KubeCon

CloudNativeCon

Europe 2018

Kubernetes Local Persistent Volumes in Production

May 2018

Michelle Au, Google
Ian Chakeres, Salesforce

Agenda

- Why Kubernetes and local storage at Salesforce
- Feature overview
- Local volume lifecycle
- Demo
- Future roadmap

Why Kubernetes Local Volumes at Salesforce

The success of Salesforce's customers is driving storage needs that look exponential or cubic rather than linear

Keeping ahead of this curve is the responsibility of our infrastructure team

In the last year, our Kubernetes (k8s) fleet size doubled and we energized >10 petabytes storage service capacity

Kubernetes Benefits for Storage Services

Our engineers are embracing the Kubernetes development lifecycle for storage services across multiple substrates

- Leveraging local storage, cloud-native, and secure
- Immutable containers, declarative manifests, and active reconciliation
- From manifest check in, to production in less than 30 minutes

Why Local vs Remote?

Performance: SSDs

Cost: Cheaper than remote storage

Utilization: Use spare disks

Tradeoffs

Inflexible placement

Lower availability

Lower data durability

NOT general purpose storage solution!

Use Cases

Distributed datastores

- Tolerant of node failure and data loss
- For example: Ceph, Cassandra, Bookkeeper, HDFS, HBase

Applications with intensive read/write profiles

- Large fast on-disk caches
- Avoid cold restarts
- Interactive analytic applications

HostPath Volume Problems

Not secure

Not portable

Not disk accountable

Not scalable

Complex operators

```
apiVersion: v1
kind: Pod
metadata:
  name: my-pod
spec:
  nodeName: some-node
  volumes:
  - name: data
    hostPath:
      path: /mnt/some-disk
  containers:
  ...
```

Local Persistent Volumes

Secure

Portable

Disk accountable

Scalable

StatefulSets

```
apiVersion: v1
kind: Pod
metadata:
  name: my-pod
spec:
  volumes:
  - name: data
    persistentVolumeClaim:
      claimName: my-pvc
  containers:
  ...
```

Feature Status

Beta in Kubernetes 1.10

Local disk as a Persistent Volume (PV)

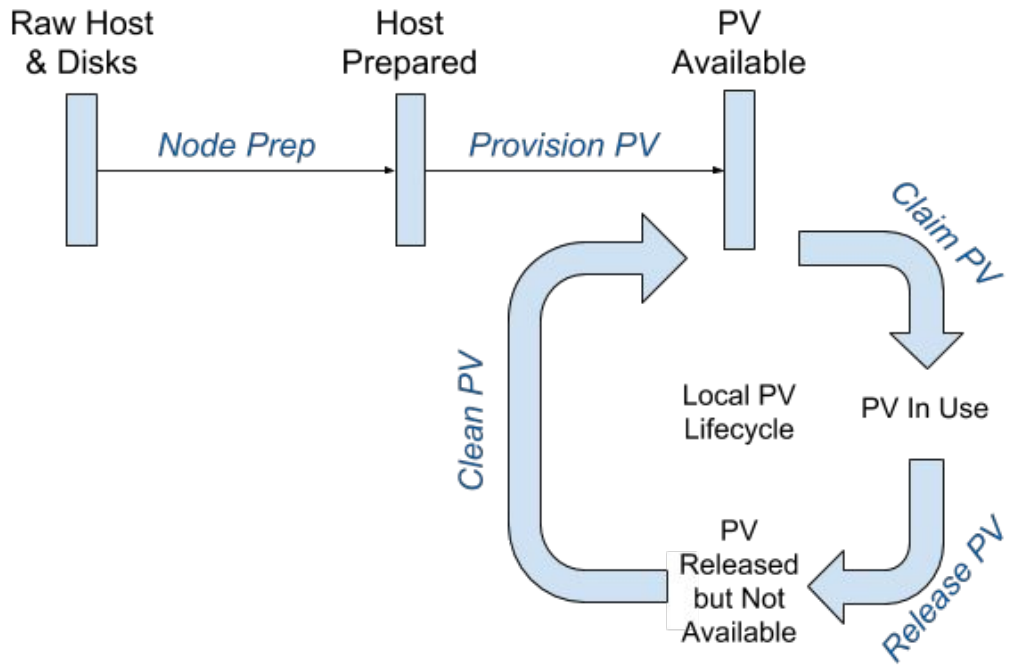
- Must be formatted and mounted first
- Dynamic provisioning NOT supported (yet)

Scheduler enhancements

- Data gravity
- Volume binding looks at Pod requirements
- Multiple PVCs in a Pod

Local Volume Lifecycle

1. Node and disk preparation
   - Specific to environment
2. Kubernetes local PV management
   - Generic to Kubernetes
   - Provided by local volume **STATIC** provisioner



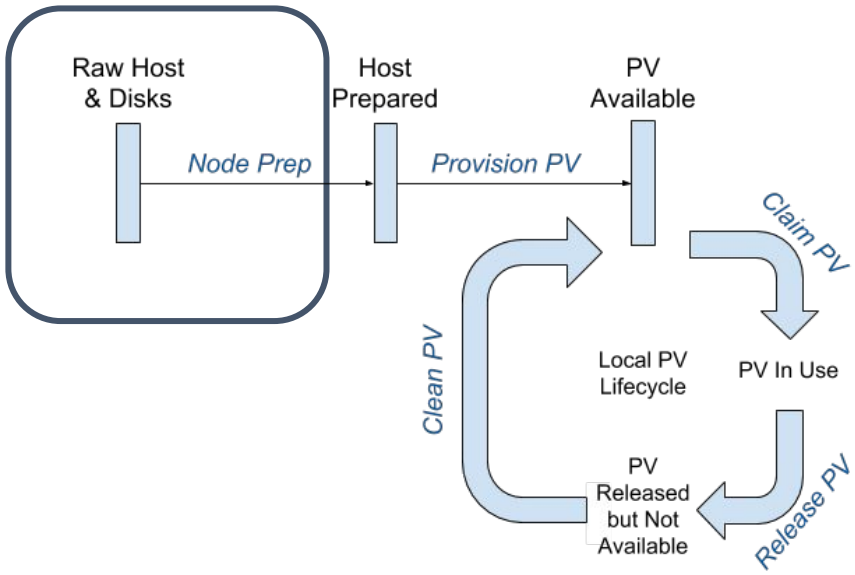
Node Preparation

Many choices

- Partitions
- Channel partitioning
- RAID 0, 1, 5, 6, 10
- LVM
- and more...

Which one (or more) to choose?

It depends...



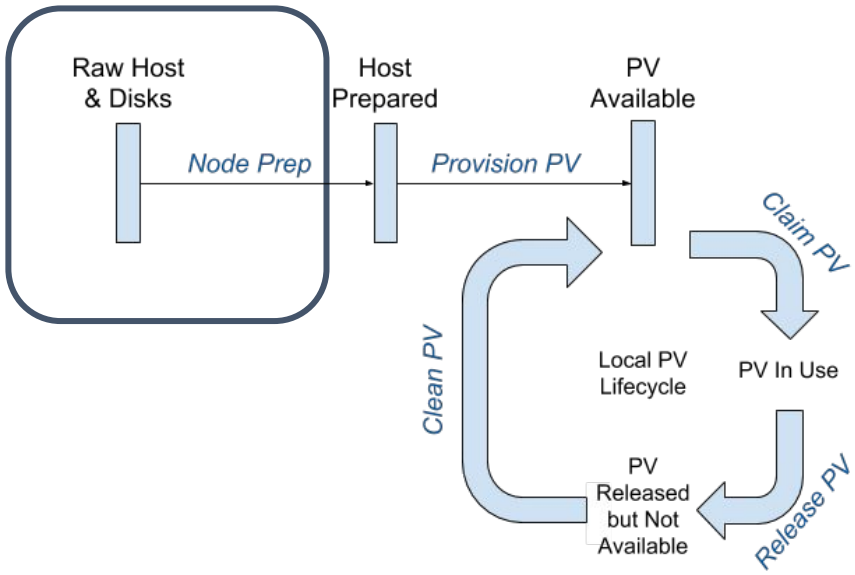
Node Preparation

Workload requirements

- Performance
- Capacity
- Scaling
- Durability

Ops requirements

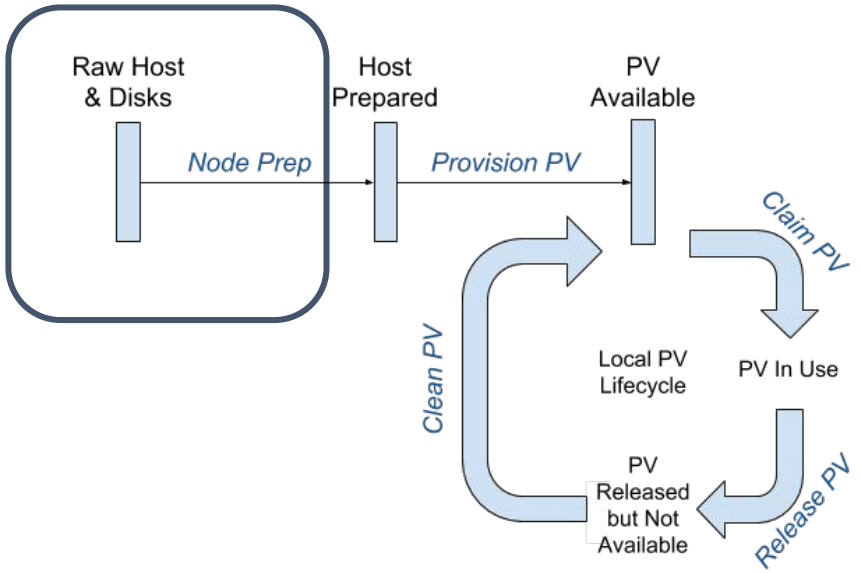
- Cost
- Utilization
- Repair
- Management
- Platform limitations
- Existing processes and tools



Node Preparation (GKE)

Specific to Google Kubernetes Engine environment
https://cloud.google.com/kubernetes-engine/docs/concepts/local-ssd

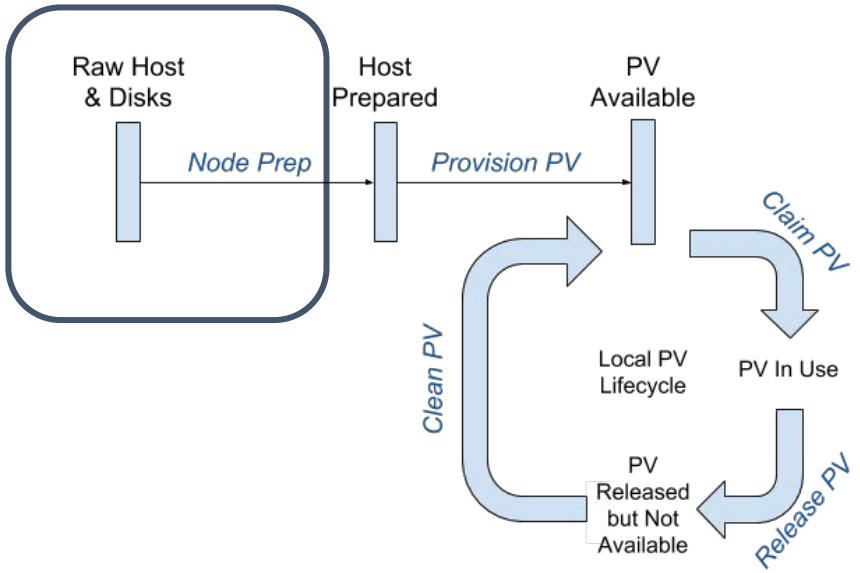
1. Create a cluster or node pool with local SSDs
2. Node VM setup script formats and mounts local SSDs to discovery directories for LV provisioner



Node Preparation (Salesforce)

Specific to Salesforce environment

1. Manifests describe and declare servers configurations
2. Nodeprep daemonset scans new servers
3. Performs volume operations for desired resources
   a. Partition, clean, and mount
4. Mounts or links resources to discovery directories for LV provisioner
5. Marks node with nodeprep complete label for Daemonset magic



DaemonSet Magic

Salesforce environment example

nodeprep daemonset

```
<snip>
spec:
  affinity:
    nodeAffinity:
      requiredDuringSchedulingIgnoredDuringExecution:
        nodeSelectorTerms:
        - matchExpressions:
          - key: storage.salesforce.com/nodeprep
            operator: DoesNotExist
```

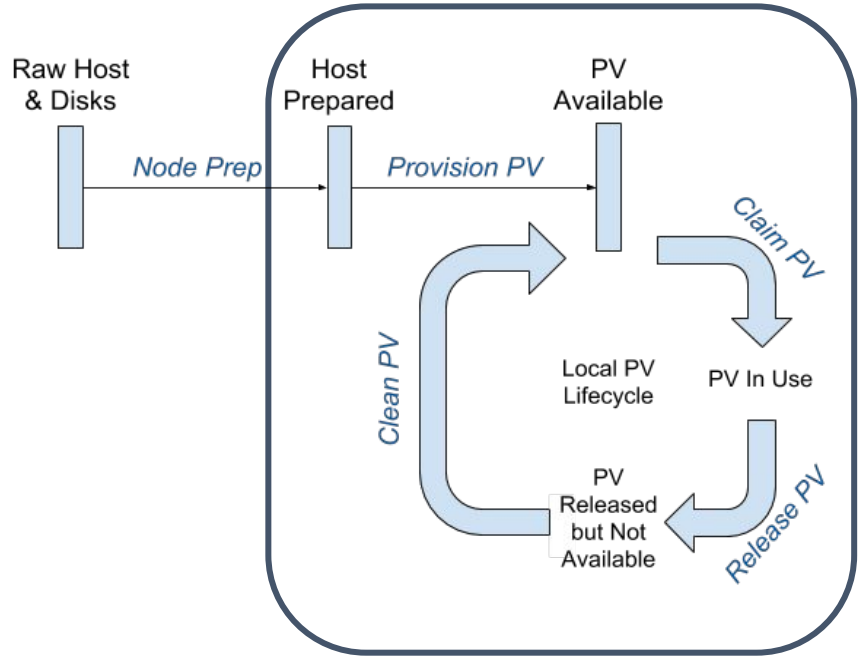
lv-provisioner daemonset

```
<snip>
spec:
  affinity:
    nodeAffinity:
      requiredDuringSchedulingIgnoredDuringExecution:
        nodeSelectorTerms:
          - matchExpressions:
          - key: storage.salesforce.com/nodeprep
          operator: In
            values:
            - mounted
```

Kubernetes PV Management

Open source LV provisioner that runs in any Kubernetes cluster
https://github.com/kubernetes-incubator/external-storage/tree/master/local-volume

1. Finds mount points under discovery directories
2. Creates local PVs
3. Workload consumes and releases PV
4. Volume data cleaned, and PV deleted
5. Repeat



Demo

KubeCon

CloudNativeCon

Europe 2018

Summary

Local disk administration is challenging, but can be automated

- Node prep automation, environment specific
- Static local PV provisioner

After environment is setup, local PVs are ready for consumption

- Same PVC/PV interface as remote storage
- Best with StatefulSets

Future Roadmap

Raw block volumes

- Alpha in Kubernetes 1.10 and works with LV provisioner v2.1.0
- Higher performance by bypassing FS
  - Small objects stored in a database
  - Example: Ceph Luminous Bluestore/Bluefs metadata

Dynamic provisioning with LVM

- Improved local disk utilization
- But performance penalty of shared disks

Handle FS formatting and mounting in Kubernetes

Documentation

This talk

https://speakerdeck.com/msau42

Kubernetes documentation

https://kubernetes.io/docs/concepts/storage/volumes/#local
https://github.com/kubernetes-incubator/external-storage/tree/master/local-volume

Blog posts

https://kubernetes.io/blog/2018/04/13/local-persistent-volumes-beta
https://medium.com/salesforce-engineering/provisioning-kubernetes-local-persistent-volumes-61a82d1d06b0

Get Involved!

Kubernetes Storage special interest group (SIG)

- Bi-monthly meetings Thursdays at 9 AM PST
- http://slack.k8s.io

Contact us with questions and feedback!

- Github, Slack: msau42 & ianchakeres
- Twitter: _msau42_