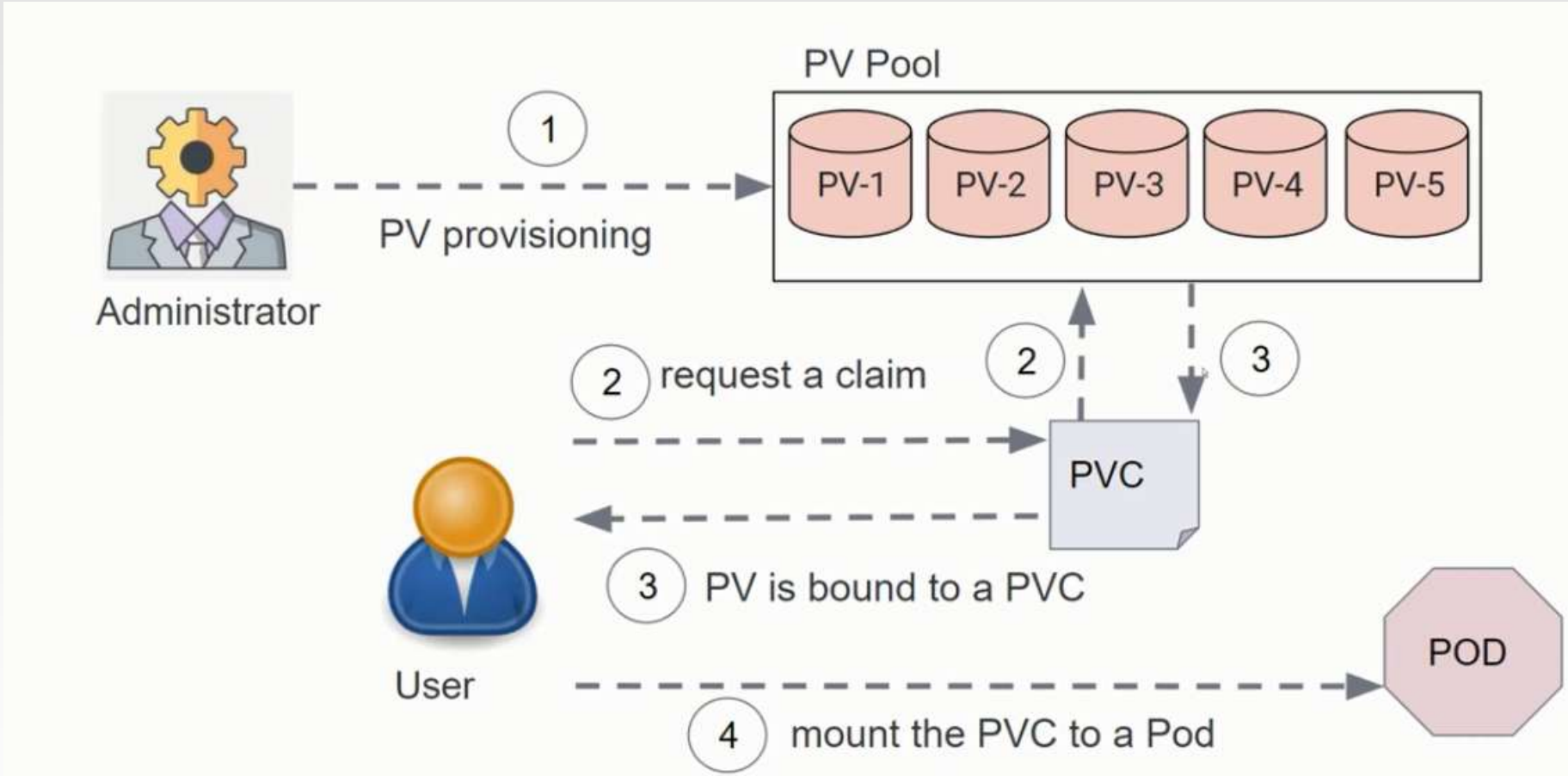
**Dynamic**

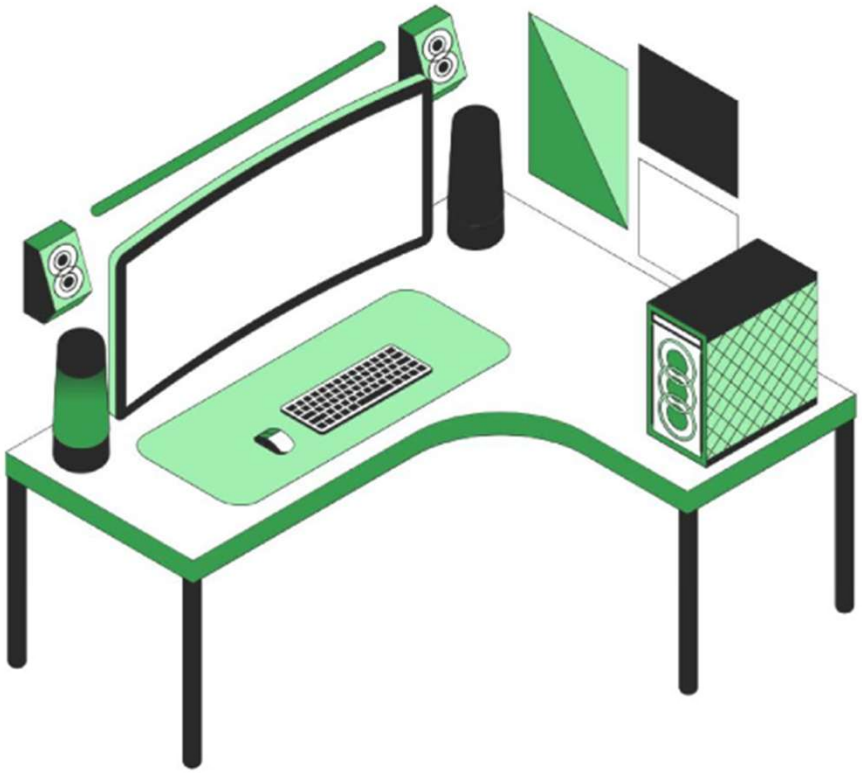
# Dynamic PV Provisioning

A **StorageClass** provides a way for administrators to describe the "classes" of storage they offer.



# Static PV Provisioning





# Do you have any questions?

Send it to us! We hope you learned something new.