

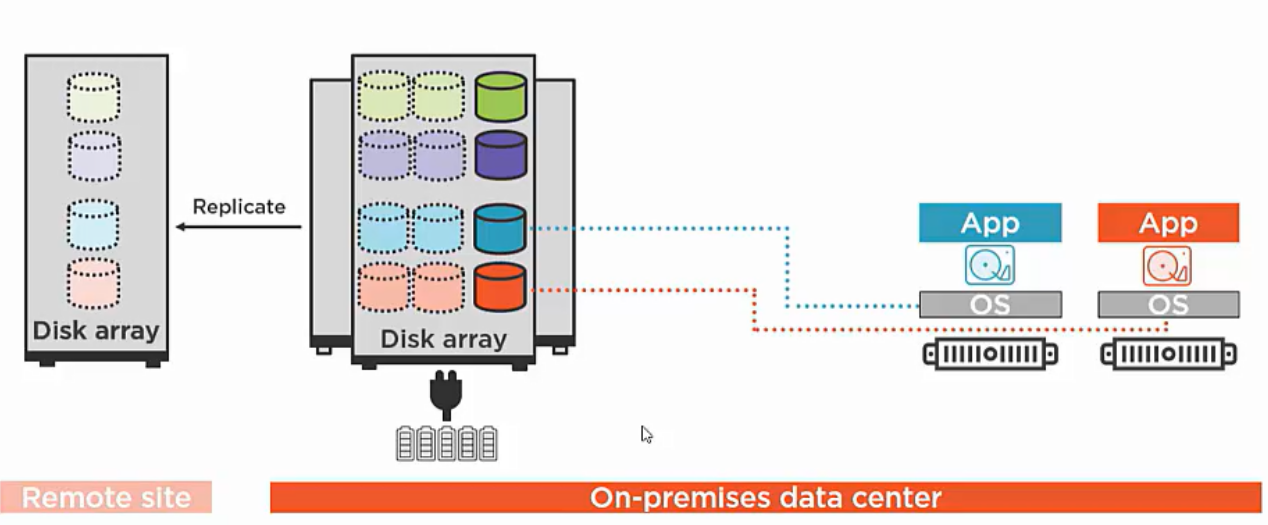


**STORAGE IN K8s**



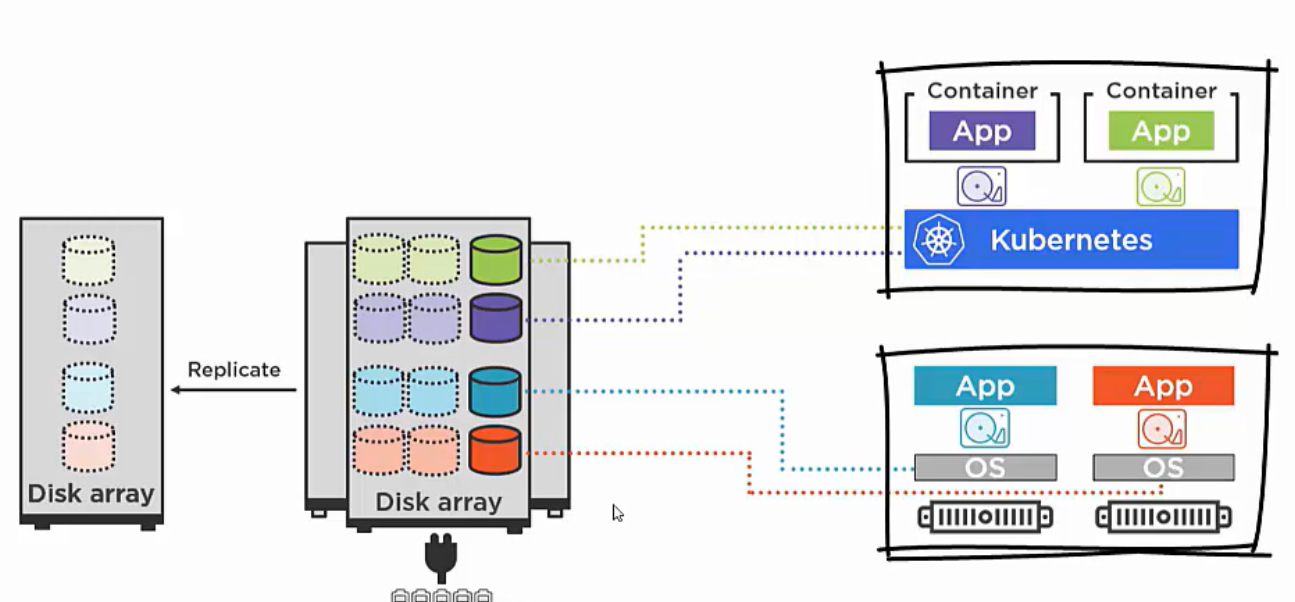
**How storage use to work in on-premise data center?**

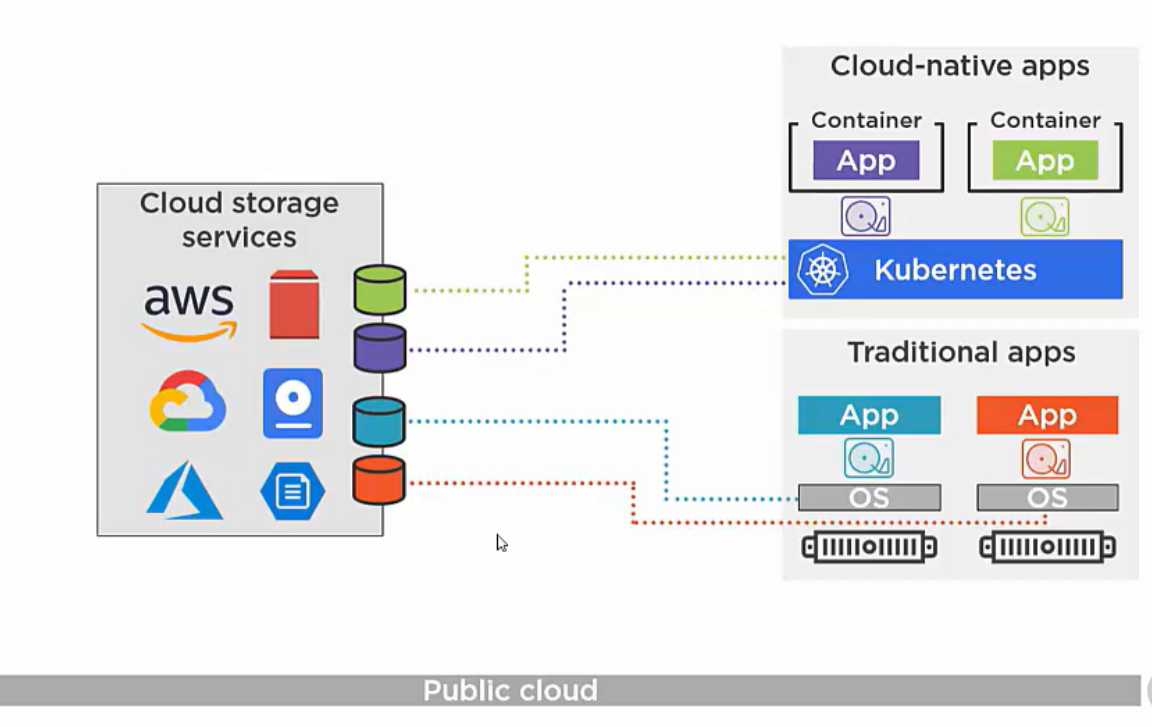
* We had some dedicated servers in which our apps were running.
* To store our application data we had some shared file system or some database.
* Behind these database or file system we had some dedicated storage hardware like EMC or netapp.
* It helps in keeping multi copy or replicate data.



**In K8s world how storage is happening?**

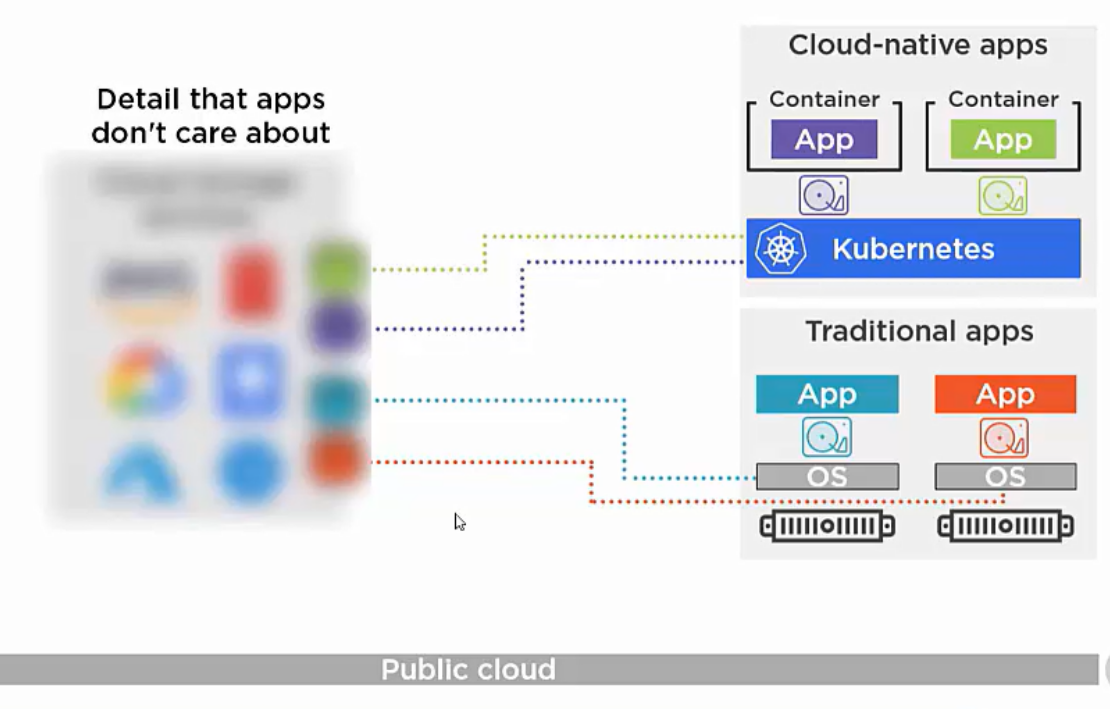
* K8s does not do any persistent storage.
* It is using some kind of external storage services either from cloud providers or on-premise.





Ex: Elastic block store by amazon.

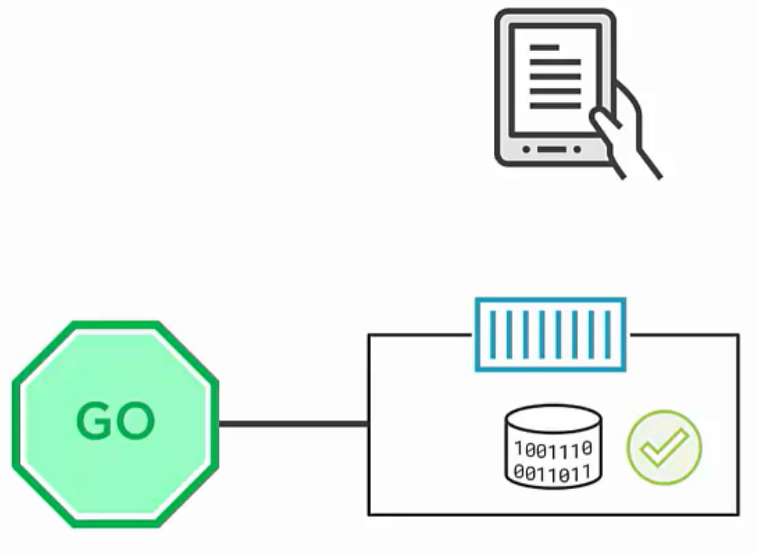
Developers doesn’t care about the how it is persisted.





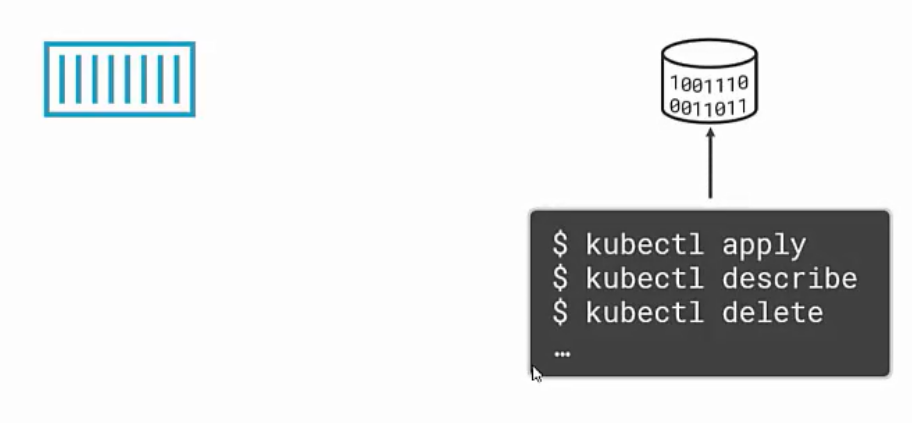


* Life cycle of non-persistent data is same as that of container.
* Once container is deleted the data is gone.



For stateful application we want our data to persist our data even if the container or apps crashes. So, we need to **decouple our data with our application**.

In container and K8s world the persistent storage, decoupled from application is the **volume**.



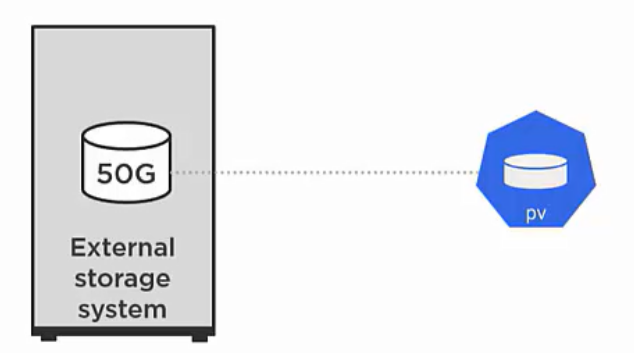
**VOLUMES are independent of container lifecycle.**

* If any apps need data we mount the volume to the container, but if the containers dies also our volume remains unchanged.

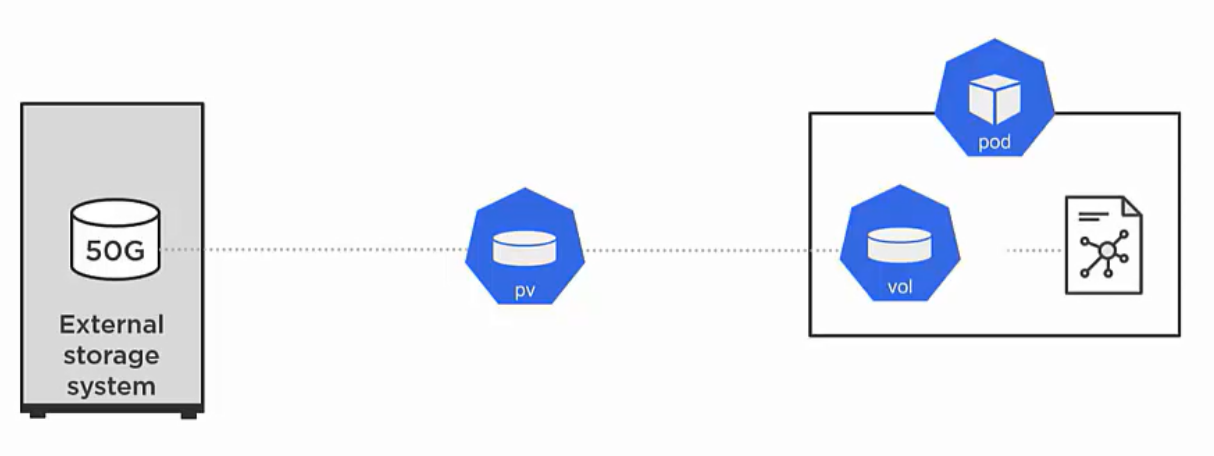


**How volume works in K8s?**

* Someone creates a volume in external storage.
* In K8s we create pv (persistent volume object) which links back to external volume.



* Once we link with pv the volume exists in its own right in K8s.
* We can claim the pv in our pods and then our apps can use it.
* If the pods gets deleted also volume is unchanged.





* Means lifecycle of pods and container are independent of volumes.
* Pods and container are not aware about how the volume is implemented.