ACADEMIND

Understanding “State”

State is data created and used by your application which must not be lost

User-generated data, user accounts, ...

Often stored in a database, but could also be files (e.g. uploads)

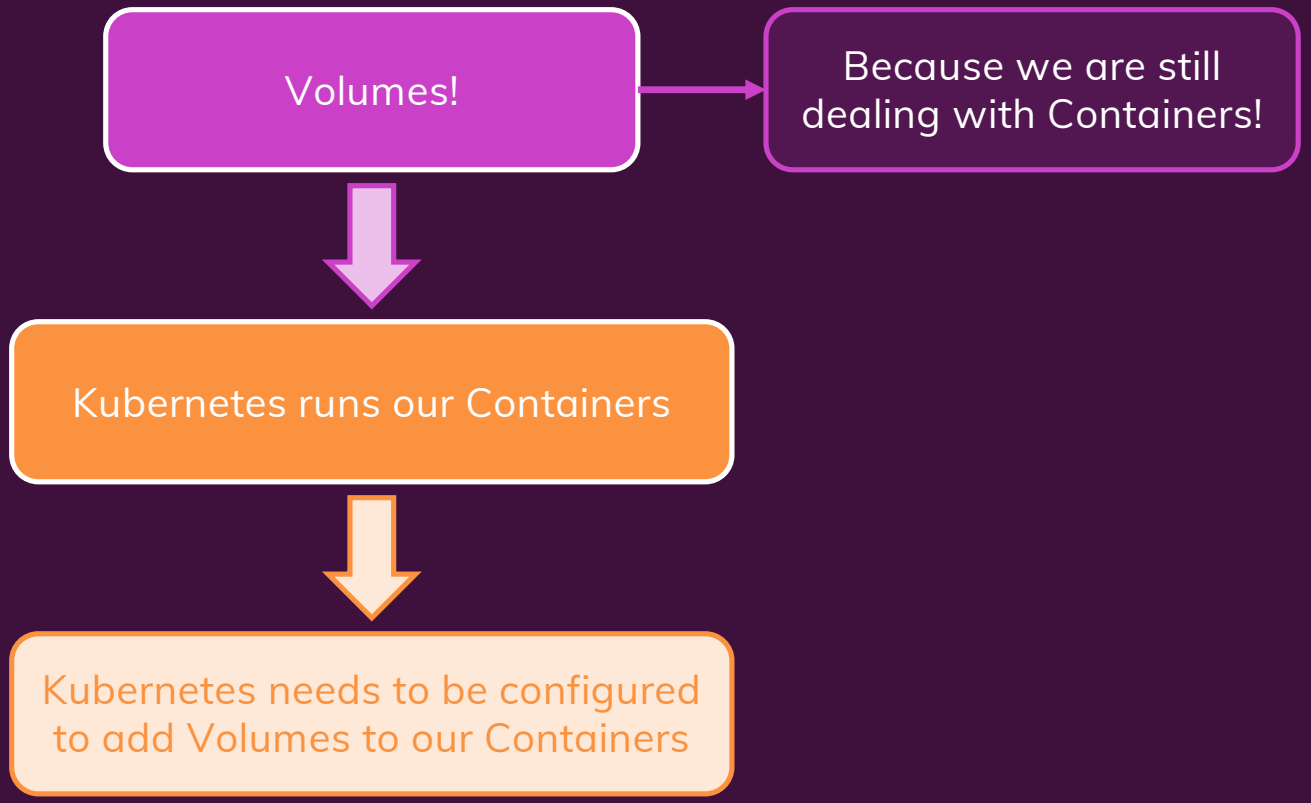
Intermediate results derived by the app

Often stored in memory, temporary database tables or files

Volumes

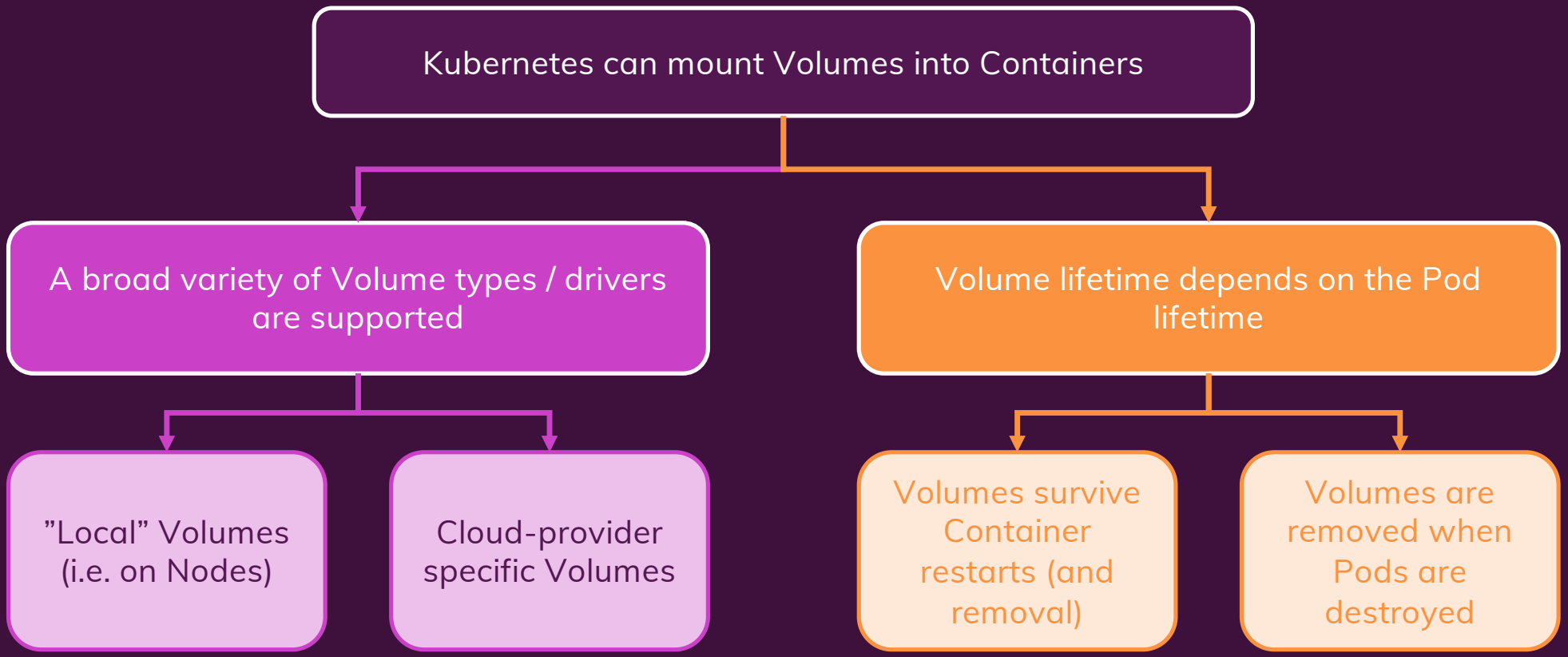
ACADEMIND

We Already Know The Solution!



ACADEMIND

Kubernetes & Volumes



ACADEMIND

Kubernetes Volumes vs Docker Volumes

Kubernetes Volumes	Docker Volumes
Supports many different Drivers and Types	Basically no Driver / Type Support
Volumes are not necessarily persistent	Volumes persist until manually cleared
Volumes survive Container restarts and removals	Volumes survive Container restarts and removals

ACADEMIND

Persistent Volumes

Volumes are destroyed when a Pod is removed

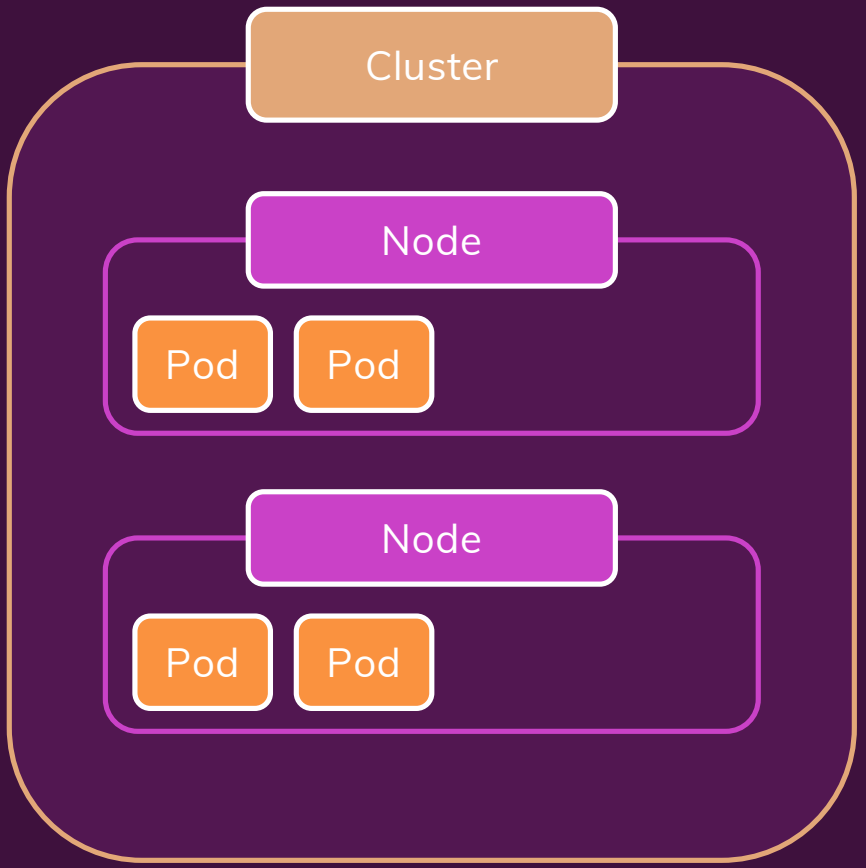
hostPath partially works around that in “One-Node” environments

Pod- and Node-independent Volumes are sometimes required

Persistent Volumes

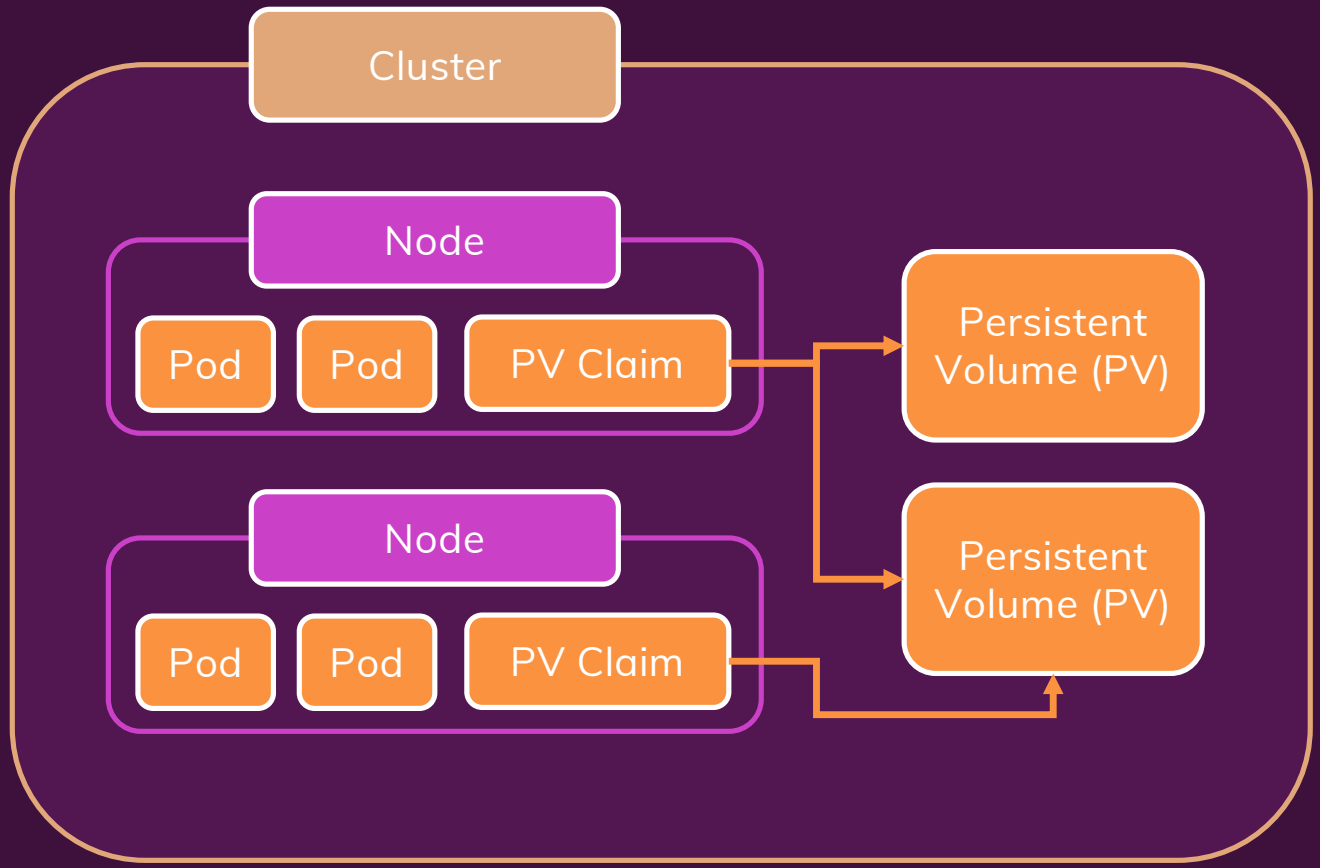
ACADEMIND

From Volumes To Persistent Volumes



ACADEMIND

Persistent Volumes & Persistent Volume Claims



ACADEMIND

“Normal” Volumes vs Persistent Volumes

Volumes allow you to persist data

”Normal” Volumes

- Volume is attached to Pod and Pod lifecycle
- Defined and created together with Pod
- Repetitive and hard to administer on a global level

Persistent Volumes

- Volume is a standalone Cluster resource (NOT attached to a Pod)
- Created standalone, claimed via a PVC
- Can be defined once and used multiple times