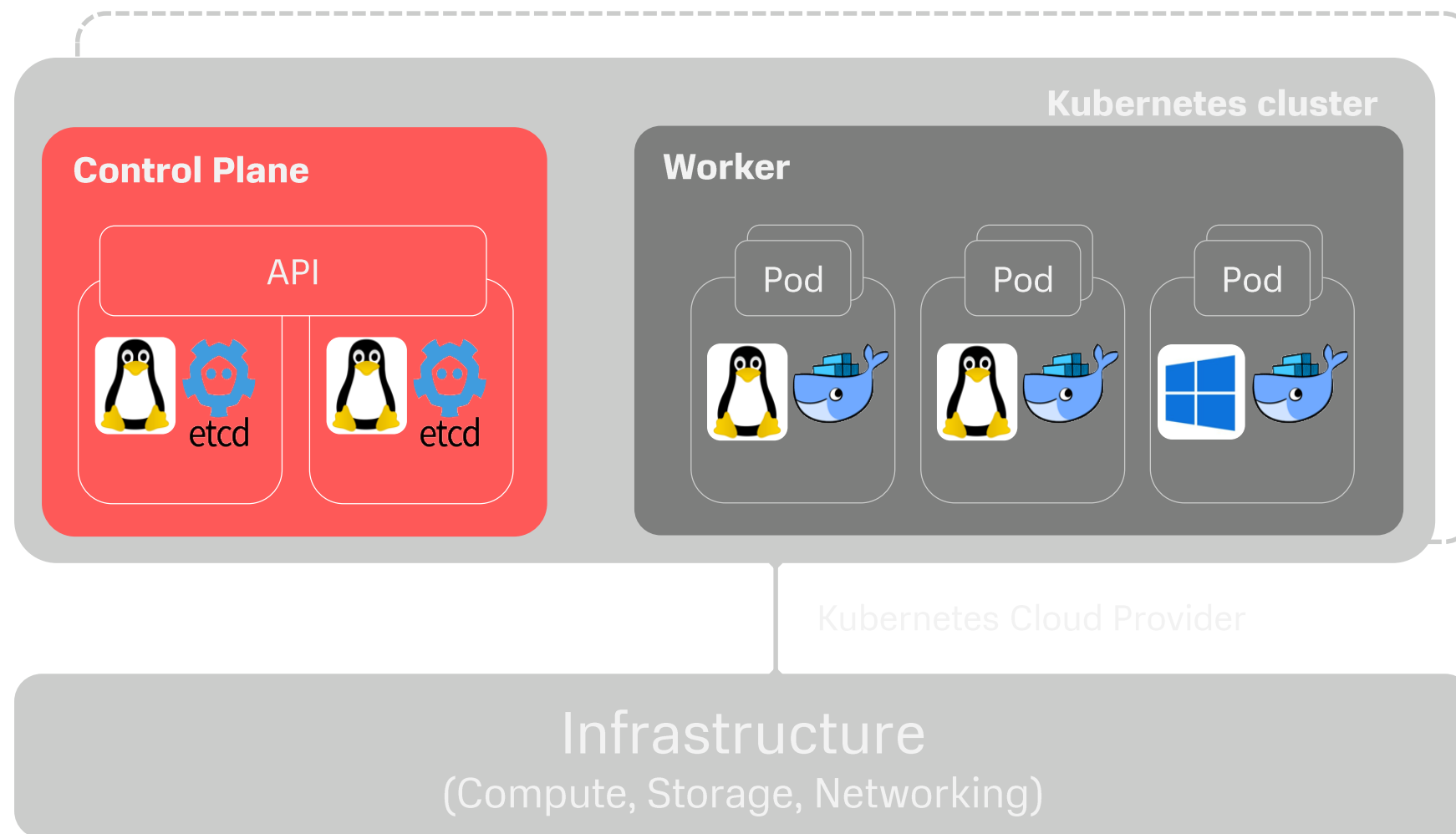


Хранение данных в контейнерах и Kubernetes

Николай Куликов
Консультант
по решениям VMware



Теперь у нас появляется оркестратор

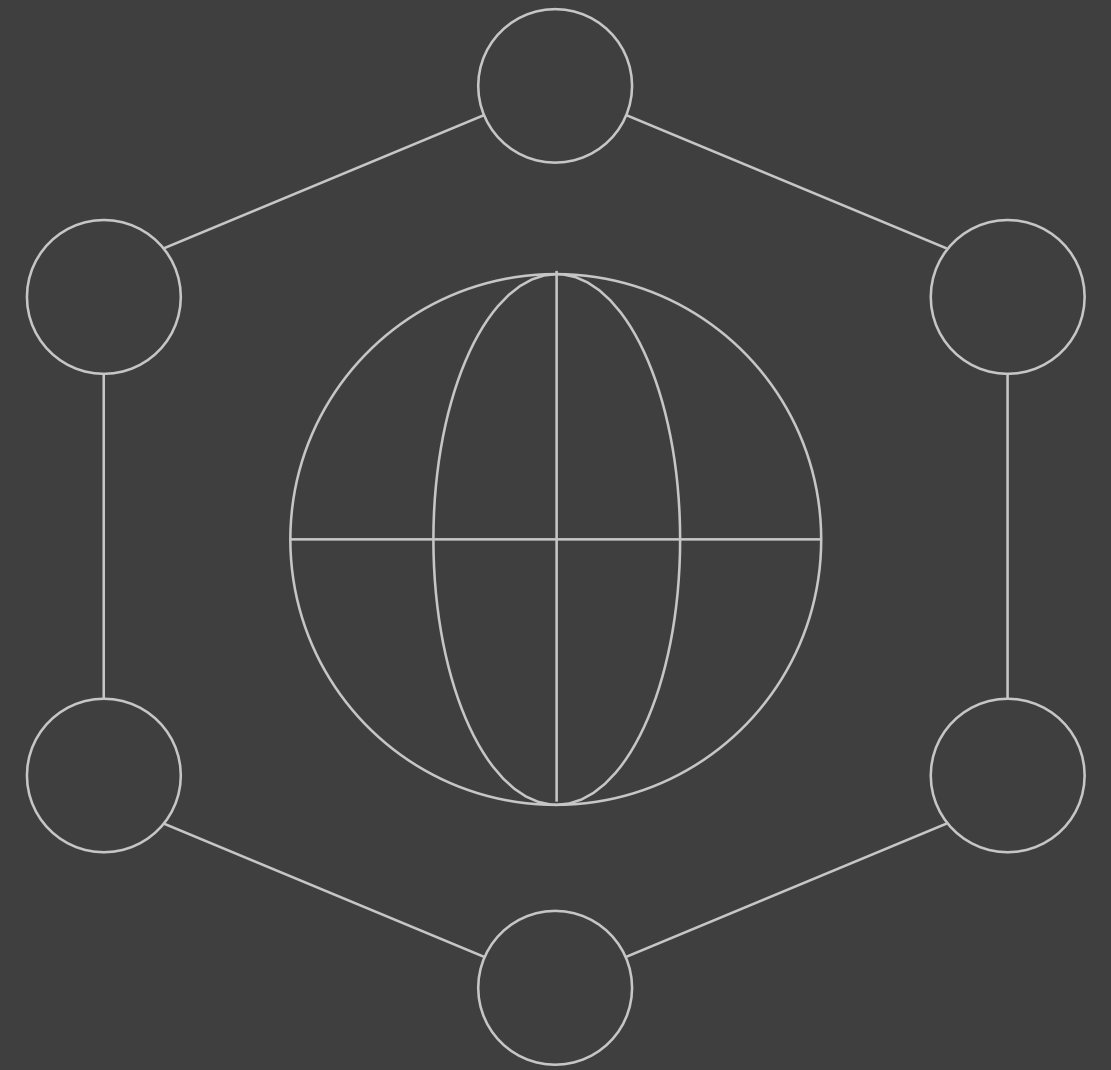


Дополнительный слой сверху, который занимается оркестрацией (управлением и автоматизацией) запуска и работы контейнеров

Некоторые особенности K8s:

- Управление путем указания целевого состояния
- Целевое состояние хранится в etcd
- Всё взаимодействие через API-сервер
- Расширяемость и интеграция через API

CSI



Container Storage Interface



CONTAINER
STORAGE
INTERFACE

Проект по выносу инфраструктурозависимого кода из K8s путем создания очередного слоя абстракции/трансляции вызов к хранилищу и разработке ответной части каждого вендора – CSI-драйвера.

Описывает базовый функционал работы с хранилищем:

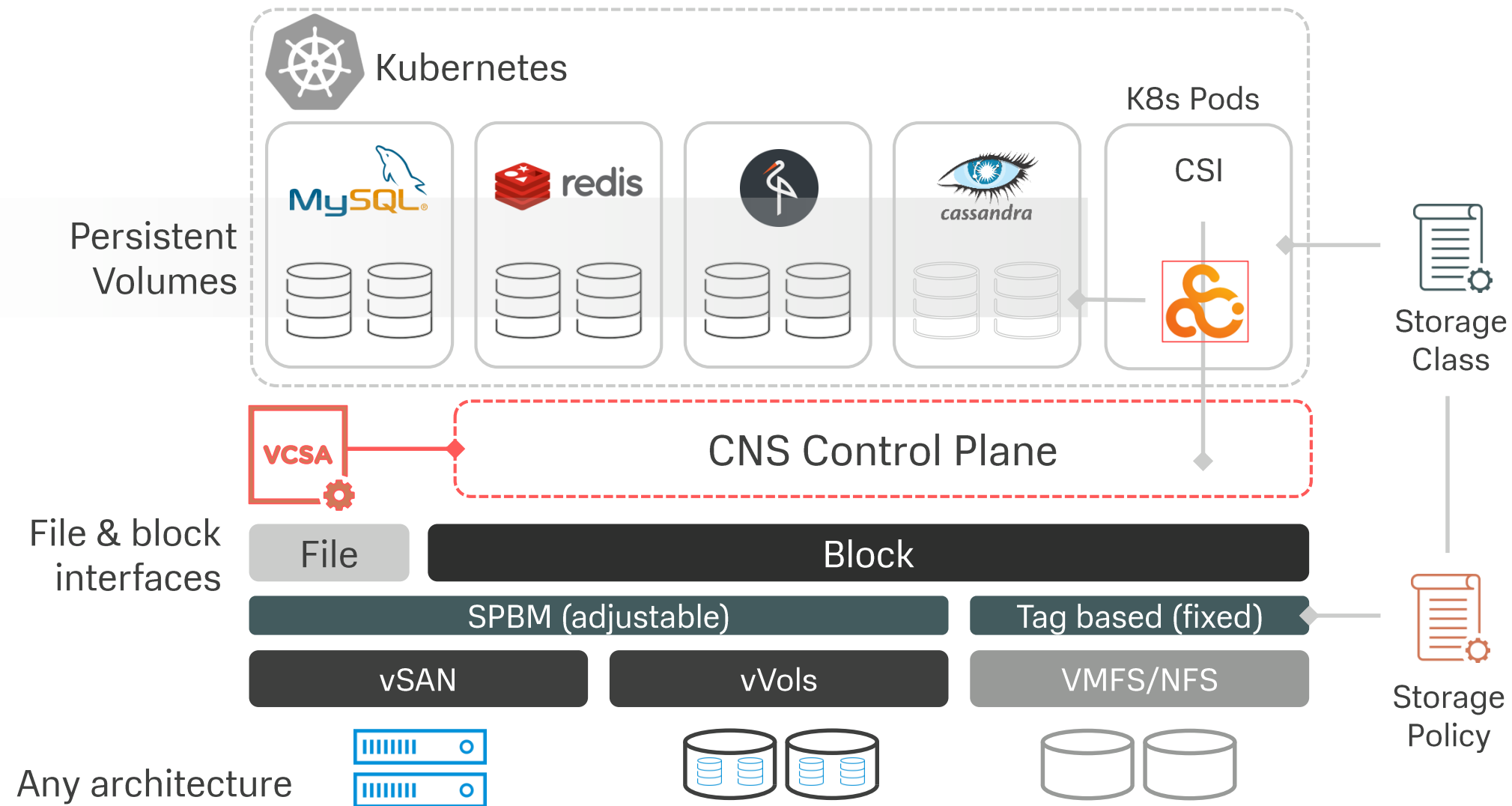
- Развертывание
- Расширение
- Клоны
- Снэпшоты
- Топология

Дальше на примере vSphere CSI

<https://vsphere-csi-driver.sigs.k8s.io/>

Cloud Native Storage в VMware

Платформа для предоставления ресурсов хранения для K8s



Динамическое размещение PV на базе **политик** хранения

Предоставляет **файловое** и **блочное** устройство хранения для **persistent volumes**

- Block: vSAN, vVols, VMFS
- File: vSAN

Абстрагирует инфраструктуру от разработчика

Единый подход к управлению данными для VM и контейнеров

CSI-драйвер имеет набор поддерживаемого функционала (API-вызовов)

Feature	Category	Supported vSphere CSI Driver Releases	vSphere 7.0u2	vSphere 7.0u1	vSphere 7.0	vSphere 6.7U3	Feature	Category	Supported vSphere CSI Driver Releases	vSphere 7.0u2	vSphere 7.0u1	vSphere 7.0	vSphere 6.7U3
Enhanced Object Health in UI for vSAN Datastores	GA	v2.0.0 to v2.3.0	✓	✓	✓	✓	Offline Volume Expansion support (Block Volume only)	Beta	v2.0.0 to v2.3.0	✓	✓	✓	✗
Dynamic Block PV support (Read-Write-Once Access Mode)	GA	v2.0.0 to v2.3.0	✓	✓	✓	✓	Encryption support via VMcrypt (Block Volume only)	GA	v2.0.0 to v2.3.0	✓	✓	✓	✗
Dynamic Virtual Volume (vVOL) PV support	GA	v2.0.0 to v2.3.0	✓	✓	✓	✓	Dynamic File PV support through vSAN 7.0 File Services on vSAN Datastores	GA	v2.0.0 to v2.3.0	✓	✓	✓	✗
Topology/Availability Zone support (Block Volume only)	Beta	v2.0.0 to v2.3.0	✓	✓	✓	✓	In-tree vSphere volume migration to CSI	Beta	v2.1.0 to v2.3.0	✓	✓	✗	✗
Static PV Provisioning	GA	v2.0.0 to v2.3.0	✓	✓	✓	✓	Online Volume Expansion support (Block Volume only)	Beta	v2.2.0 to v2.3.0	✓	✗	✗	✗
K8s Multi-node Control Plane support	GA	v2.0.0 to v2.3.0	✓	✓	✓	✓	XFS Filesystem support	Alpha	v2.3.0	✓	✓	✓	✓
WaitForFirstConsumer	Beta	v2.0.0 to v2.3.0	✓	✓	✓	✓	Raw Block Volume support	Alpha	v2.3.0	✓	✓	✓	✓

CNS поддерживает все основные дистрибутивы K8s

vSphere CSI Driver	Minimum Kubernetes Release	Maximum Kubernetes Release	Deprecated	End of Life
v2.3.0	1.19	1.21	No	-
v2.2.1	1.18	1.20	Yes	August 2022
v2.2.0	1.18	1.20	Yes	August 2022
v2.1.1	1.17	1.19	Yes	August 2022
v2.1.0	1.17	1.19	Yes	August 2022
v2.0.1	1.17	1.19	Yes	January 2022
v2.0.0	1.16	1.18	Yes	January 2022
v1.0.3	1.14	1.16	Yes	June 2021
v1.0.2	1.14	1.16	Yes	January 2021

CNS позволяет работать со всеми дистрибутивами K8s благодаря единому API:

VMware **Tanzu**

RedHat **OpenShift**

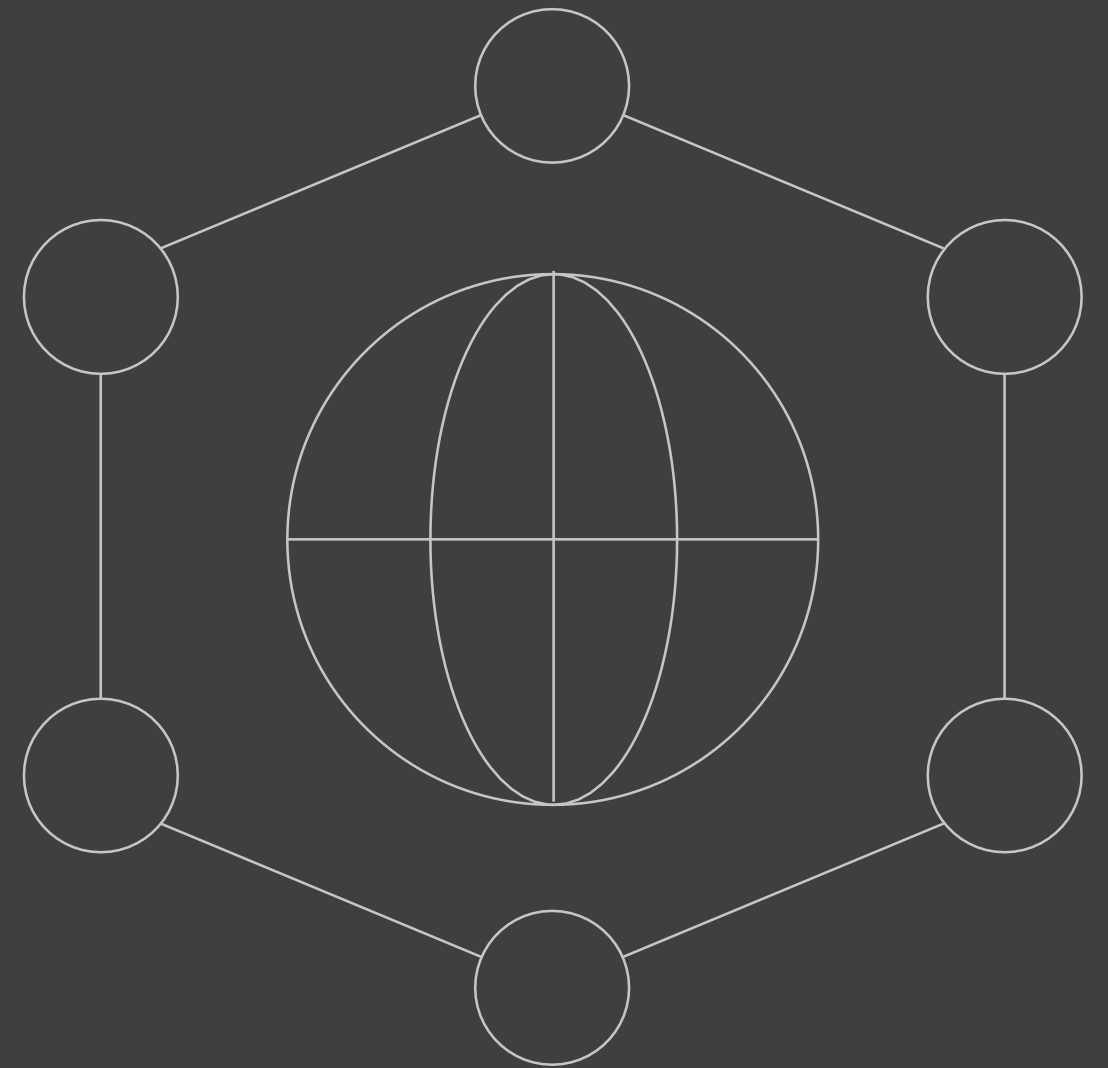
Google **Anthos GKE**

Docker Enterprise Edition

Rancher

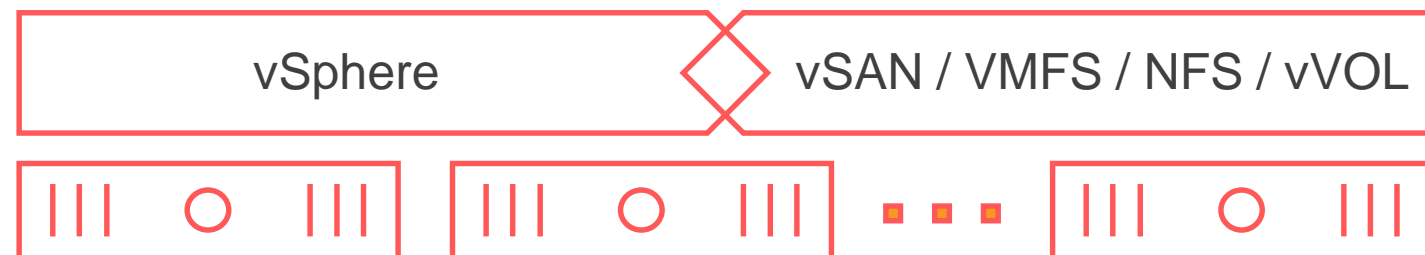
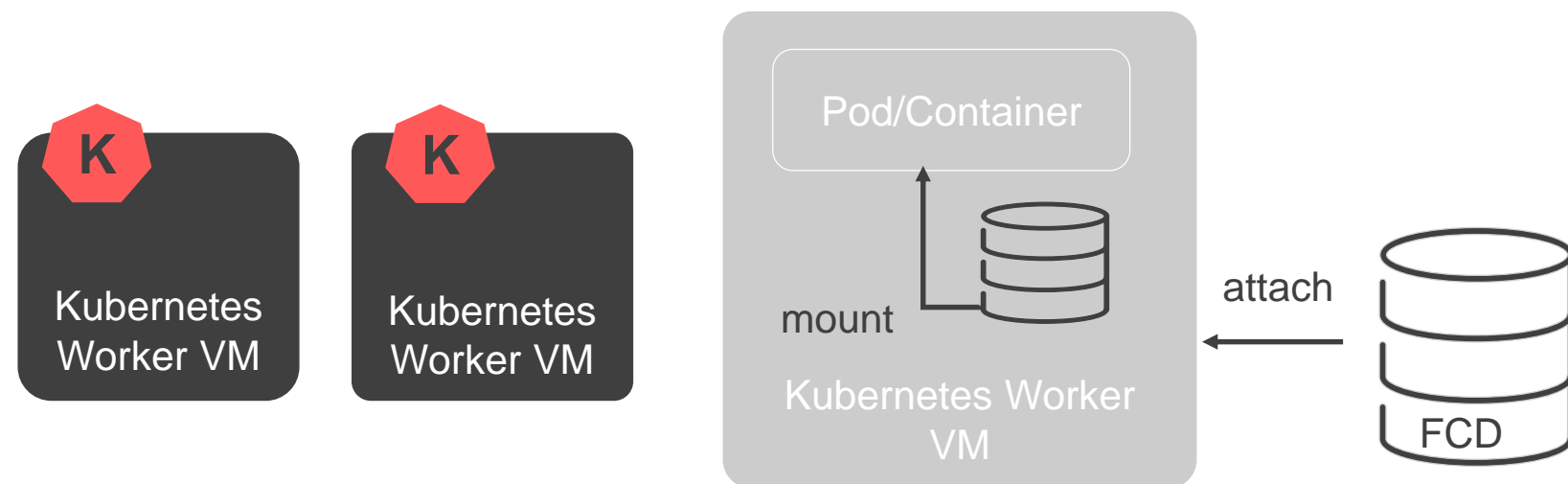
Vanilla

Persistent Volumes



Хранение данных в контейнерах

Пример, когда kubernetes-кластер развернут в виде VM на vSphere



K8s в формате VM поверх vSphere:

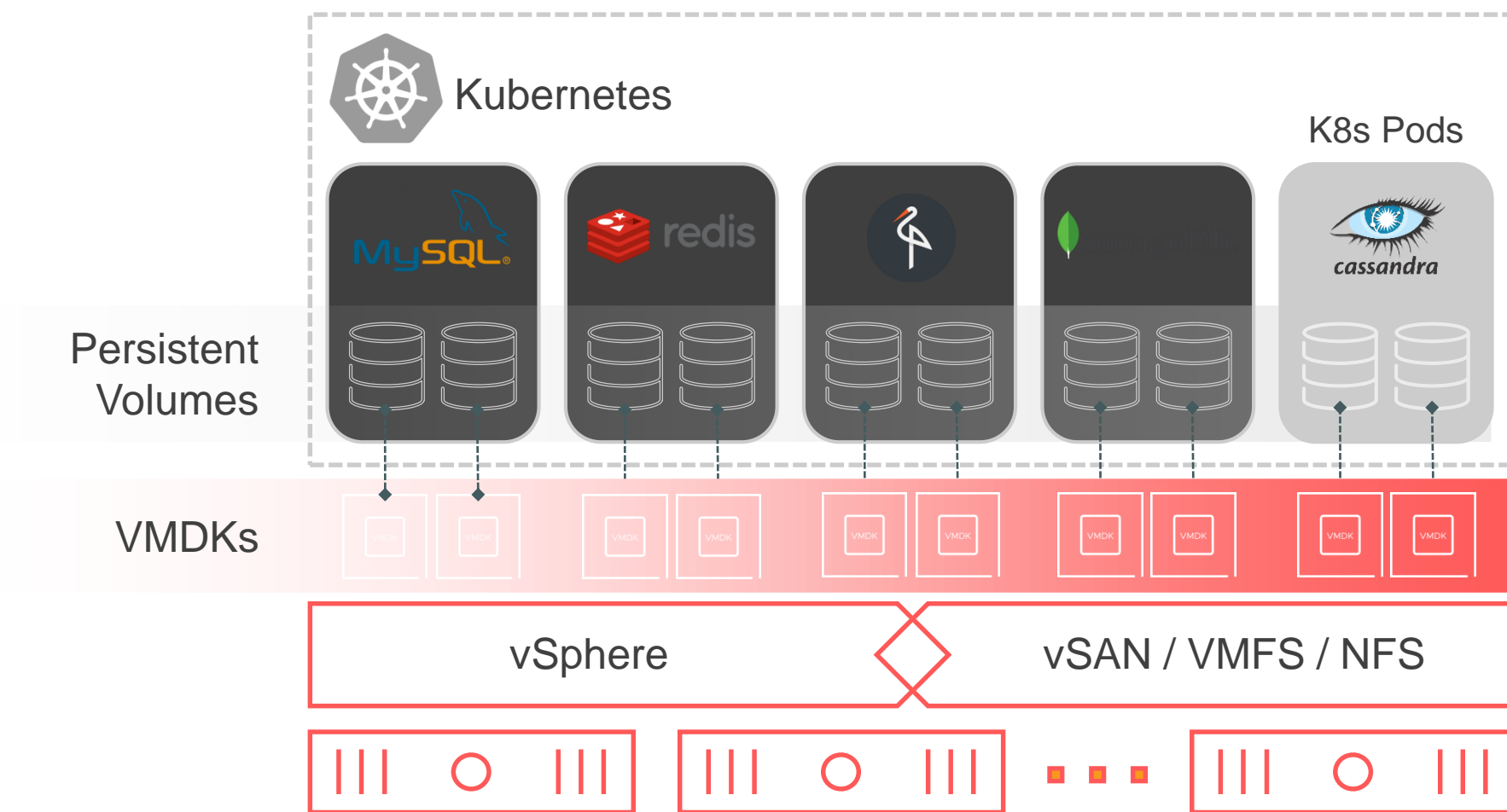
Создается FCD/IVD диск в vSphere

Нужно подключить **диск, который будет хранить состояние**

Том **форматируется и монтируется** в Pod/Container

PersistentVolume (PV) – место хранения данных

Ключевой примитив хранения в Kubernetes



Устройство, где **храним** актуальные данные
Жизненный цикл **не зависит** от контейнеров
Может быть создан **статически** или **динамически**
В динамическом режиме вы никогда **не создаете** эти устройства **вручную**

PersistentVolumeClaim (PVC) – запрос ресурсов хранилища

Ресурсы хранения **запрашиваются** пользователем:

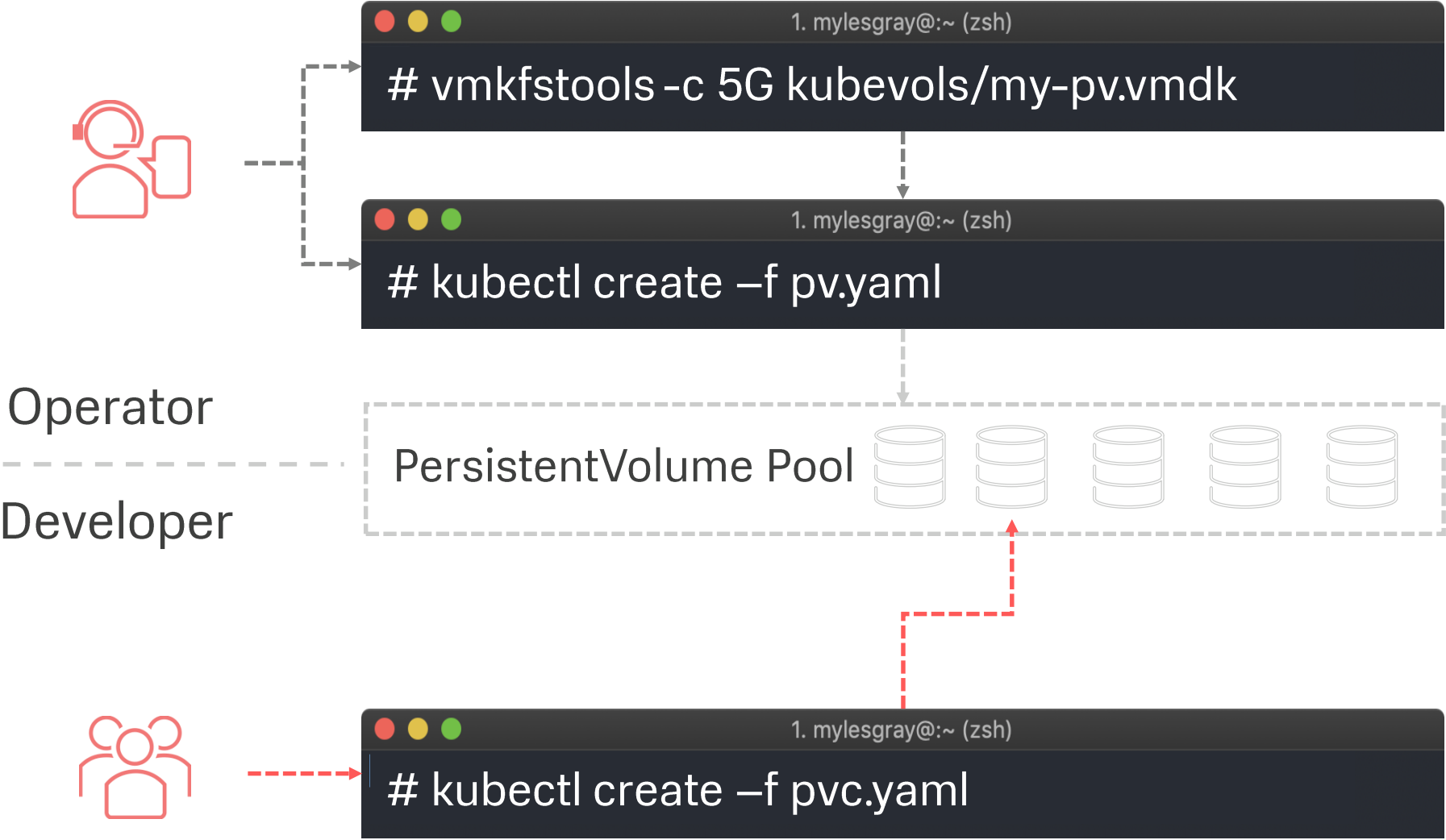
PersistentVolume в виде **статического пути**
(Static Provisioning)

StorageClass (Dynamic Provisioning)

```
1. mylesgray@:~ (zsh)
~ cat PersistentVolumeClaim.yaml

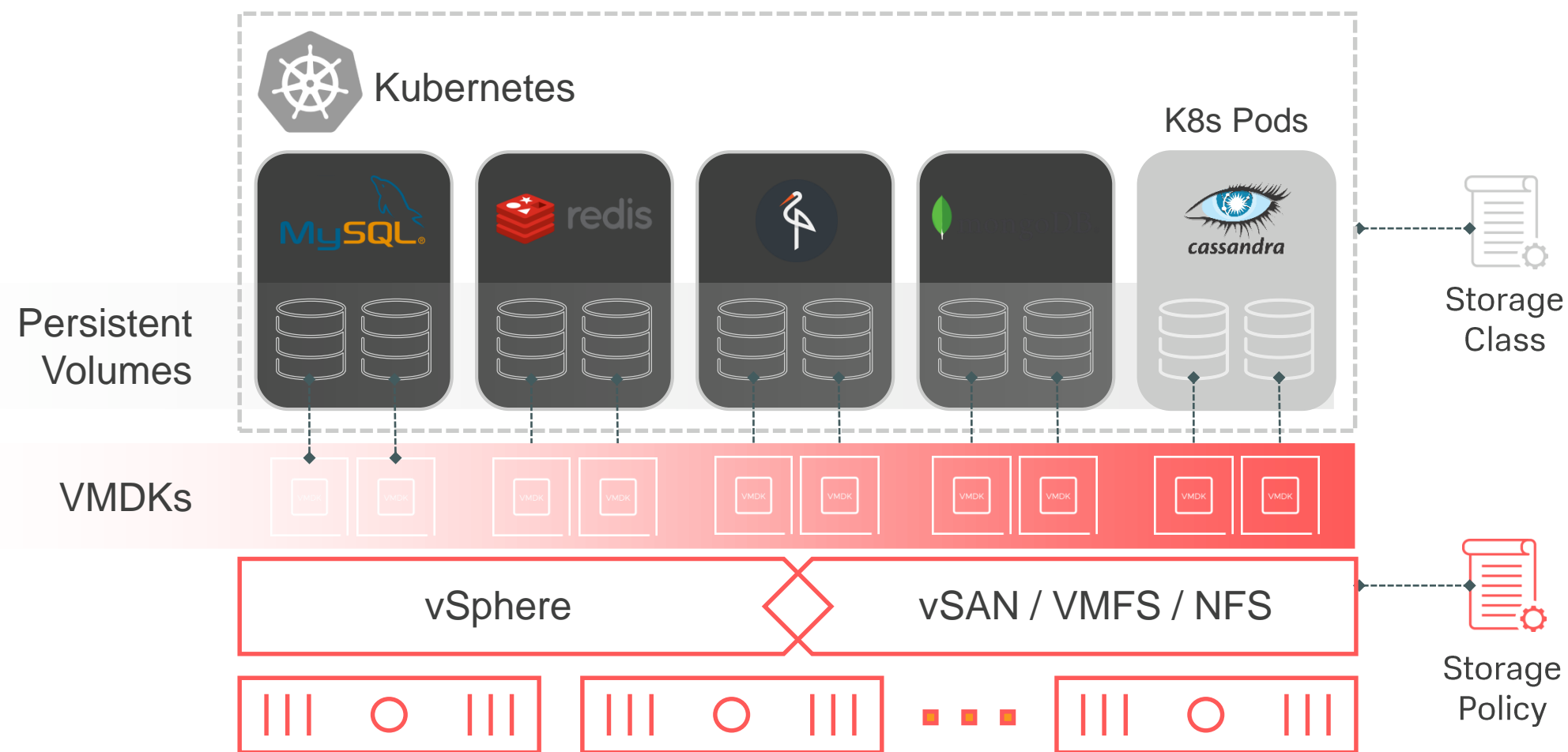
kind: PersistentVolumeClaim
metadata:
  name: my-pvc
spec:
  accessModes:
  - ReadWriteOnce
  resources:
    requests:
      storage: 5Gi
  storageClassName: vsan-default
```

Static Provisioning – ручной режим



1. Ops создает каждый FCD/VMDK в ручном режиме
2. Ops регистрирует VMDK как PV, прописывая путь
3. Разработчик назначает PV через PVC
4. Разработчик создает Pod, который использует PVC

StorageClass (SC) – способ динамически назначать устройства



Динамически создает тома на нужном устройстве

Рекомендуемый способ управления PV

Полностью автоматизированно и не требует вовлечения людей

Подозрительно напоминает VMware Storage Policy (since 2014 год)

StorageClass (SC)

```
1. mylesgray@:~ (zsh)
~ cat StorageClass.yaml

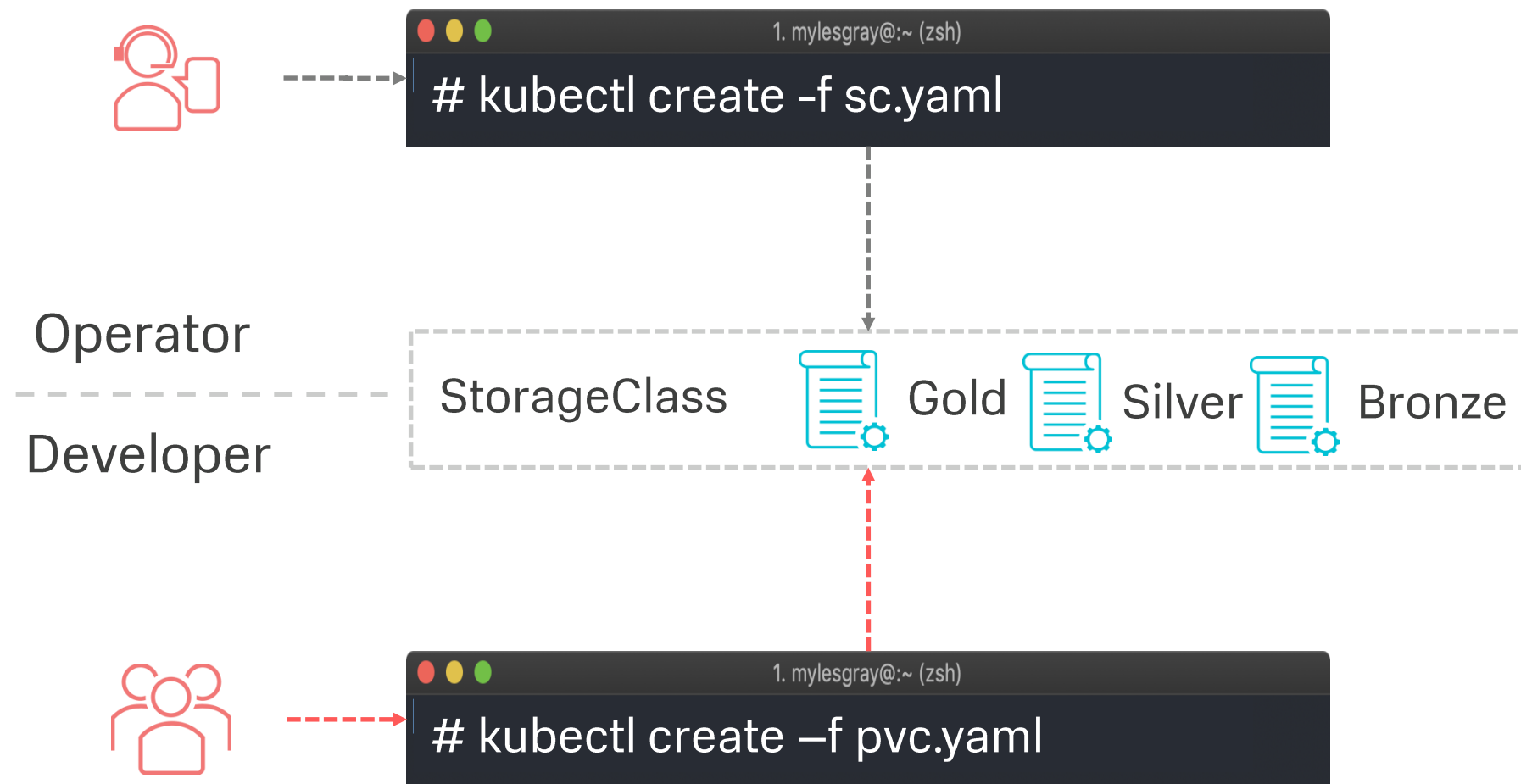
kind: StorageClass
metadata:
  name: vsan-default
  provisioner: csi.vsphere.vmware.com
parameters:
  StoragePolicyName: "vSAN Default Storage Policy"
```

Даем **название**

Описываем, с кем будет **взаимодействие**
(provisioner)

Передаем **параметры**

Dynamic Provisioning



1. Ops **создает StorageClass**
2. Разработчик **направляет PVC** в StorageClass
3. Система **автоматически создает** VMDK и PV
4. Система **подключает** PV к PVC

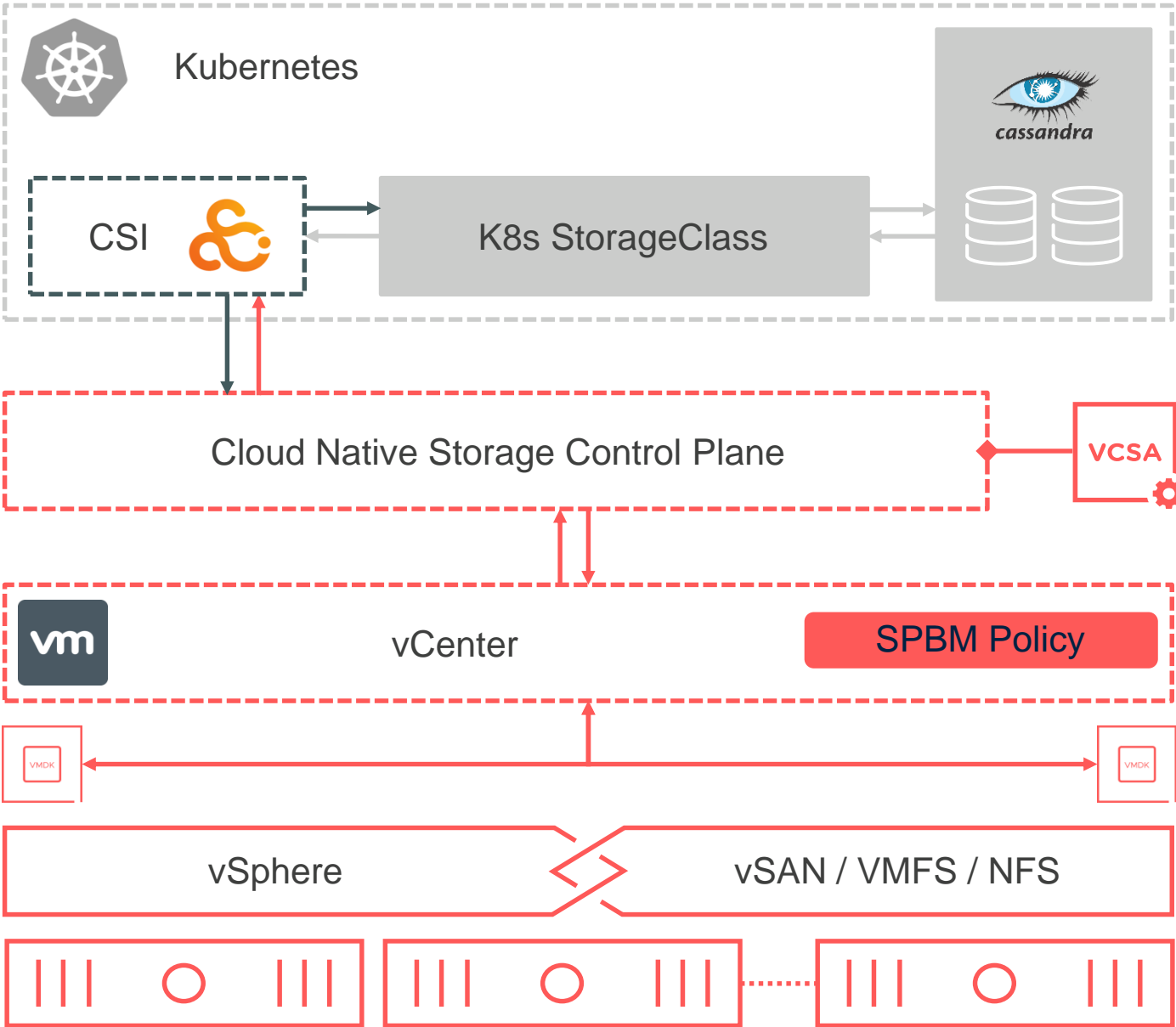
Динамическое размещение блочного PV

ReadWriteOnce

```
1. mylesgray@:~ (zsh)
~ cat StorageClass.yaml

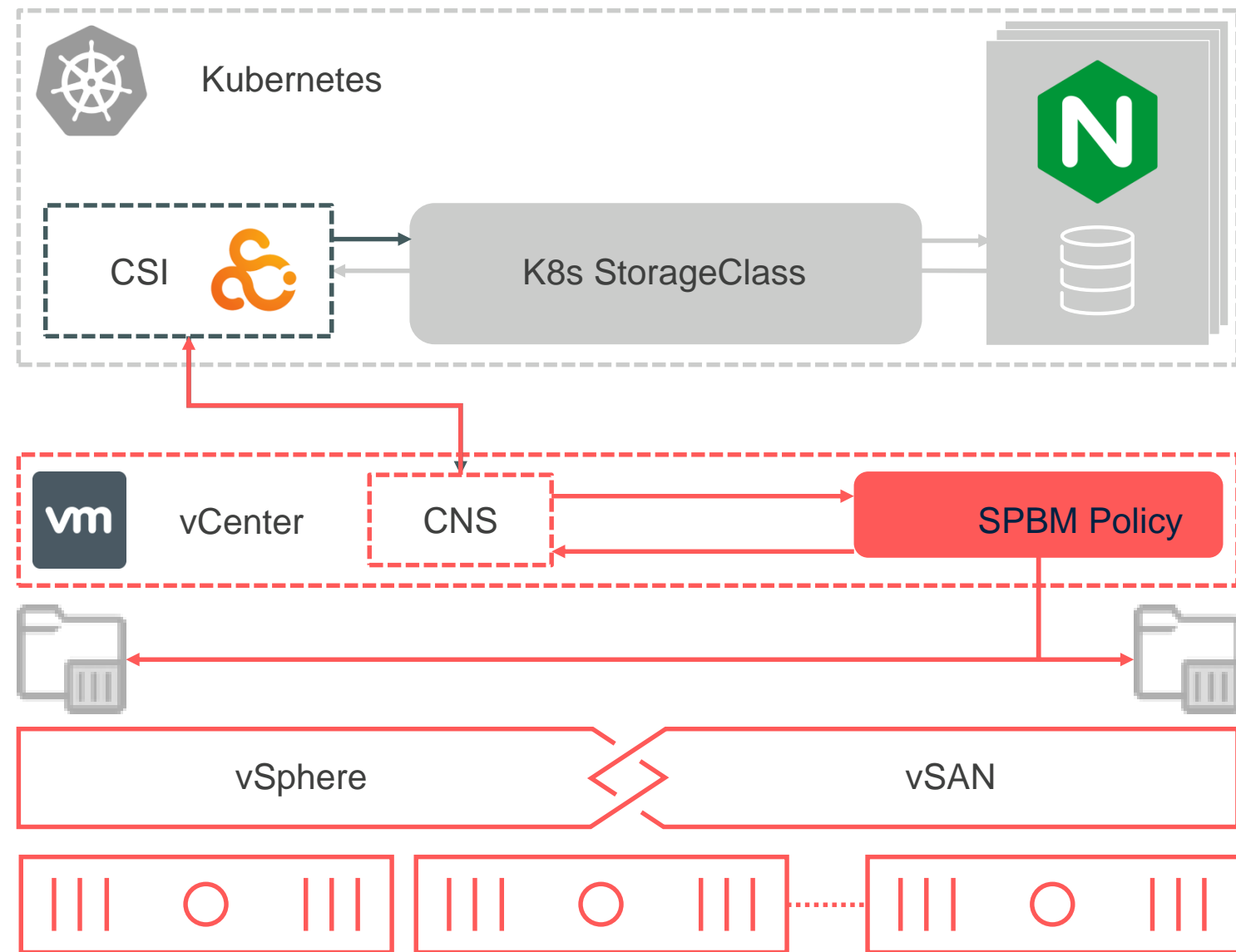
kind: StorageClass
metadata:
  name: vsan-default
  provisioner: csi.vsphere.vmware.com
parameters:
  StoragePolicyName: "vSAN Default Storage Policy"
```

General	
Name	vSAN Default Storage Policy
Description	Storage policy used as default for vSAN datastores
Rule-set 1: VSAN	
Placement	
Storage Type	VSAN
Site disaster tolerance	None - standard cluster
Failures to tolerate	1 failure - RAID-1 (Mirroring)
Number of disk stripes per object	1



Динамическое размещение файлового PV

ReadWriteMany



Необходимо, когда **много контейнеров должны иметь доступ к одним** и тем же данным

Реализуется по файловым, а не по блочным протоколам (так как нужно управлять блокировками/доступами)

Необходимо иметь аналогичное динамическому развертывание

Динамическое размещение файлового PV

ReadWriteMany

```
1. mylesgray@:~ (zsh)
~ cat StorageClass.yaml

kind: StorageClass
apiVersion: storage.k8s.io/v1
metadata:
  name: vsan-file
provisioner: csi.vsphere.vmware.com
parameters:
  storagepolicyname: "vSAN Default Storage
Policy"
  csi.storage.k8s.io/fstype: nfs4
```

Динамически создает файловую шару на vSAN

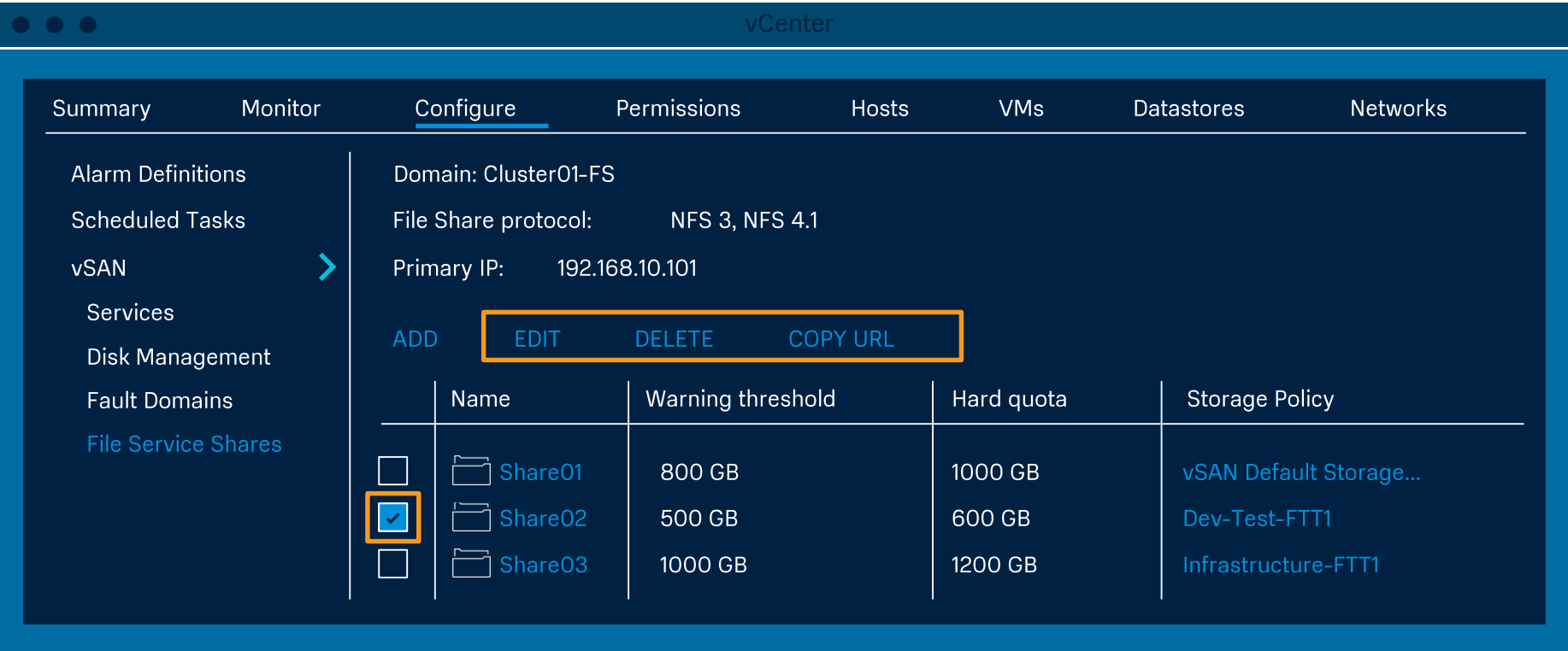
ACL и RW указываются в момент создания

Требования к хранению через тот же SPBM, аналогично RWO

Полностью автоматизированный процесс управления шарами

Файловый доступ

ReadWriteMany



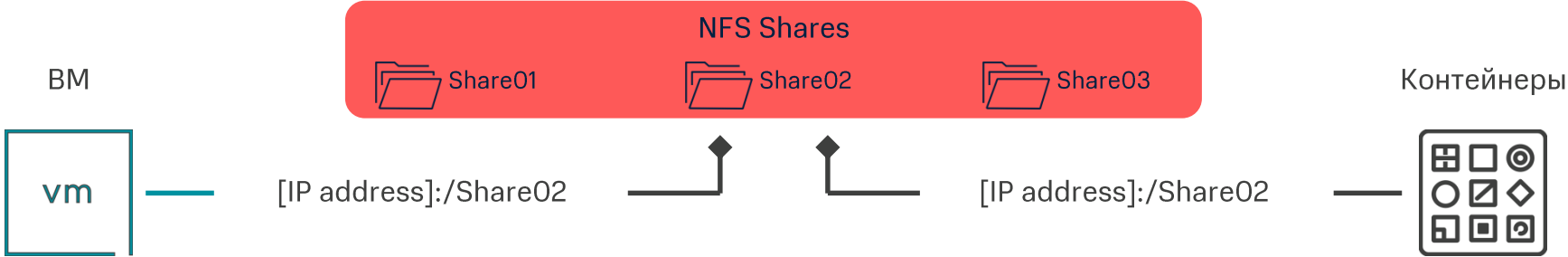
Включается несколькими кликами из vCenter

Простое создание сетевой папки:

- Указываем имя
- Storage Policy
- Квоты
- Метки
- Ограничение по сетевому доступу

Предоставляет подключение по NFS

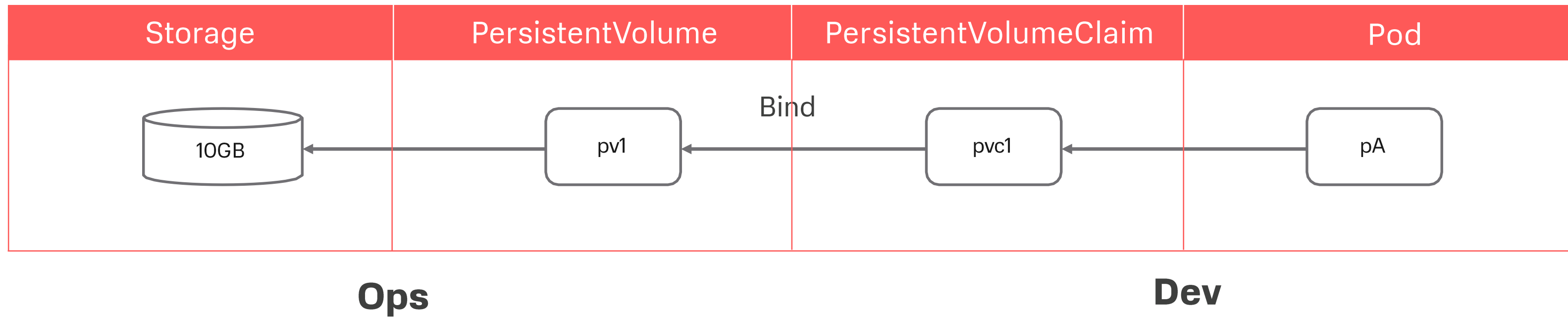
NFS 3
NFS 4.1



Как это выглядит в UI



В итоге





vSAN-Cluster

ACTIONS

- Summary
- Monitor**
- Configure
- Permissions
- Hosts
- VMs
- Datastores
- Networks
- Updates

- 10.198.24.1
 - vSAN-DC
 - vSAN-Cluster**
 - 10.198.17.5
 - 10.198.17.6
 - 10.198.17.7
 - 10.198.17.8
 - 10.198.23.255
 - 10.198.24.0
 - ESX Agents
 - k8s-master01
 - k8s-worker01
 - k8s-worker02
 - k8s-worker03
 - k8s-worker04
 - k8s-worker05

- VM DRS Score
- CPU Utilization
- Memory Utilization
- Network Utilization
- Resource Allocation**
 - CPU
 - Memory
 - Storage
- Utilization
- Storage Overview
- Security
- vSAN**
 - Skyline Health
 - Virtual Objects
 - Physical Disks
 - Resyncing Objects
 - Proactive Tests
 - Capacity
 - Performance
 - Performance Diagnostics
 - Support
 - Data Migration Pre-check
- Cloud Native Storage**
 - Container Volumes

Container providers: Kubernetes

[LEARN MORE](#)








REAPPLY POLICY | Add filter

<input type="checkbox"/>	<input type="checkbox"/>	Volume Name	Type	Label	Datastore	Health Status	Capacity Quota
<input type="checkbox"/>	<input type="checkbox"/>	pvc-b95dfecf-cfb1-4d...	BLOCK	component master SEE ALL	vsanDatast...	✓ Accessible	8.00 GB
<input type="checkbox"/>	<input type="checkbox"/>	pvc-345a9263-47d8-4...	BLOCK	app postgresql SEE ALL	vsanDatast...	✓ Accessible	8.00 GB
<input type="checkbox"/>	<input type="checkbox"/>	pvc-1546ef6f-d0a9-44...	FILE	app space-odyssey	vsanDatast...	✓ Healthy	50.00 GB
<input type="checkbox"/>	<input type="checkbox"/>	pvc-cd20d607-2344-...	BLOCK	app postgresql SEE ALL	vsanDatast...	✓ Accessible	8.00 GB
<input type="checkbox"/>	<input type="checkbox"/>	pvc-568294e2-2f40-4...	BLOCK	app cassandra release cassandra	vsanDatast...	✓ Accessible	8.00 GB
<input type="checkbox"/>	<input type="checkbox"/>	pvc-f3bd64a5-21c2-40...	BLOCK	app concourse-worker release concourse	vsanDatast...	✓ Accessible	20.00 GB
<input type="checkbox"/>	<input type="checkbox"/>	pvc-a815febc-f295-4d...	BLOCK	app mongodb SEE ALL	vsanDatast...	✓ Accessible	8.00 GB
<input type="checkbox"/>	<input type="checkbox"/>	pvc-367e0d46-ca07-4...	BLOCK	app mongodb SEE ALL	vsanDatast...	✓ Accessible	8.00 GB
<input type="checkbox"/>	<input type="checkbox"/>	pvc-218661b0-2d5e-4...	BLOCK	app gitaly release gitlab	vsanDatast...	✓ Accessible	50.00 GB
<input type="checkbox"/>	<input type="checkbox"/>	pvc-2d1ab2d0-c042-4...	BLOCK	app concourse-worker release concourse	vsanDatast...	✓ Accessible	20.00 GB
<input type="checkbox"/>	<input type="checkbox"/>	pvc-0384508e-15be-4...	BLOCK	app prometheus SEE ALL	vsanDatast...	✓ Accessible	8.00 GB
<input type="checkbox"/>	<input type="checkbox"/>	pvc-f019d144-f794-42...	BLOCK	app minio SEE ALL	vsanDatast...	✓ Accessible	10.00 GB
<input type="checkbox"/>	<input type="checkbox"/>	pvc-50f39722-e7b3-4...	BLOCK	app minio SEE ALL	vsanDatast...	✓ Accessible	10.00 GB

Сопоставление Kubernetes Volumes с объектами vSphere



Container providers: Kubernetes

REAPPLY POLICY | Add filter

<input type="checkbox"/>	Volume Name	Label
<input type="checkbox"/>	 pvc-b95dfecf-cfb1-4d...	component m
<input type="checkbox"/>	 pvc-345a9263-47d8-4...	app postgresq
<input checked="" type="checkbox"/>	 pvc-1546ef6f-d0a9-44...	app space-ody
<input type="checkbox"/>	 pvc-cd20d607-2344-...	app postgresq
<input type="checkbox"/>	 pvc-568294e2-2f40-4...	app cassandra
<input type="checkbox"/>	 pvc-f3bd64a5-21c2-40...	app concourse
<input type="checkbox"/>	 pvc-a815febc-f295-4d...	app mongodb

pvc-1546ef6f-d0a9-4414-b6e8-4d78d365c6ca

Basics | Kubernetes objects

Type	FILE
Volume ID	file:9602906e-bd6c-464d-a6bc-008f6ef9fff2
Datastore	 vsanDatastore
Storage Policy	 vSAN Default Storage Policy
Compliance Status	<input checked="" type="checkbox"/> Compliant
Health Status	<input checked="" type="checkbox"/> Healthy









- Issues and Alarms >
- Performance >
- Tasks and Events >
- vSphere DRS >
- Resource Allocation >
- Utilization
- Storage Overview
- Security
- vSAN >
- Cloud Native Storage >

Container Volumes

Container providers: **Kubernetes**

LEARN MORE

REAPPLY POLICY | Add filter

<input type="checkbox"/>	Volume Name	Type
<input type="checkbox"/>	  pvc-d1eae183-fa25-4e3...	FILE
<input type="checkbox"/>	  pvc-3dc8cd77-fc1b-4e...	BLO
<input type="checkbox"/>	  pvc-36488f67-51f6-45...	BLO
<input type="checkbox"/>	  pvc-e22a5d57-fa8c-4f...	BLO

1-4 of 4 container volumes

pvc-d1eae183-fa25-4e3e-8e94-a04f8cb2d3f0

Basics **Kubernetes objects**

Kubernetes cluster: Cormac-Demo **Vanilla Kubernetes** ← **K8s Cluster Name & Type**

Persistent volume

Name	pvc-d1eae183-fa25-4e3e-8e94-a04f8cb2d3f0
Labels	--

Namespace default ← **K8s Namespace**

Persistent volume claim

Name	file-pvc ← PVC Name
Labels	--

Pod file-pod-a, file-pod-b ← **Pods using PV**

Physical Placement

52406c2c-0ac6-4505-b9ac-7a6b1d7ddf15



Group components by host placement

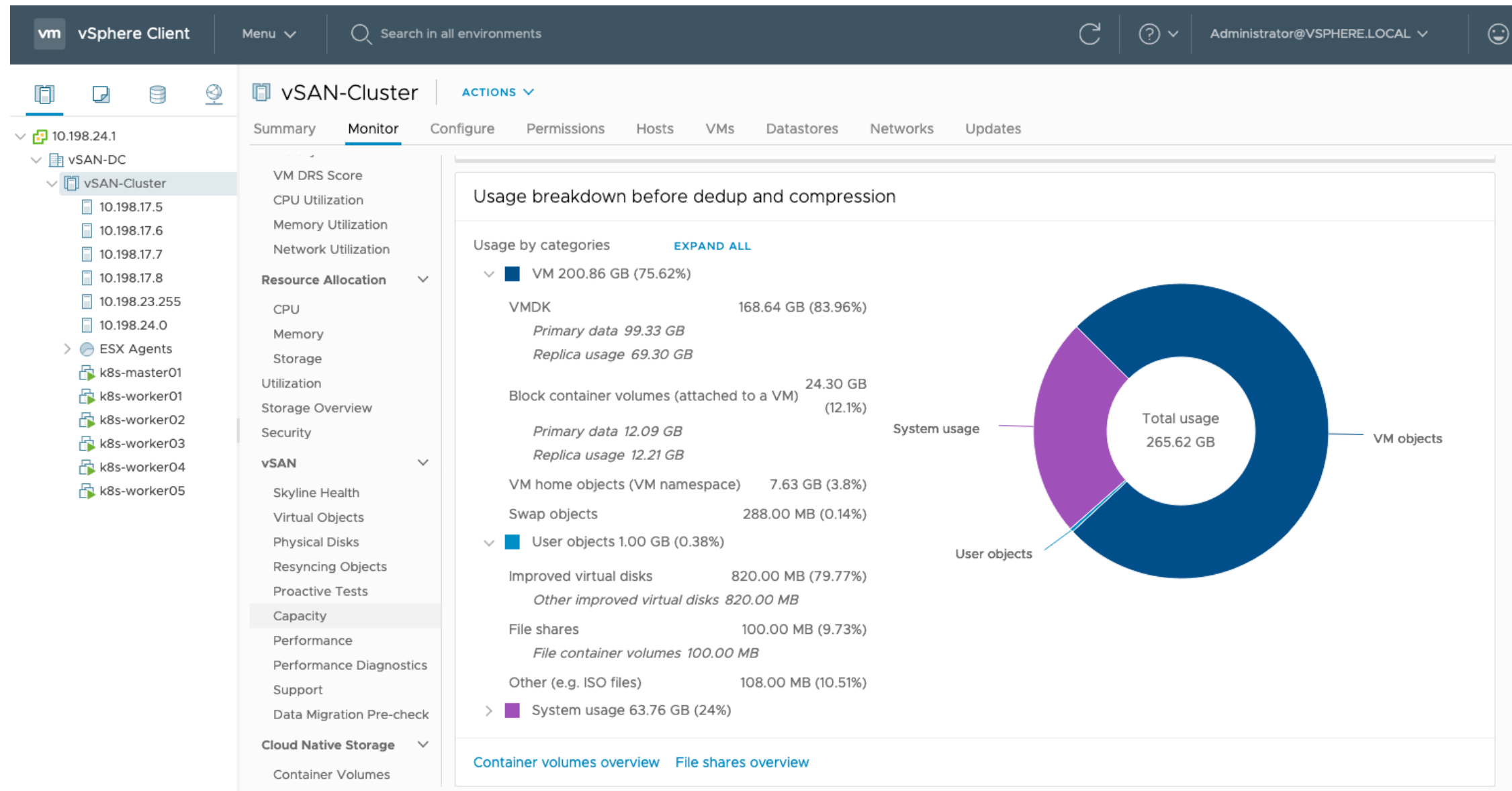
Hosts and Disks

Name	State
10.198.17.6	✔ Health...
Disk Group (Local VMware Disk (mpx.vmhba0:C0:T4:L0))	✔ Health...
Local VMware Disk (mpx.vmhba0:C0:T1:L0)	✔ Health...
10.198.24.0	✔ Health...
Disk Group (Local VMware Disk (mpx.vmhba0:C0:T4:L0))	✔ Health...
Local VMware Disk (mpx.vmhba0:C0:T1:L0)	✔ Health...
10.198.17.5	✔ Health...
Disk Group (Local VMware Disk (mpx.vmhba0:C0:T4:L0))	✔ Health...
Local VMware Disk (mpx.vmhba0:C0:T2:L0)	✔ Health...

Virtual Object Components

pvc-1546ef6f-d0a9-4414-b6e... > 52406c2c	
Witness	✔ Active

Мониторим емкость и ресурсы



Спасибо!

