CLOUD NATIVE COMPUTING FOUNDATION

Webinar Series

Cloud Native Storage

May 17, 2017

Your Presenters

Mark Balch

VP of Products and
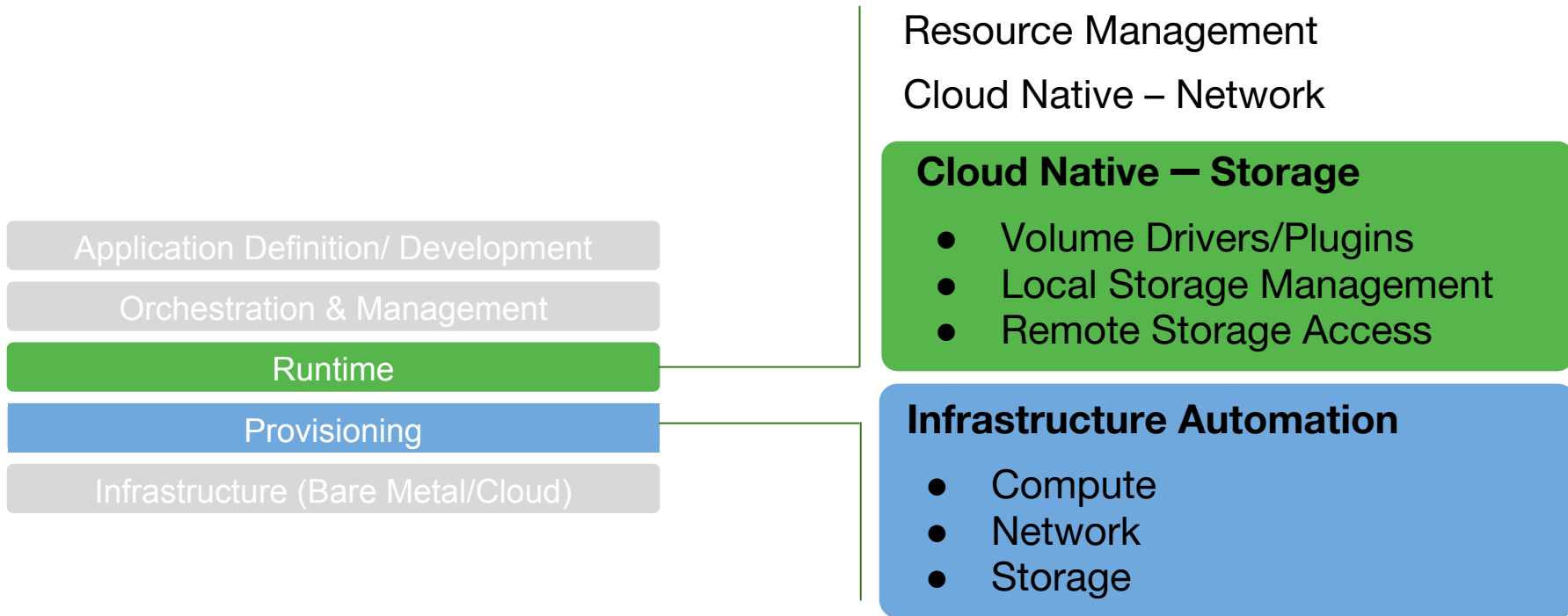Marketing, Diamanti
@markbalch

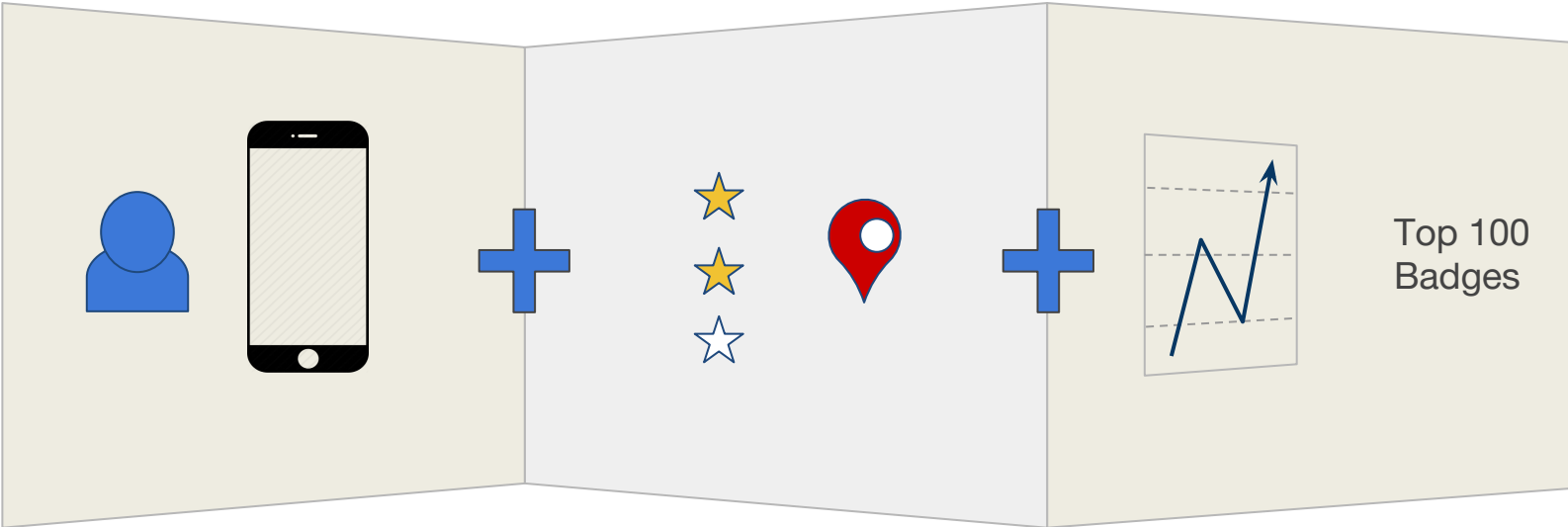**Eric Han**

VP of Product Management,
Portworx
@eric7han

Clint Kitson

Technical Director,
{code} by Dell EMC
@clintkitson

CLOUD NATIVE
COMPUTING FOUNDATION

Storage in CNCF Reference Architecture

Resource Management

Cloud Native – Network

Cloud Native — Storage

- Volume Drivers/Plugins
- Local Storage Management
- Remote Storage Access

Application Definition/ Development

Orchestration & Management

Runtime

Provisioning

Infrastructure (Bare Metal/Cloud)

Infrastructure Automation

- Compute
- Network
- Storage
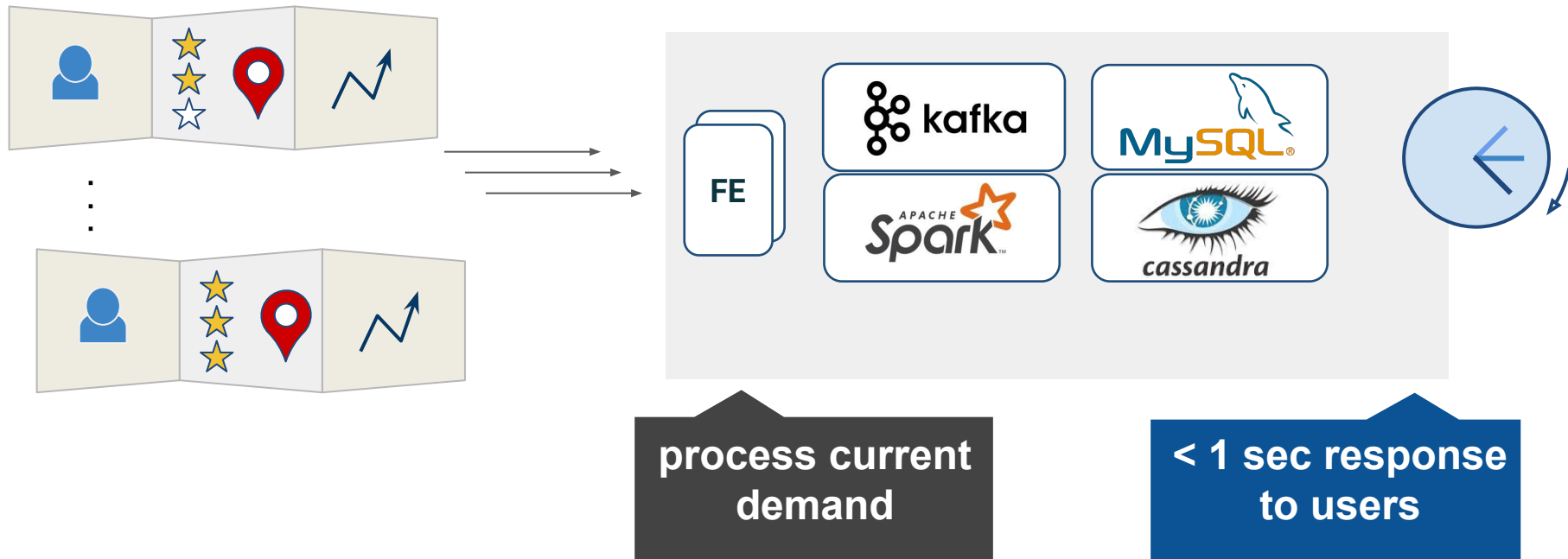
CLOUD NATIVE
COMPUTING FOUNDATION

Millions of Events

Check-ins + Photos   Ratings + Location   Trends + Stats

Top 100 Badges

Cloud Native Patterns with Storage

process current demand

< 1 sec response to users

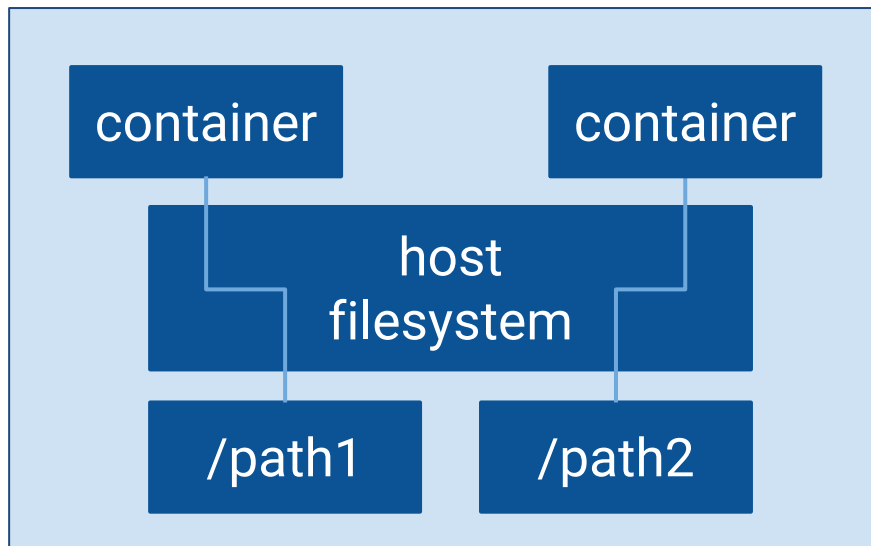CLOUD NATIVE
COMPUTING FOUNDATION

Before: nighttime batch processing
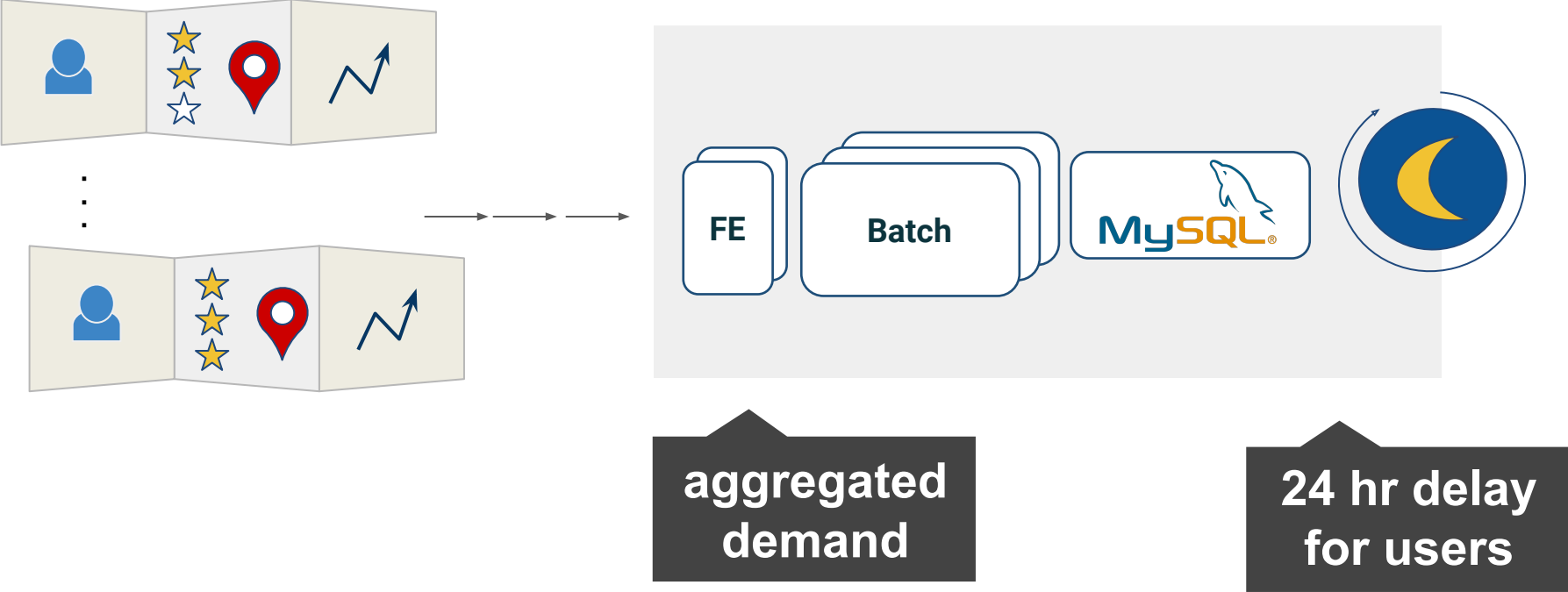
First Iteration of Container Storage: Host Mount
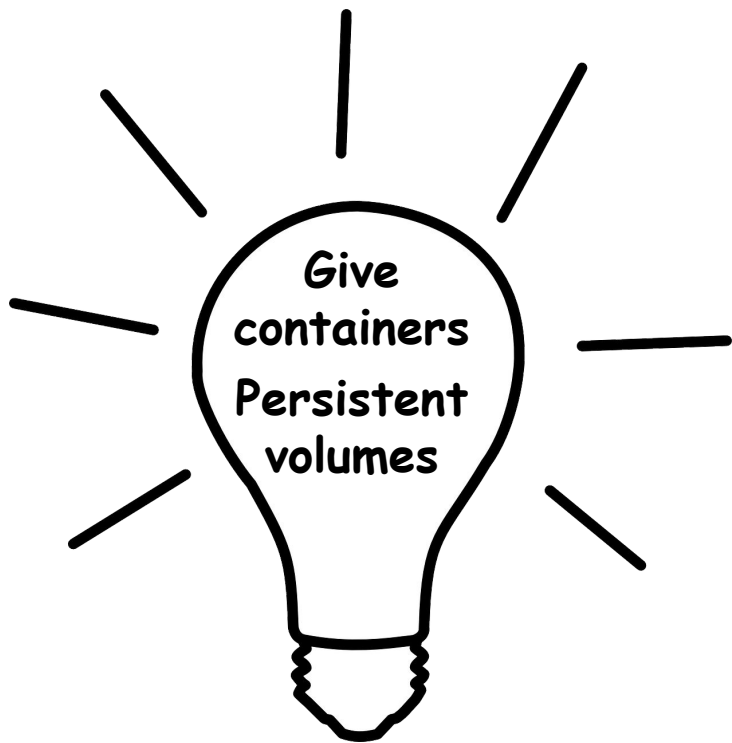
`/host/pathA:/container/path1`   `/host/pathB:/container/path2`
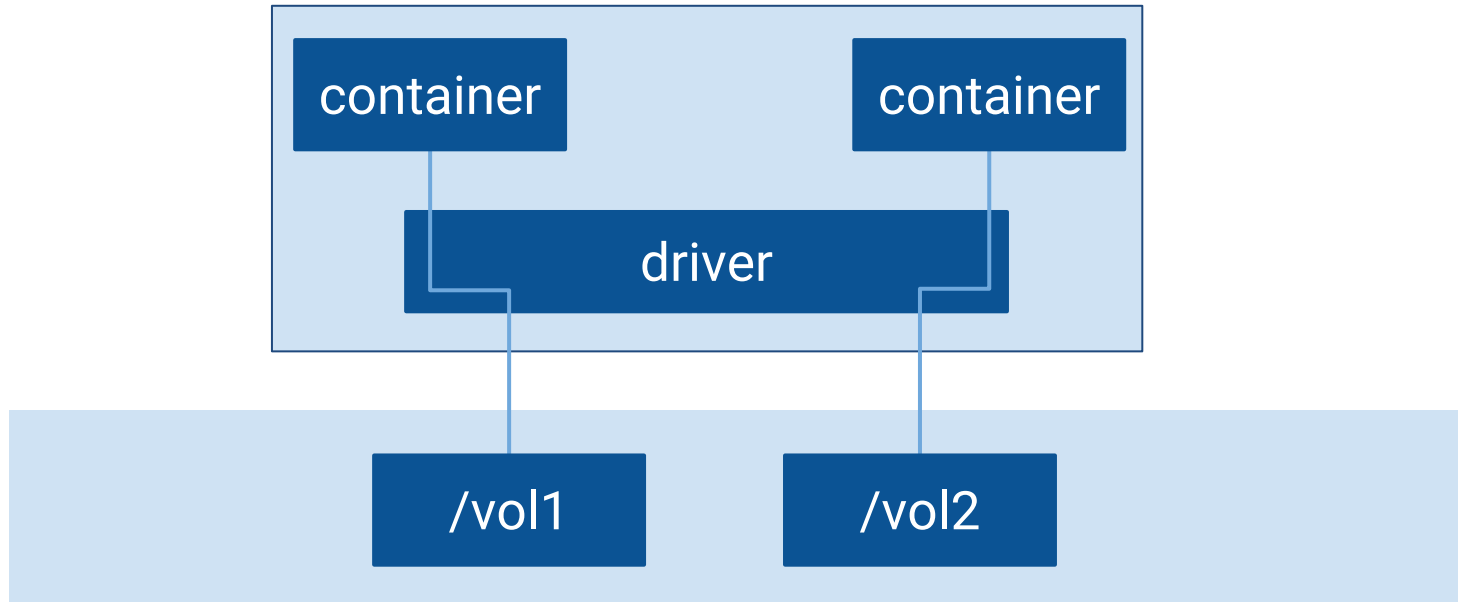

Kinda works... but ...
- No persistence beyond host lifetime (above)
- Prohibitive to stateful workloads (availability, reuse, …)
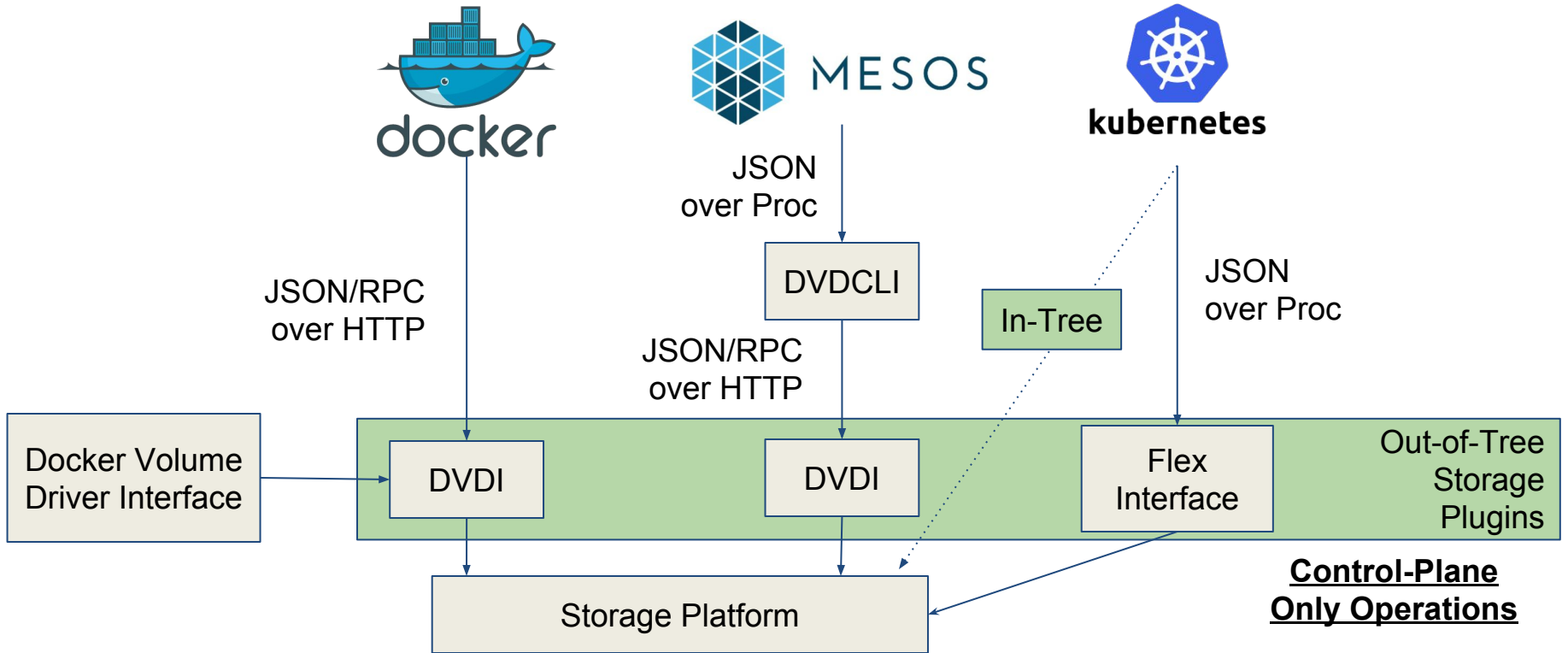
Enter Cloud Native Storage ...

Give containers

Persistent volumes

CLOUD NATIVE
COMPUTING FOUNDATION

Give containers persistent volumes

CLOUD NATIVE
COMPUTING FOUNDATION

Storage interoperability today

Container persistent volumes

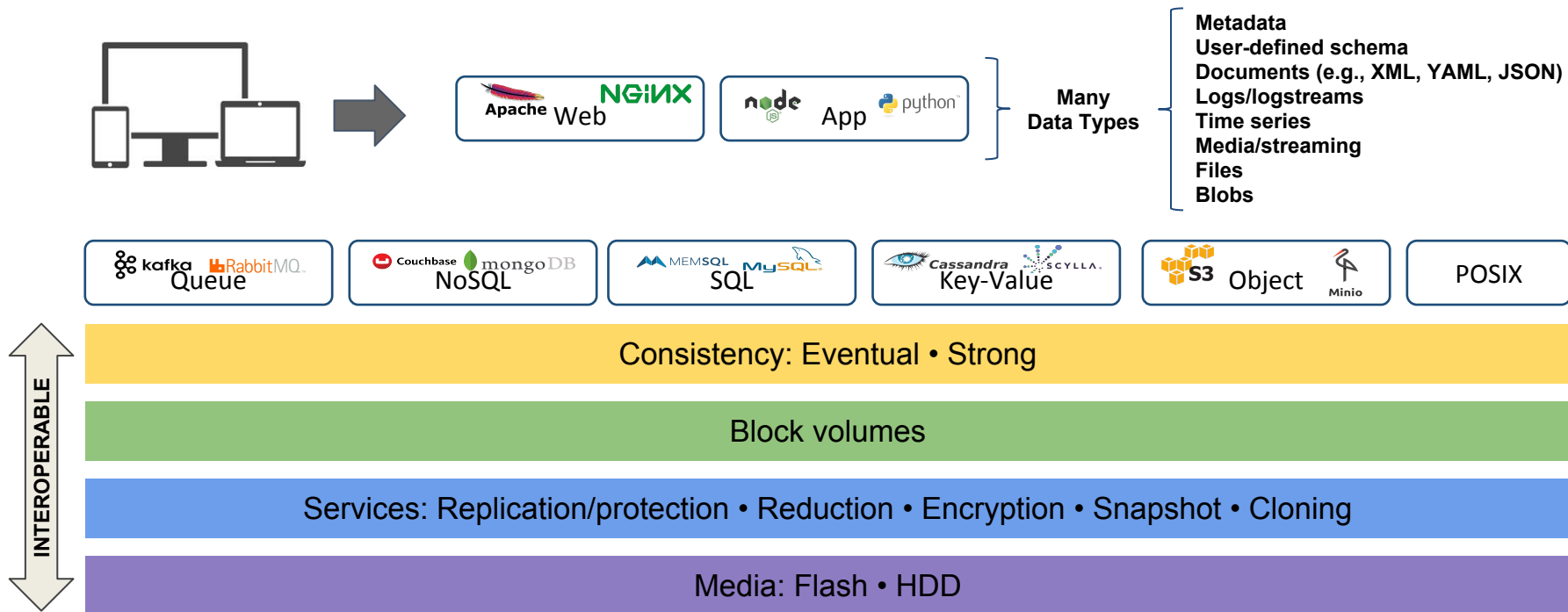**Persistent Volumes**
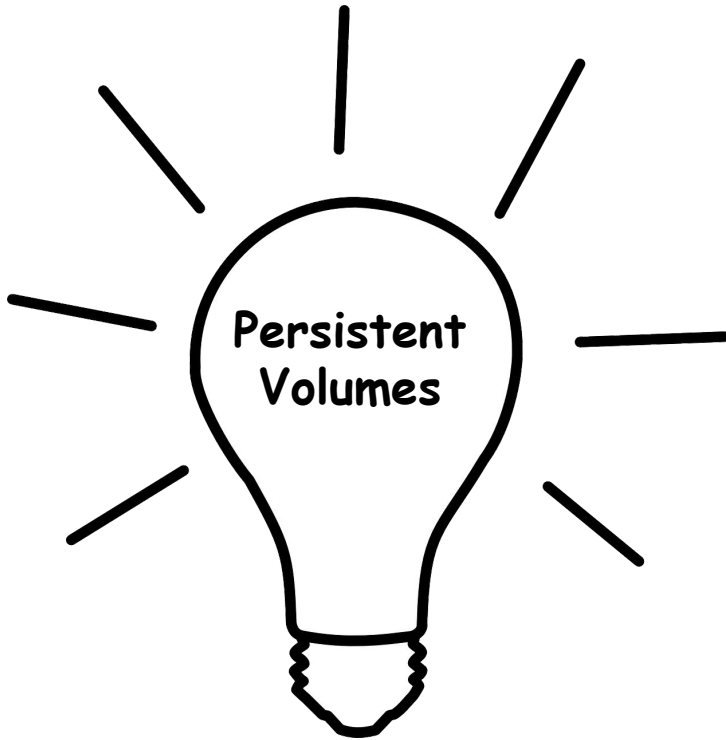
✓ Data survives (persists) beyond container, pod, host

✓ Workload can choose its best storage

✓ Kubernetes, Docker, and Mesos took this approach

✓ We know this is a portable pattern

CLOUD NATIVE
COMPUTING FOUNDATION

Cloud native storage takes many forms

Many Data Types
- Metadata
- User-defined schema
- Documents (e.g., XML, YAML, JSON)
- Logs/logstreams
- Time series
- Media/streaming
- Files
- Blobs

| kafka RabbitMQ Queue | Couchbase mongoDB NoSQL | MEMSQL MySQL SQL | Cassandra SCYLLA Key-Value | S3 Object Minio | POSIX |

INTEROPERABLE

Consistency: Eventual • Strong

Block volumes

Services: Replication/protection • Reduction • Encryption • Snapshot • Cloning

Media: Flash • HDD

CLOUD NATIVE
COMPUTING FOUNDATION

Cloud native storage lets users choose

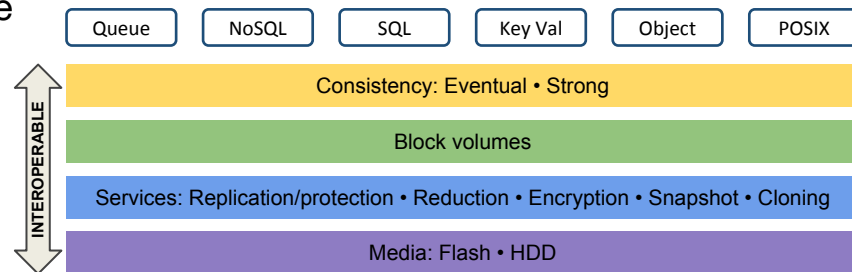Wide range of storage services, speed, cost

Differing storage capabilities according to workload

- eCommerce transactions: SQL with strong consistency and replication
- Product photos: low-cost object storage with caching layer
- Recommendation engine: queue and NoSQL on flash volumes

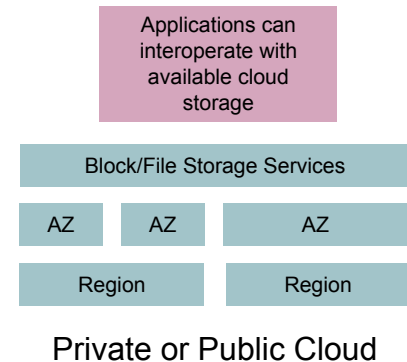Automates and abstracts underlying storage infrastructure

- Interfaces to container runtime and orchestration
- Pools capacity in multi-tenant environment
  - Locally, globally
- Meters and enforces quotas
- Ensures performance for each workload
  - QoS, IOPS, latency...

Metadata
User-defined schema
Documents
Logs/logstreams
Time series
Media/streaming
Files
Blobs

Queue	NoSQL	SQL	Key Val	Object	POSIX

INTEROPERABLE

Consistency: Eventual • Strong

Block volumes

Services: Replication/protection • Reduction • Encryption • Snapshot • Cloning
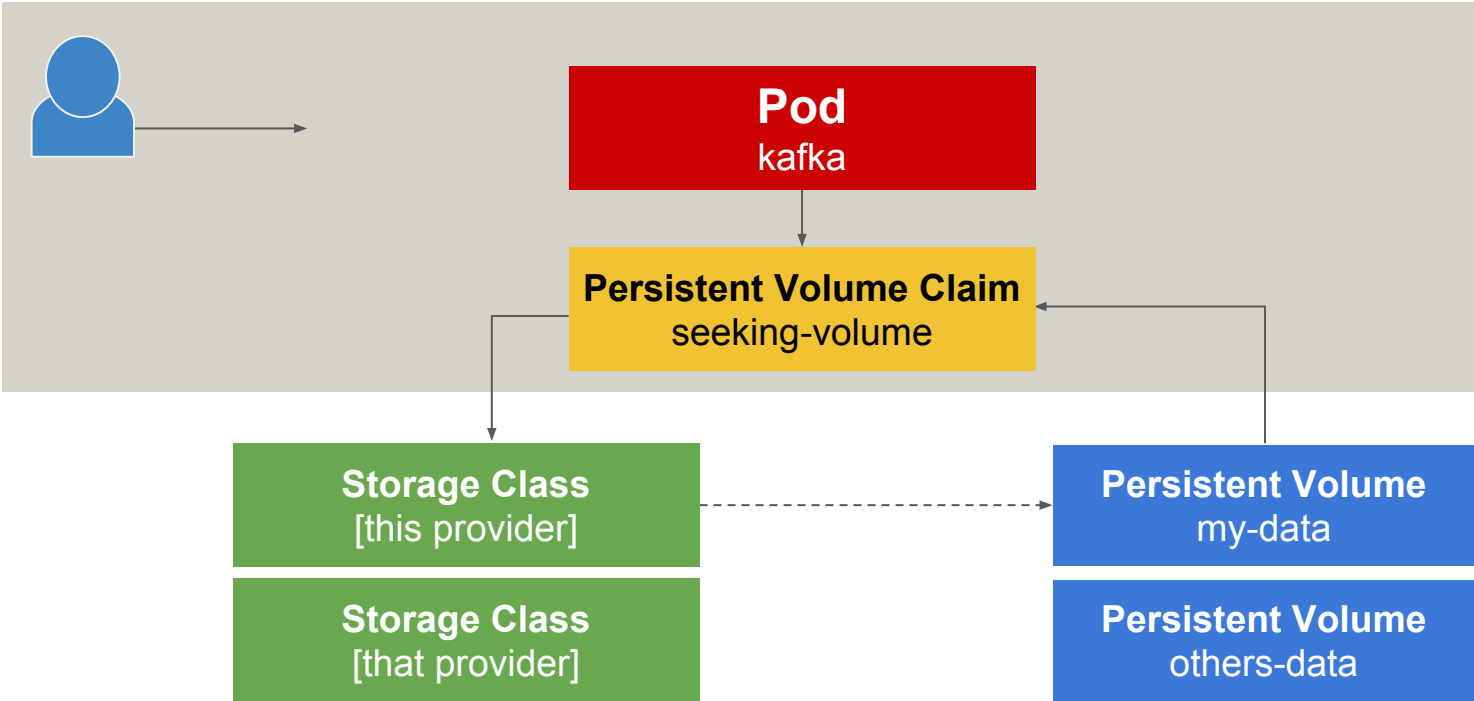
Media: Flash • HDD

Do I have cloud native storage?

- Qualities to look for in available storage resources
  - Interoperates with container orchestrators and runtimes
  - Common abstraction of core capabilities (size, type, IOPS…)
  - Common data services (snapshots, replication, encryption, …)
  - Role-based access control
  - Elasticity of capacity and performance
  - Lifecycle and operations can be automated
- Things to stay away from
  - Hardware-defined provisioning and configuration
  - Slow control-plane orchestration
  - Complex availability
- Do I need something different?
  - Application requirements may be stringent
  - Running bare-metal and shared storage is not available
  - Portability of storage services

Applications can interoperate with available cloud storage

Block/File Storage Services

| AZ | AZ | AZ |

| Region | Region |

Private or Public Cloud

Persistence + Portability

CLOUD NATIVE
COMPUTING FOUNDATION

Pod to Persistent Volume

Pod

```
kind: Pod
metadata:
  name: mypod
spec:
  containers:
    - name: kafka
      image:kafka
      volumeMounts:
      - mountPath: /var/lib/kafka
        name: mypd
  volumes:
    - name: kafka-data
      persistentVolumeClaim:
        claimName: seeking-volume
```

Persistent Volume Claim

```
kind: PersistentVolumeClaim
metadata:
  name: seeking-volume
spec:
  resources:
    requests:
      storage: 8Gi
  storageClassName: fast
  selector:
    matchLabels:
      release: stable
```

Persistent Volume

```
kind: PersistentVolume
metadata:
  name: ebs-disk-1
spec:
  capacity:
    storage: 10Gi
  awsElasticBlockStorage:
    fsType: ext4
    pdName: aws-ebs-1
```

Storage Class

```
kind: StorageClass
metadata:
  name: fast
provisioner: [provider]
```
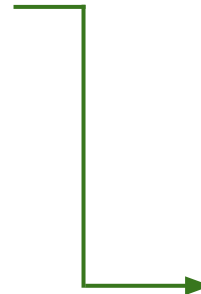
1

2

4

CLOUD NATIVE COMPUTING FOUNDATION

Cloud Native Computing Foundation

Thank You