Spark on Kubernetes Best Practice

Junjie Chen, Jerry Shao

Tencent Cloud

About US

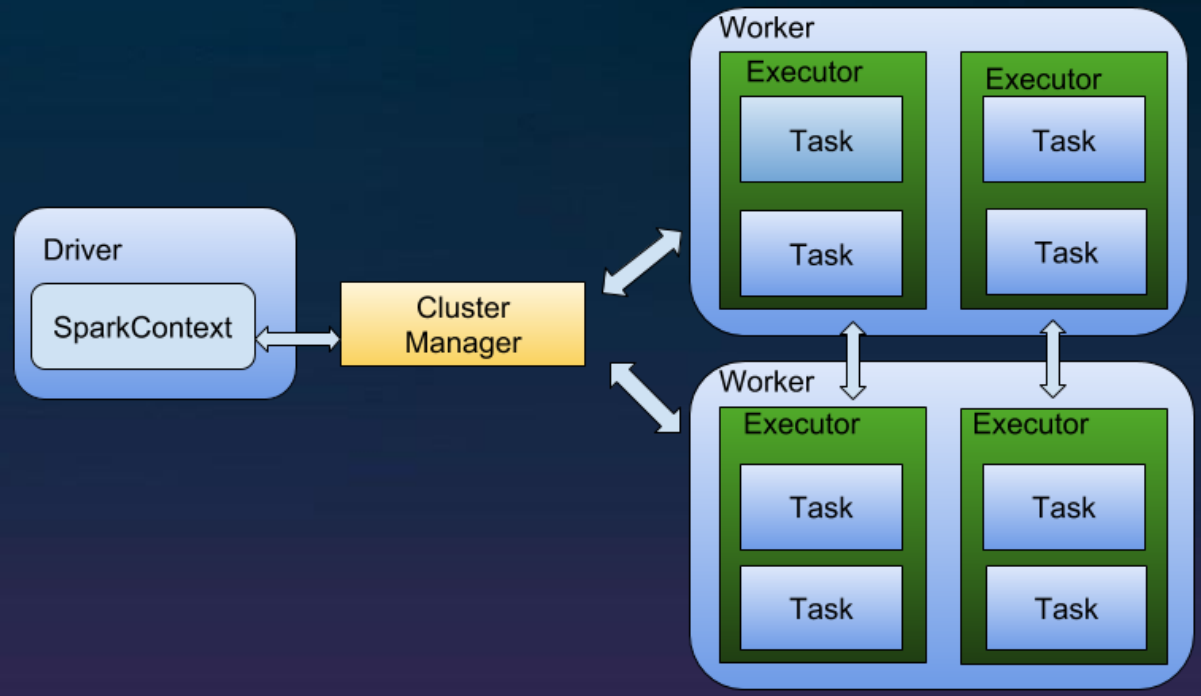
- Junjie Chen
  - Senior Software Engineer at Tencent. Focus on big data and cloud area over years, past work experience at Intel.
- Jerry Shao
  - Expert Software Engineer at Tencent, Apache Spark Committer. Focus on open source Big Data area, past work experience at Hortonworks, Intel.

About Spark

A unified analytics engine for large-scale data processing.

- DAG based task scheduler
- Explicit Cache API & In-memory computation
- Catalyst as the optimizer & Tungsten project for native execution acceleration

Execution Model



Spark Cluster Manager

Apache Mesos	Spark Standalone	Apache Hadoop Yarn
• Spark support Apache Mesos in early stage	• Lightweight built-in cluster manager	• Comes from Hadoop 2.0

Spark Cluster Manager

Apache Mesos	Spark Standalone	Apache Hadoop Yarn	Kubernetes
• **Spark support Apache Mesos in early stage**	• **Lightweight built-in cluster manager**	• **Comes from Hadoop 2.0**	• Start from Spark 2.3.0

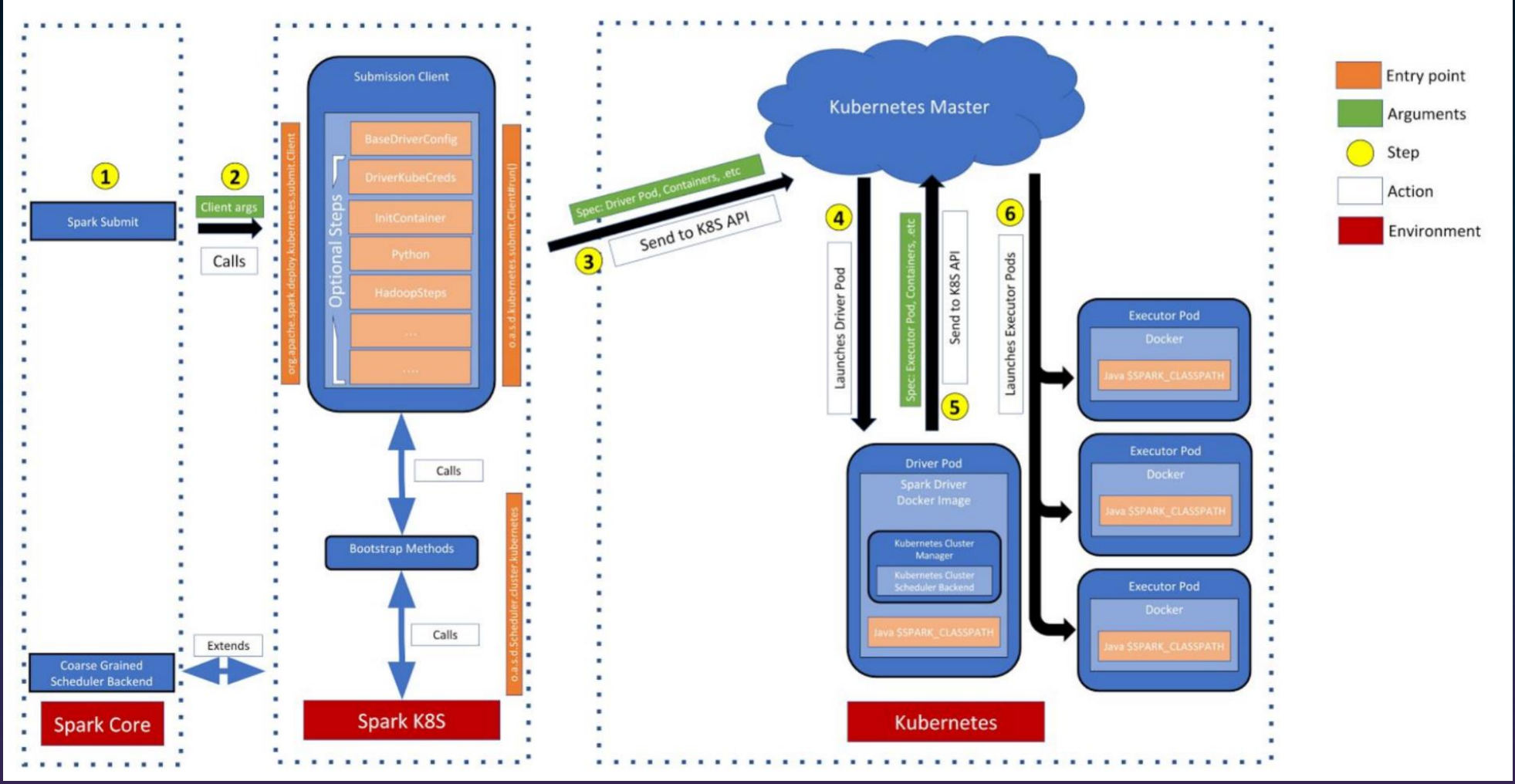
Why Kubernetes

- Kubernetes advantages
  - Portability of Docker
  - Out of box management support, namespace, RBAC, authentication, logging, etc..
  - Docker ecosystem and large OSS community

Deployment Consideration

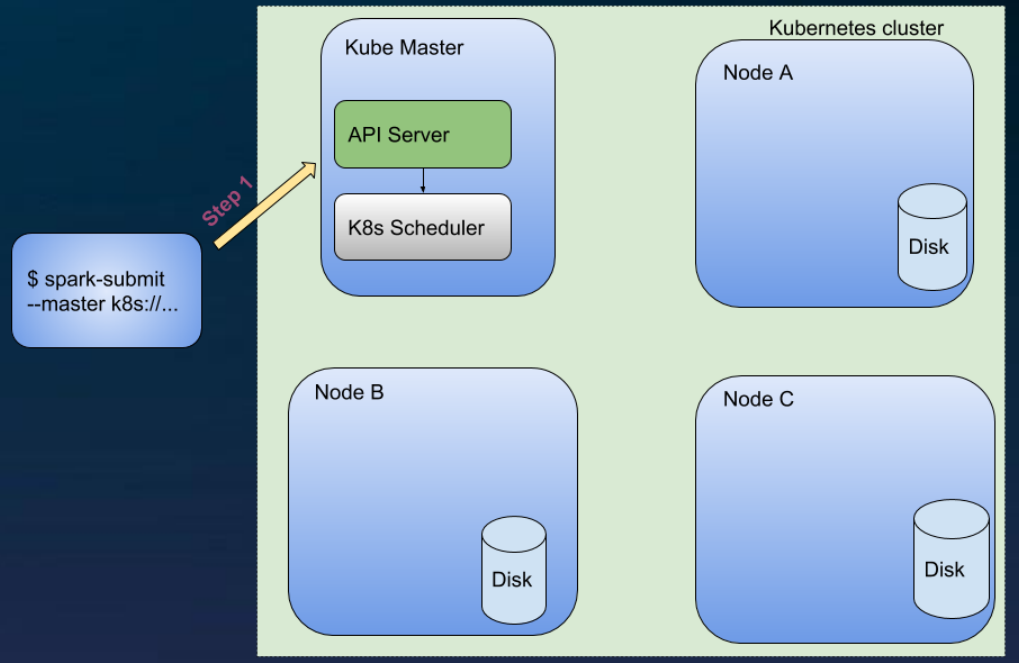
- **Heterogeneous Deployment**
- **Serverless**

Spark on Kubernetes Architecture



Spark on Kubernetes Detail

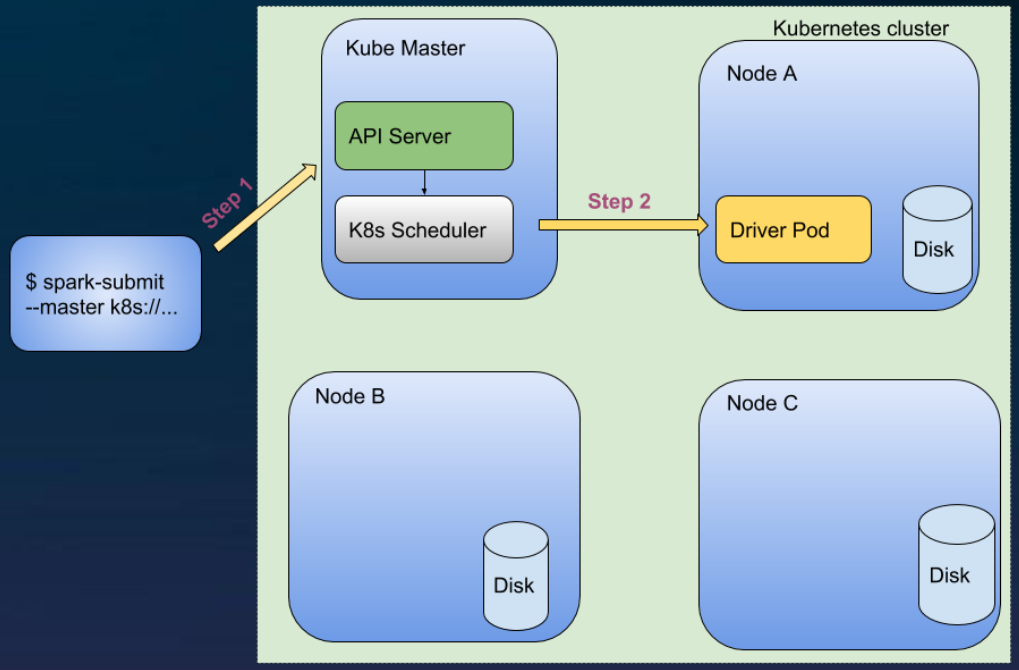
Step 1: Submit application through spark submit



Spark on Kubernetes Detail

Step 1: Submit application through spark submit

Step 2: Scheduler allocate driver pod

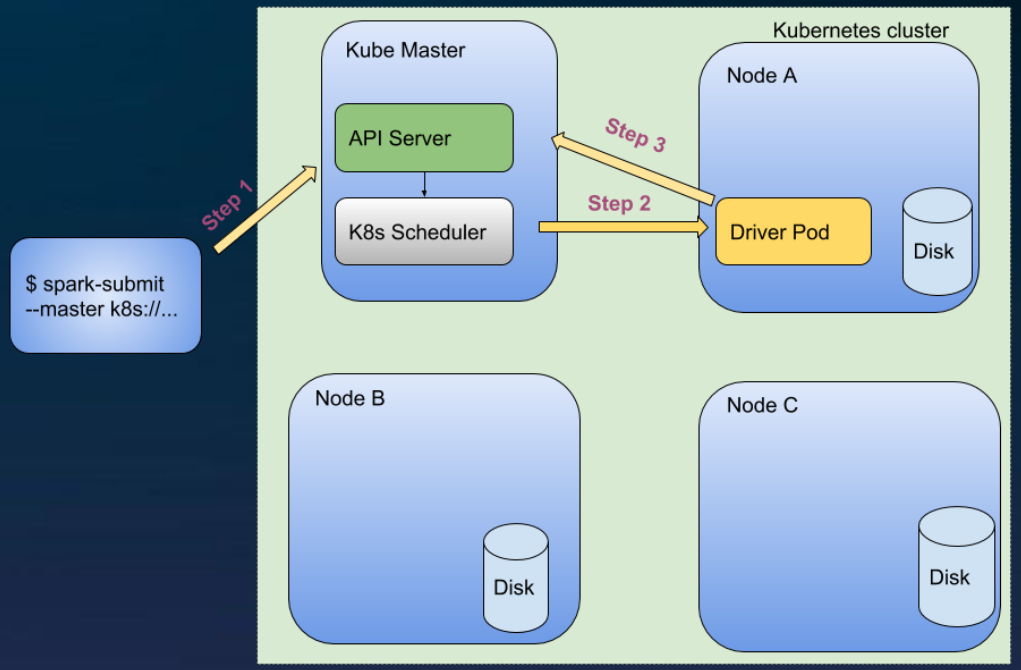


Spark on Kubernetes Detail

Step 1: Submit application through spark submit

Step 2: Scheduler allocate driver pod

Step 3:Driver Pod ask k8s scheduler to allocate executor pods



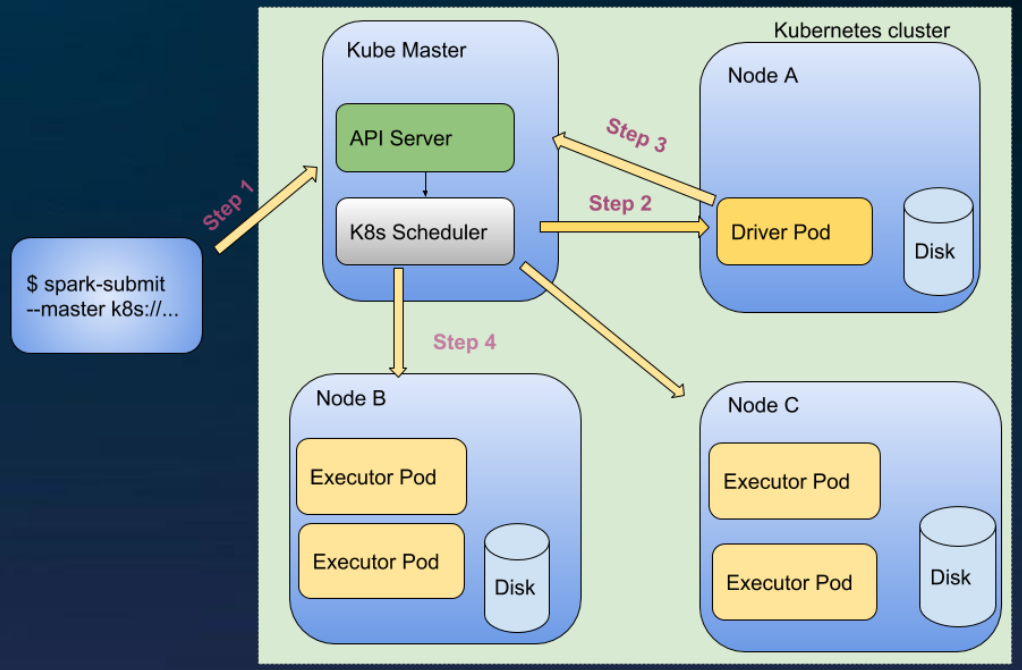
Spark on Kubernetes Detail

Step 1: Submit application through spark submit

Step 2: Scheduler allocate driver pod

Step 3:Driver Pod ask k8s scheduler to allocate executor pods

Step4: Executor pods execute tasks



How to run

```
bin/spark-submit \
    --master k8s://https://<k8s-apiserver-host>:<k8s-apiserver-port> \
    --deploy-mode cluster \
    --name spark-pi \
    --class org.apache.spark.examples.SparkPi \
    --conf spark.executor.instances=5 \
    --conf spark.kubernetes.container.image=<spark-image> \
    local:///path/to/examples.jar
```

Spark on k8s Status

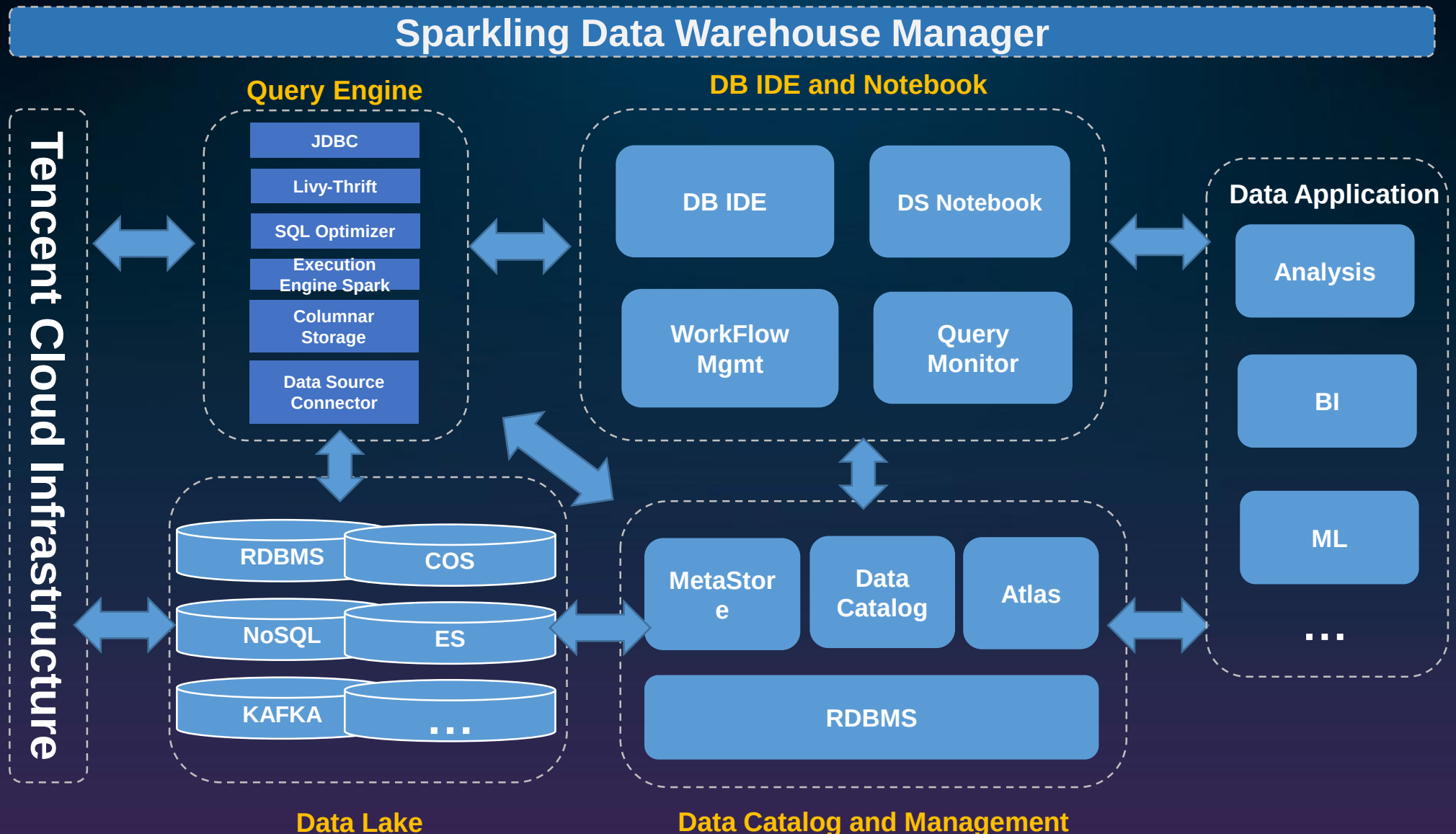
- Done in 2.4
  - Cluster/Client mode spark submit
  - Static allocation
  - Staging server for dependencies management
  - Java, Scala, Python, R.
- Working in Progress
  - Dynamic allocation
  - External shuffle service
  - Security (Kerberos)

Use Case

Sparkling

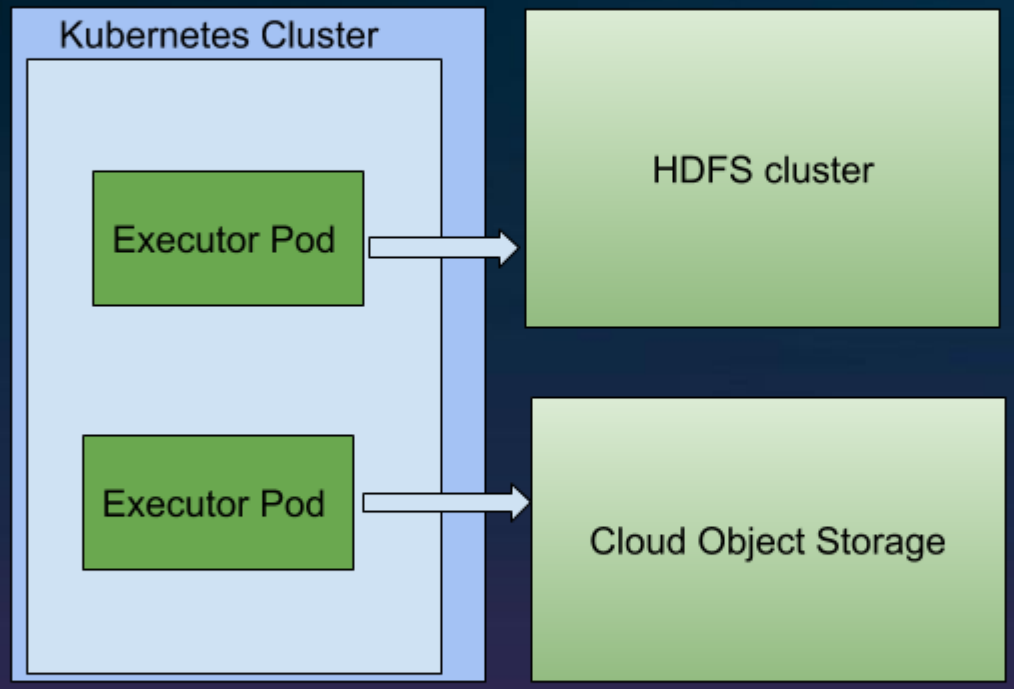
- A PB scale EDW with benefits of fast deployment, resource elasticity, high performance and cost-effective.

Architecture Overview

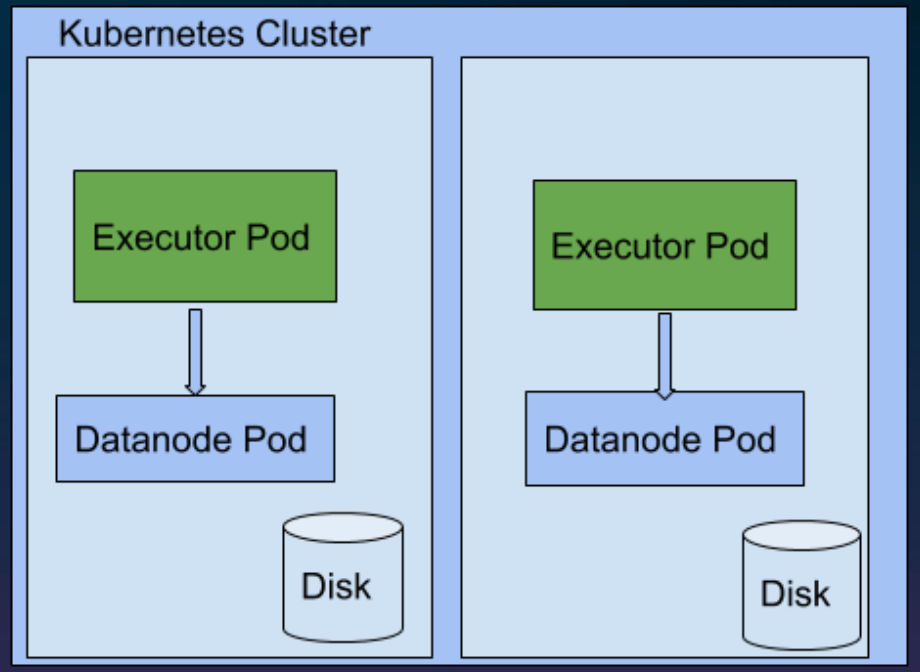


Data Process Model

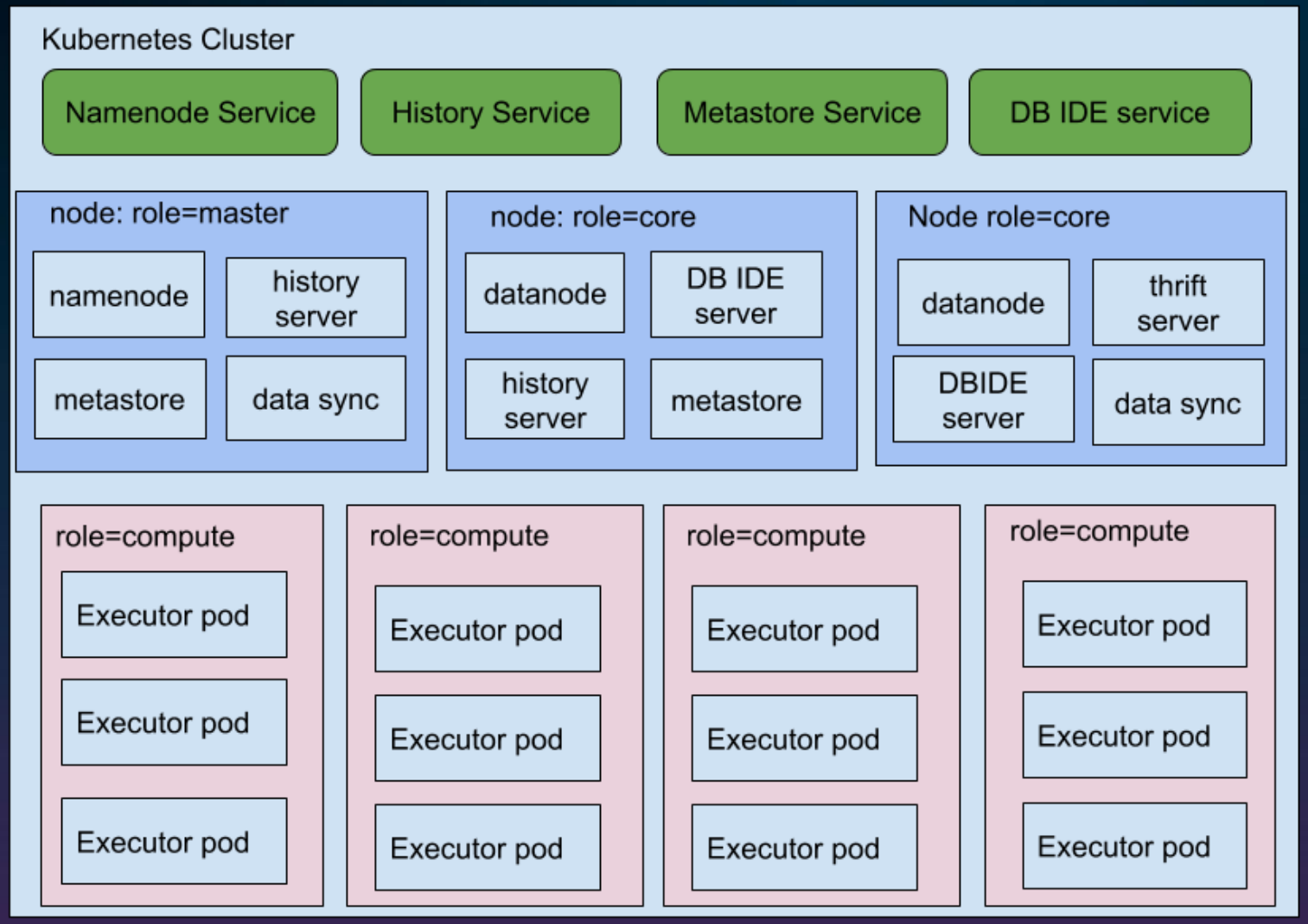
- Storage outside cluster
  Actual use cases



- Storage inside cluster
  Benefit from Data Locality



Sparkling on Kubernetes



Deploy HDFS

- Use same image for Namenode, Datanode.
- Identify pod type through docker env.
- Define Namenode as NodePort service.

```yaml
---
apiVersion: v1
kind: Service
metadata:
    name: k8s-hadoop-master
spec:
    type: NodePort
    selector:
        app: k8s-hadoop-master
    ports:
        - name: rpc
          port: 9000
          targetPort: 9000
        - name: http
          port: 50070
          targetPort: 50070
          nodePort: 32007
```

Deploy HDFS Cont.

- Namenode

```
spec:
    containers:
        - name: k8s-hadoop-master
          image: sherrytima/hadoop:v4
          imagePullPolicy: IfNotPresent
          ports:
            - containerPort: 9000
            - containerPort: 50070
          env:
            - name: HADOOP_NODE_TYPE
              value: namenode
            - name: HDFS_MASTER_SERVICE
              valueFrom:
                    configMapKeyRef:
                        name: ku8-hadoop-conf
                        key: HDFS_MASTER_SERVICE
          resources:
    restartPolicy: Always
    nodeSelector:
       ROLE: master
```

- Datanode

```
spec:
    containers:
        - name: hadoop-datanode-1
          image: sherrytima/hadoop:v4
          imagePullPolicy: IfNotPresent
          env:
            - name: HADOOP_NODE_TYPE
              value: datanode
          volumeMounts:
            - mountPath: /mnt
              name: data-volume
    volumes:
        - name: data-volume
          hostPath:
            path: /mnt
    nodeSelector:
        HDFS: datanode1
    restartPolicy: Always
```

Other components

- Deploy as NodePort services

```yaml
---
apiVersion: v1
kind: Service
metadata:
    name: spark-history-server
spec:
    type: NodePort
    selector:
        app: spark-history-server
    ports:
        - name: http
          port: 18080
          targetPort: 18080
          nodePort: 30080
```

```yaml
apiVersion: v1
kind: Service
metadata:
  name: metastore-service
  labels:
    name: metastore-service
spec:
  ports:
  - port: 9083
    targetPort: 9083
  selector:
    app: hive-server
```

Services finding in pods

- Executor and Driver Pod could find services through
  - Environment
  - k8s-dns : http://k8s-hadoop-master.default:50070

```
SPARK_HISTORY_SERVER_SERVICE_PORT_HTTP=18080
SPARK_HISTORY_SERVER_SERVICE_HOST=192.168.255.242
K8S_HADOOP_MASTER_SERVICE_PORT_HTTP=50070
KUBERNETES_SERVICE_PORT_HTTPS=443
K8S_HADOOP_MASTER_SERVICE_PORT=9000
K8S_HADOOP_MASTER_SERVICE_PORT_RPC=9000
SPARK_HISTORY_SERVER_SERVICE_PORT=18080
KUBERNETES_SERVICE_PORT=443
K8S_HADOOP_MASTER_SERVICE_HOST=192.168.255.45
KUBERNETES_SERVICE_HOST=192.168.255.1
bash-4.4#
```

Performance

Environment

- Spark on K8S env1
  - Allocated through TKE
  - 10 x (32core + 128G + 4T HDD x 12)
- Spark on Yarn env1
  - Allocated through CVM
  - 6 x (56 core + 224G + 4T HDD x 12)
- Spark on K8S env2
  - Allocated through TKE
  - 10 x (16core + 128G + 200G SSDx1)
- Spark on Yarn env2
  - Allocated through CVM
  - 10 x (16core + 128G + 200G SSDx1)

Workload

- TPD-DS
  - The TPC Benchmark DS is a decision support benchmark.
  - It includes 99 queries of statistic, report, OLAP, data mine
  - Skew data, close to real scenario.
- Scale
  - 1T

Configurations

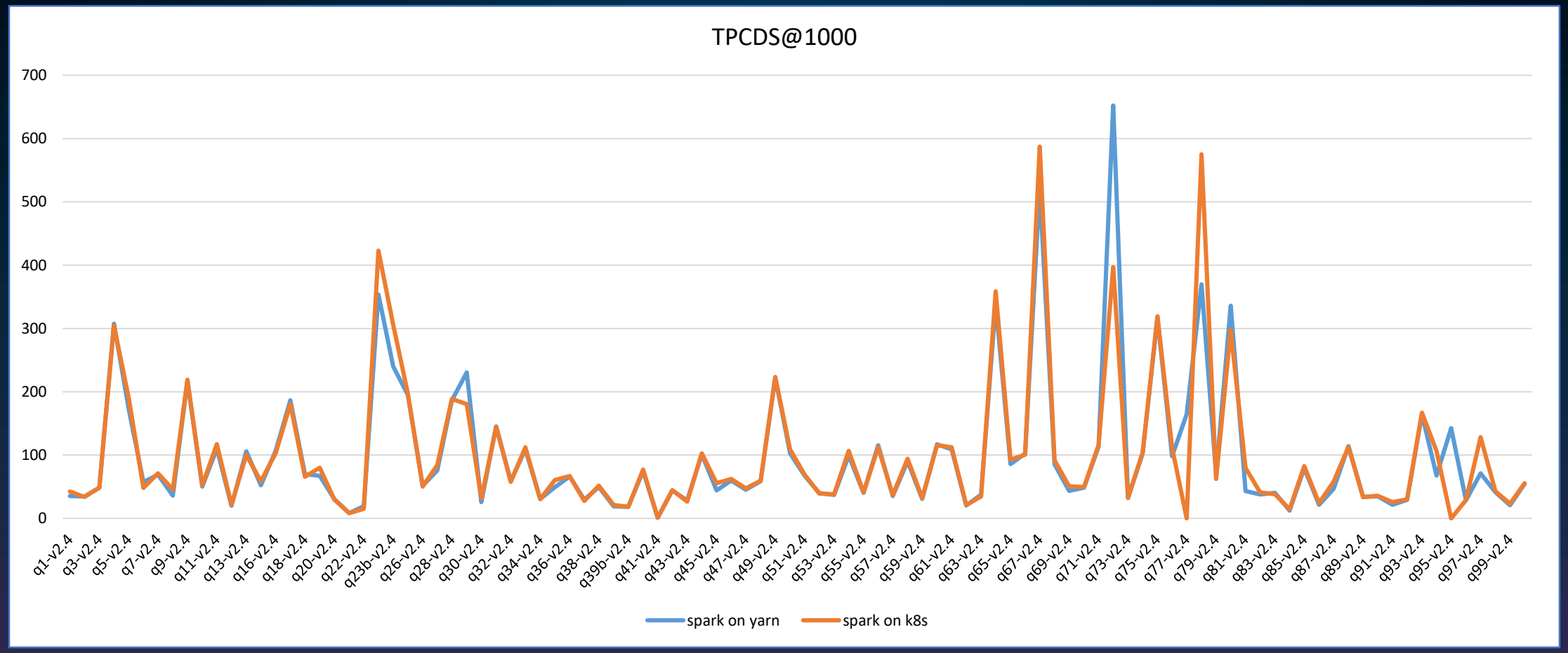
- Common configuration

key	Value
spark.driver.memory	16G
spark.executor.memory	6G
spark.executor.cores	1
spark.executor.memoryOverhead	1G
spark.sql.shuffle.partition	1024
spark.sql.autoBroadcastJoinThreshold	32m
spark.executor.instances	150

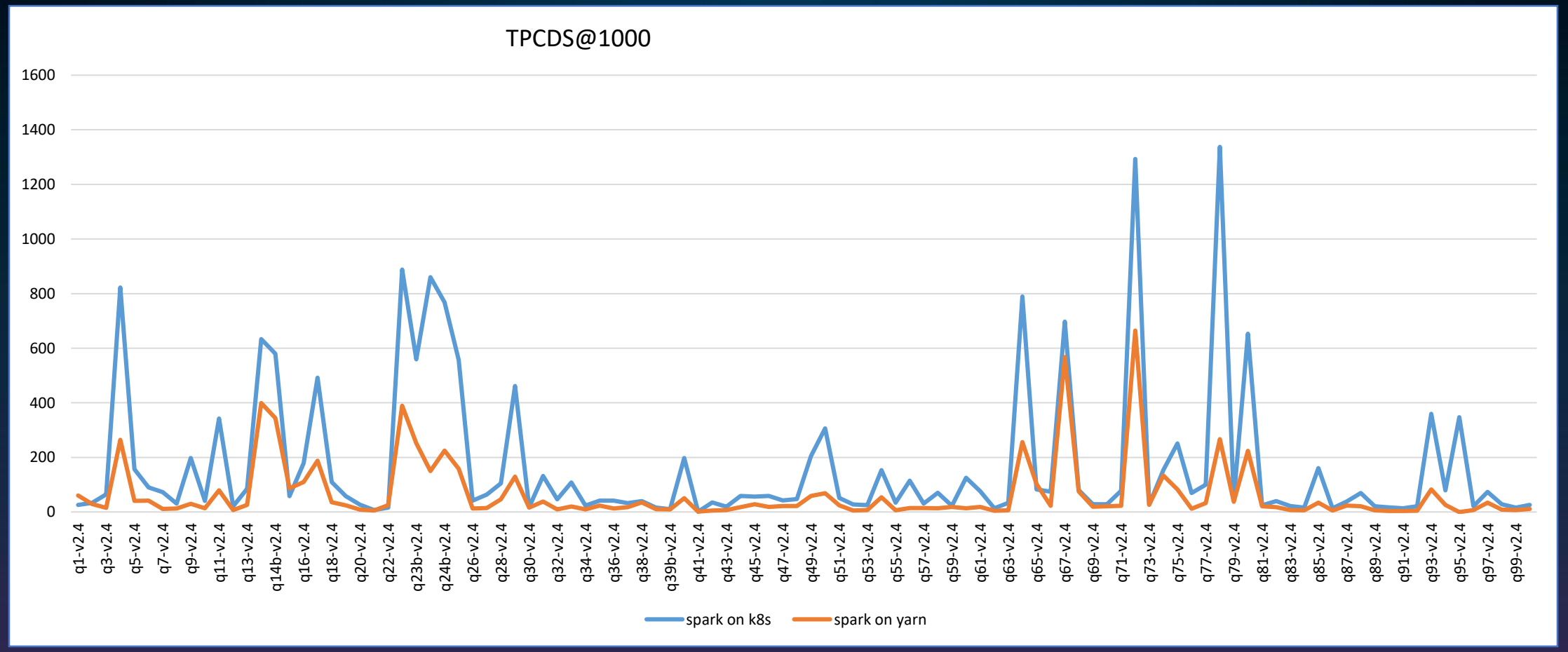
- Kubernetes specific configuration

Key	Value
spark.kubernetes.allocation.batch.size	5
spark.kubenetes.driver.limit.cores	1
spark.kubernetes.executor.limit.cores	1
spark.kubernetes.node.selector.executorkey	compute
spark.kubernetes.authenticate.driver.serviceAccountName	spark
spark.kubernetes.driver.volumes.hostpath.result.mount.path	/mnt
spark.kubernetes.executor.volumes.hostpath.shuffle.mount.path	/mnt

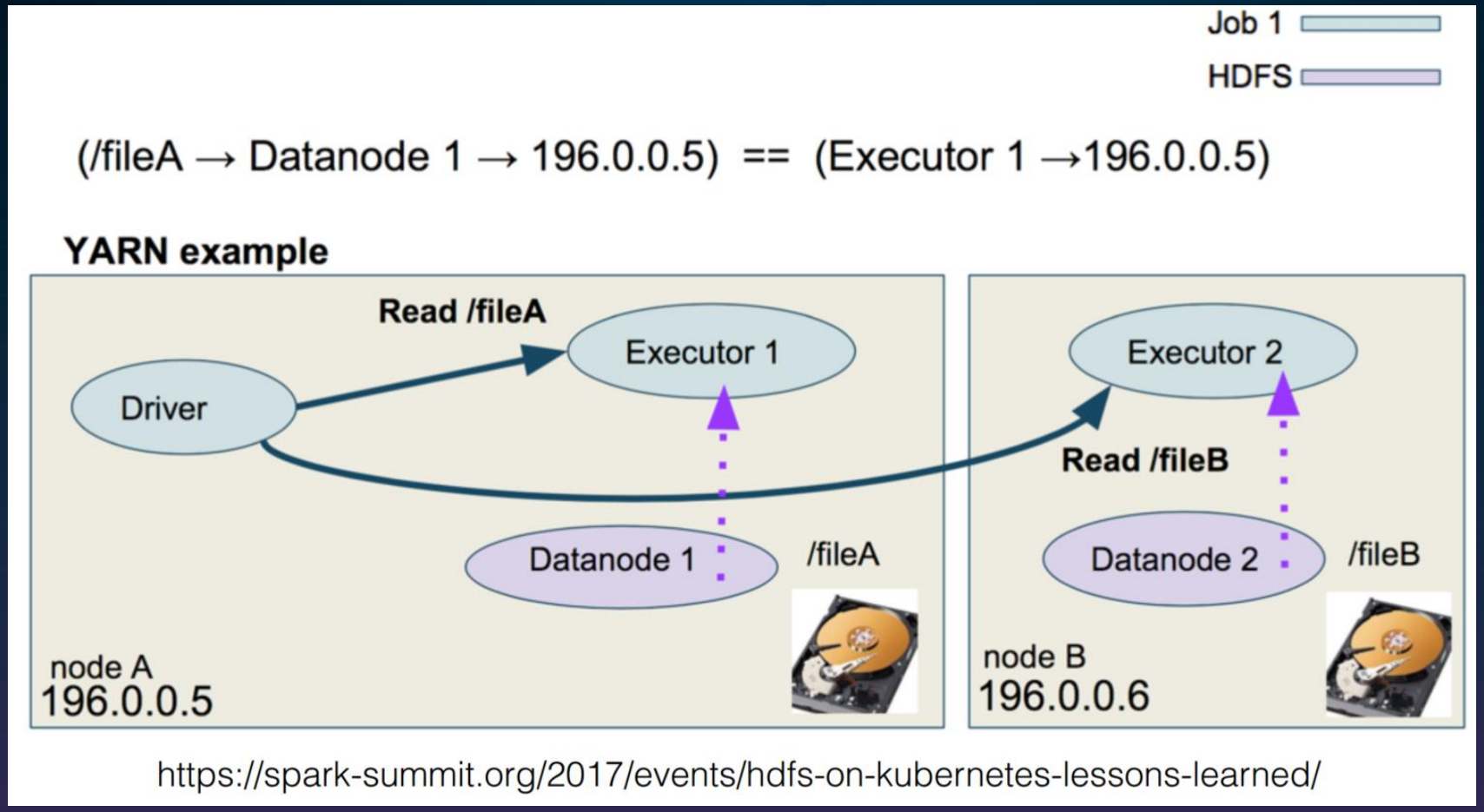
HDFS outside



HDFS Inside

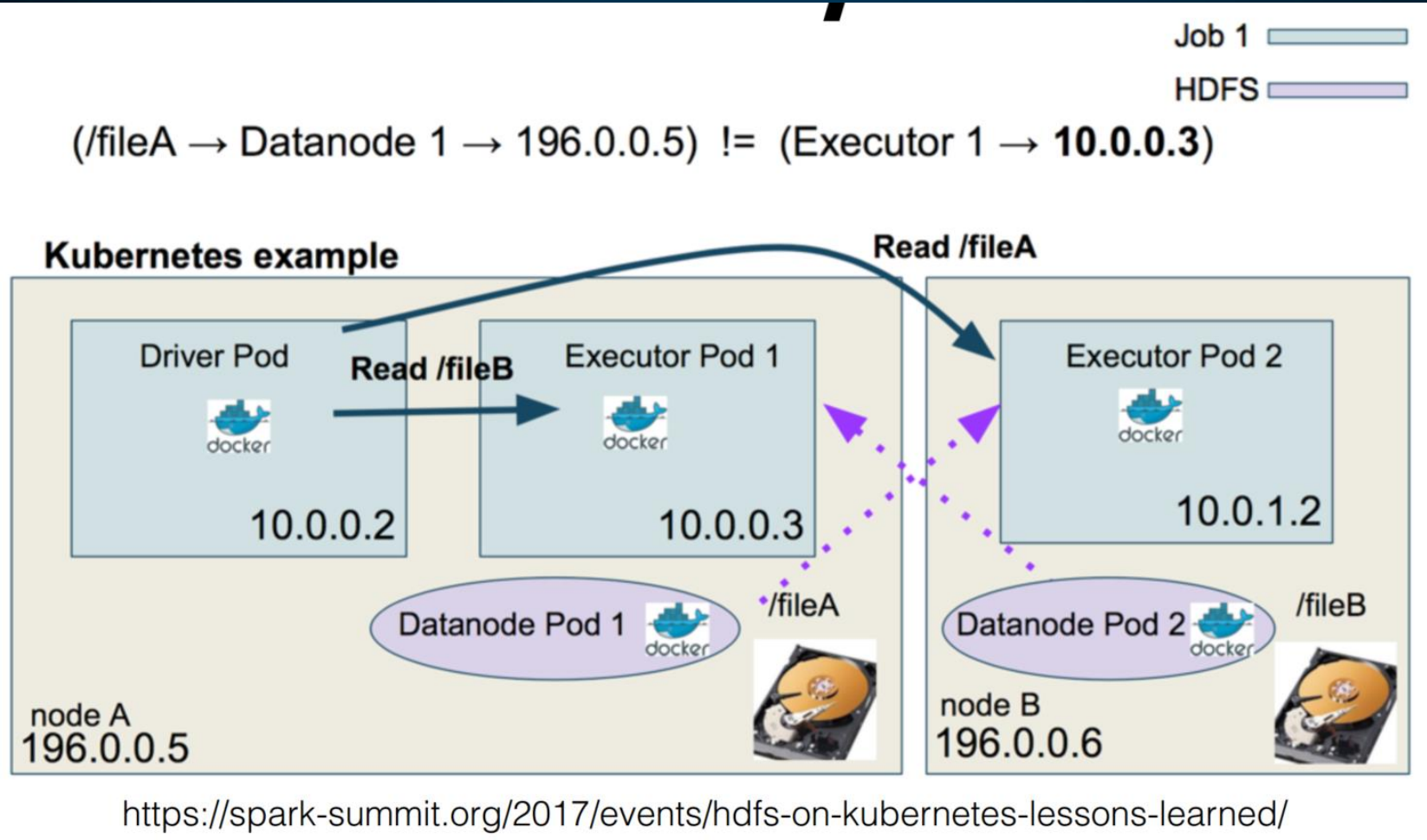

TPCDS@1000
1600
1400
1200
1000
800
600
400
200
0
q1-v2.4
q3-v2.4
q5-v2.4
q7-v2.4
q9-v2.4
q11-v2.4
q13-v2.4
q14b-v2.4
q16-v2.4
q18-v2.4
q20-v2.4
q22-v2.4
q23b-v2.4
q24b-v2.4
q26-v2.4
q28-v2.4
q30-v2.4
q32-v2.4
q34-v2.4
q36-v2.4
q38-v2.4
q39b-v2.4
q41-v2.4
q43-v2.4
q45-v2.4
q47-v2.4
q49-v2.4
q51-v2.4
q53-v2.4
q55-v2.4
q57-v2.4
q59-v2.4
q61-v2.4
q63-v2.4
q65-v2.4
q67-v2.4
q69-v2.4
q71-v2.4
q73-v2.4
q75-v2.4
q77-v2.4
q79-v2.4
q81-v2.4
q83-v2.4
q85-v2.4
q87-v2.4
q89-v2.4
q91-v2.4
q93-v2.4
q95-v2.4
q97-v2.4
q99-v2.4
spark on k8s
spark on yarn

Data Locality on Yarn



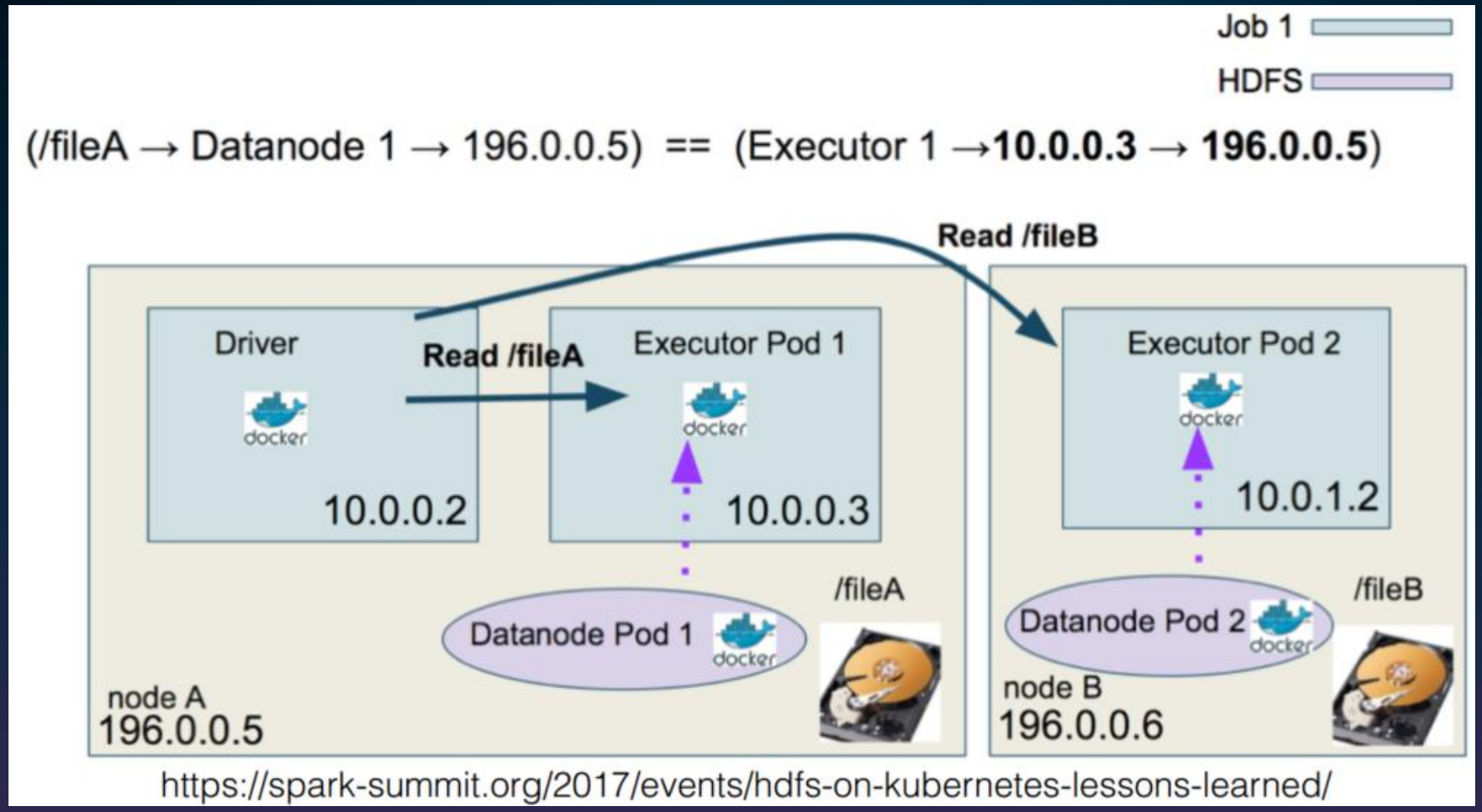
https://spark-summit.org/2017/events/hdfs-on-kubernetes-lessons-learned/

Data Locality on k8s

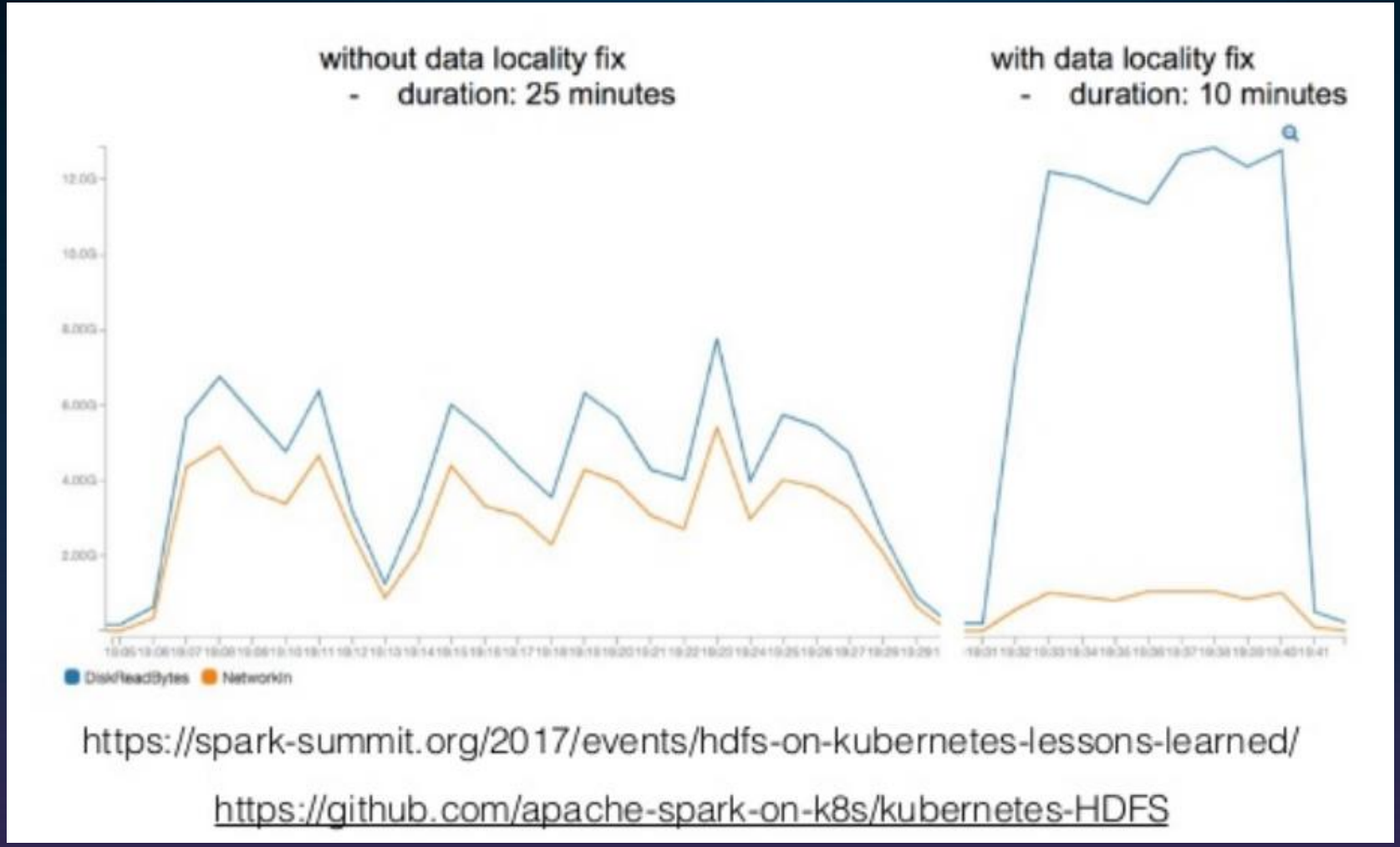


https://spark-summit.org/2017/events/hdfs-on-kubernetes-lessons-learned/

Data Locality on k8s



Rescue Locality Issue



https://spark-summit.org/2017/events/hdfs-on-kubernetes-lessons-learned/

https://github.com/apache-spark-on-k8s/kubernetes-HDFS

Summary

- Unify deployment environment with container
- Containerize Big Data application without performance penalty
- To deeply optimize Big Data workload on k8s should still be improved

Tencent 腾讯 | 腾讯云

+

kubernetes

We are hiring!

jerryshao@tencent.com