Felix Hupfeld and Jörg Schad

Apache Mesos Storage Now and Future

MESOSPHERE

Motivation

Why does storage matter?

- MESOS offers great support for stateless services

- But what about data persistence?
  - Distributed Databases
  - Distributed Filesystems
  - Docker Volumes on distributed storage
- Two perspectives:
  - Support for Distributed Storage Frameworks
  - Support for Frameworks using the Distributed Storage Frameworks

Motivation

Why does this presentation matter?

- Mesos storage is an evolving topic

- **Now**
  - What can you do right now?
  - What are others doing right now?
- **Future**
  - What can you do in the future?
  - What do you want to be able to do in the future?

Now

Now

Mesos and Storage

- **HDFS**: works… kind of :-)
  - Storage drivers hacks
  - HDFS storage managed outside Mesos
  - Agent failover
- Cassandra
- ArangoDB
- …

- Disk Resources and Isolation
  - Use it!

Now

Persistence Primitives

My Task just died on the Mesos Agent*

- Problem: No guarantees for reoffered resources
  - Dynamic Reservations (MESOS-2018)

- Problem: Task's sandbox is garbage collected
  - Persistent Volumes (MESOS-1554)

Now

External Volumes

- Storage backed by third party storage services
  - E.g. EMC ScaleIO, EC2, NFS based storage system
  - Not Mesos Managed (≠remote storage)
  - Not tied to particular agent

How can I access those volumes (esp. without docker)?

Now

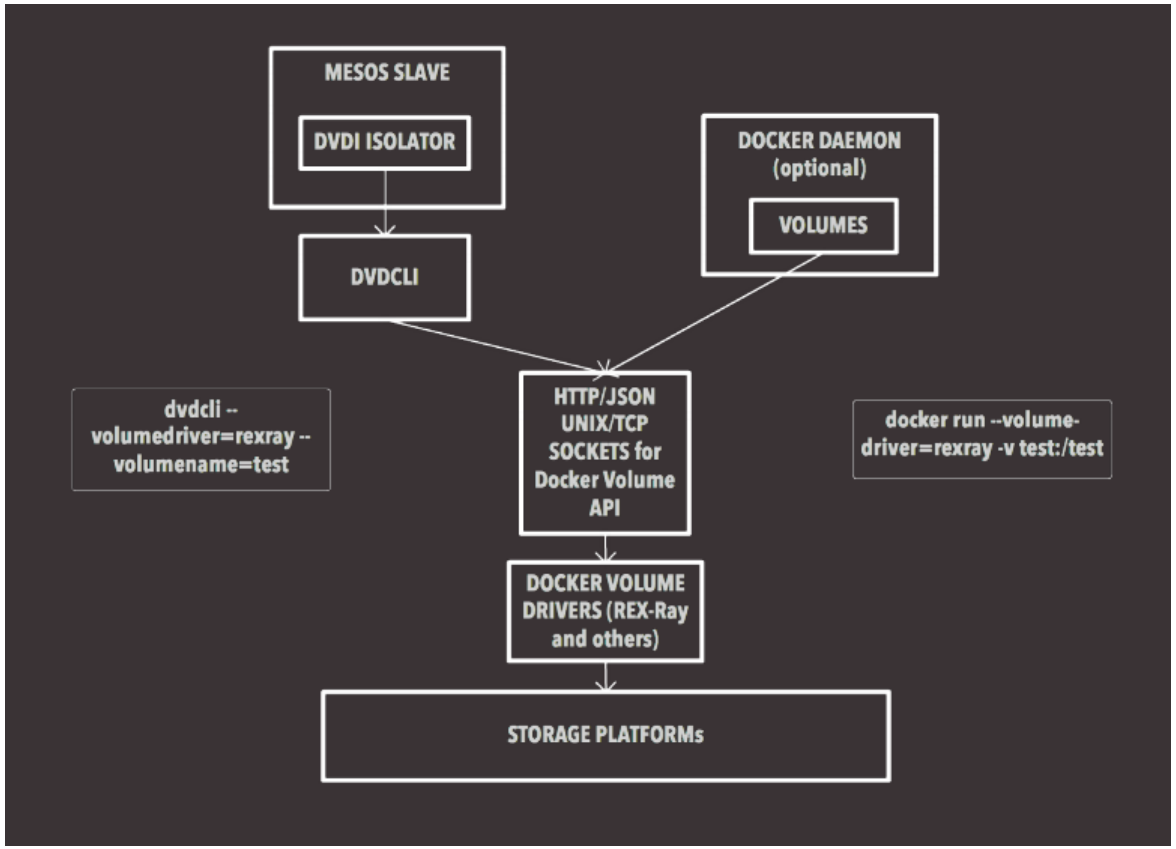
Docker Volume Driver Isolator Module

- Create/mount external volumes at task startup
- Exposes existing Docker Volume Driver to non-docker tasks
  - e.g. RexRay

```
{
  "id": "my-marathon-app",
  "cmd": "while [ true ] ; do touch /var/lib/rexray/volumes/test12345/hello ; sleep 5 ;
done",
  …,
  "env": {
    "DVDI_VOLUME_NAME": "test12345",
    "DVDI_VOLUME_OPTS": "size=5,iops=150,volumetype=io1,newfstype=xfs,
overwritefs=true"
  }
}
```

Now

Docker Volume Driver Isolator Module



Demo

Now

Quobyte

Now

Quobyte Mesos Framework

Current framework features:

- Deploys Quobyte on a Mesos cluster
- Auto-detects Quobyte devices and schedules corresponding services (registry, data, metadata)
- Uses Mesos-DNS SRV records for discovery of Quobyte registry
- Rolling deployment of new Quobyte releases

Open:

- Declarative device management: via Persistent Volumes?

Now

Quobyte Fault-tolerance Demo

Host 1

MySQL Container

Quobyte Storage

Host 2

Wordpress Container

Quobyte Storage

Host 2

Quobyte Storage

Demo

Now

Quobyte's Wishlist

Interface to applications:

- Specify use of dynamically mounted volumes (auto-mount and bind)
  "quobyte:/volumeA/"
- Specify QoS demands (pass through demands to storage system)

Interface to frameworks

- Back-channel of locality information to frameworks
  (task wants to access /volumeA/app1, where shall I schedule it)

Future

Future

DFS as Mesos Managed Resource

At the moment DFS managed out of band..

Who is allowed to access which filesystem?

Data Locality based scheduling for frameworks?

Future

Clusterwide Resources (MESOS-2728)

DFS is a resource not tied to a particular agent

- Isolation?
- Zombie Tasks
- Offered by …?

Still client agent requires Storage Drivers

```
{
  "id": "/product/service/myApp",
  …,
  "uris": [ "hdfs://namenode/mydep",
  "quobyte://registry/dep2"]
}
```

Future

Storage Discovery

My Mesos Cluster runs multiple HDFS, Quobyte, and Ceph. How do I discover and address each of them from within my tasks?

- Fstab like Mount Tables
  - Fixed Mount Point in universal Mesos namespace
- Mesos-DNS for Metadata Server

Thank you!

- Questions?
- Feedback?
- Further Wishes?

Feel feel to comment on Jira(s)!