Data-aware scheduling

Johannes M. Scheuermann, Cloud Platform Engineer, *inovex GmbH*
Felix Hupfeld, CTO, Quobyte Inc.

Who is standing there?

- Johannes (@inovex/@johscheuer)
- Felix (@Quobyte)
- Partnership and Trainings since 2014
- inovex: systems integration house

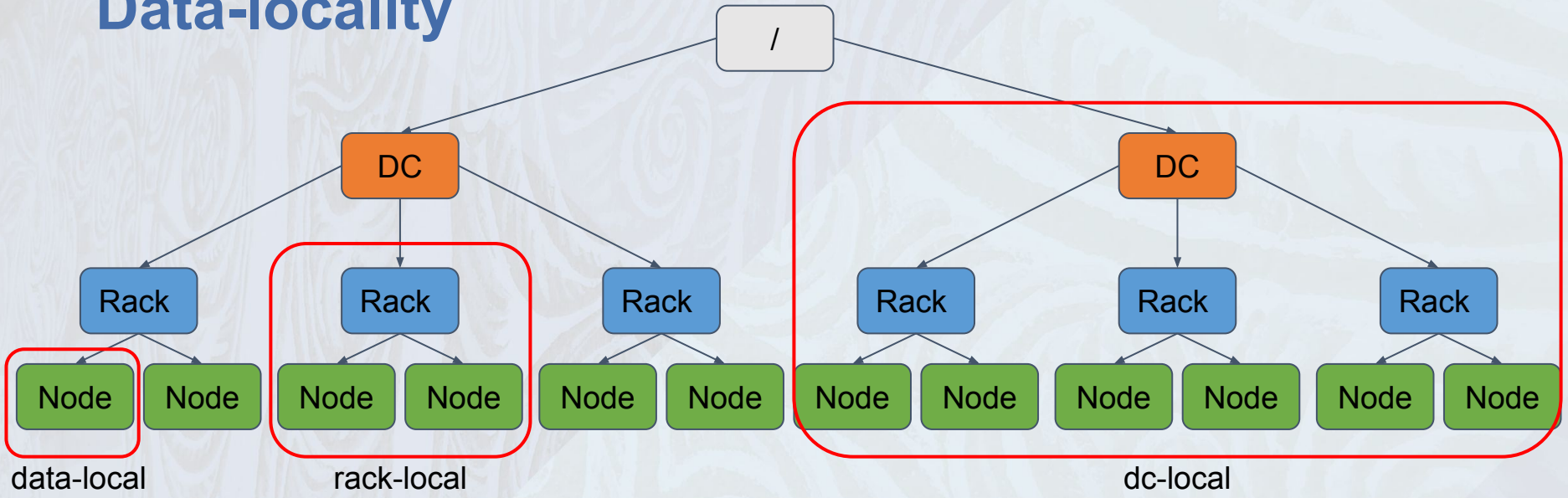
Agenda

- Why data-aware scheduling?
- Data-aware for non Big Data
- Data-aware Scheduler
- Big Data on Kubernetes
- Outlook

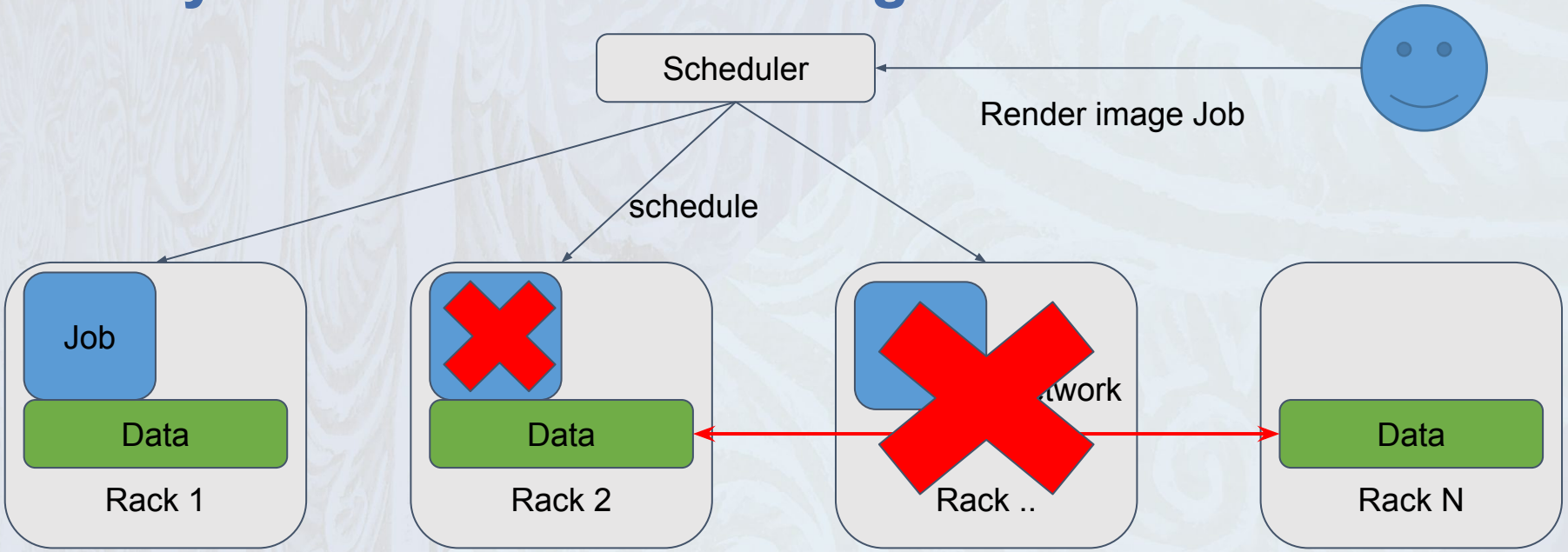
Spoiler(s) ☺

- I’m not an scheduling expert
- Concept is an PoC
- Share learnings
- Get feedback from the community

Data-locality



Why data-aware scheduling



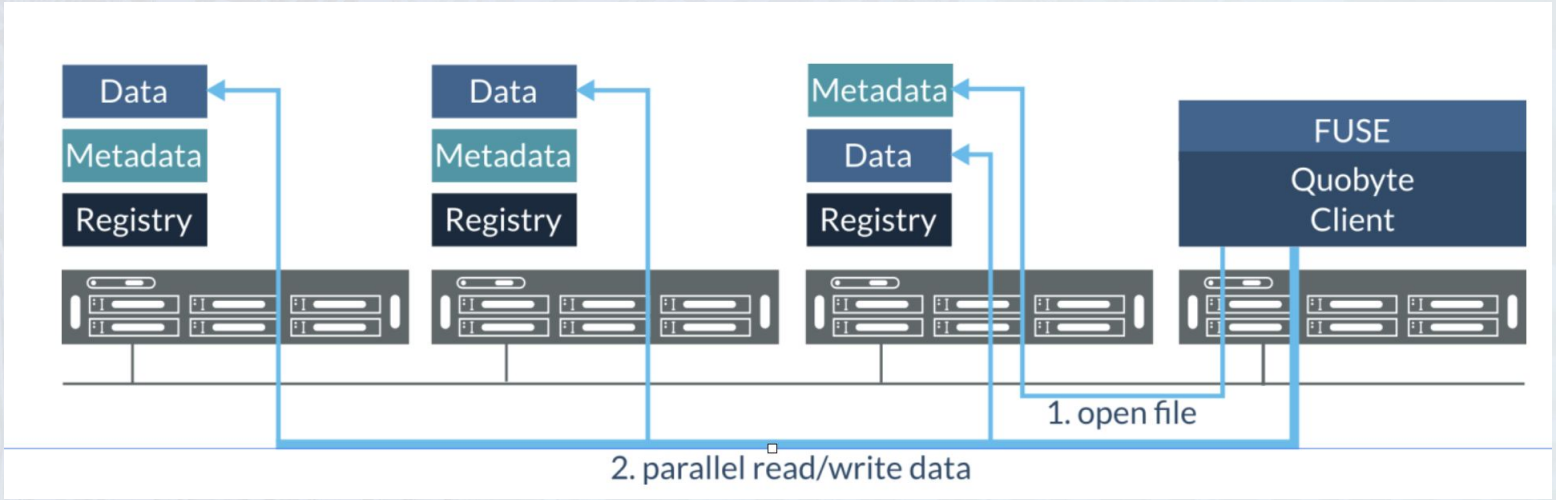
Data-aware scheduling for non Big-Data Apps

- Databases
- (large) Image Processing
- Video encoding
- (Web)-Cache
- Data-intensive workloads

Quobyte - What is Quobyte

- Distributed (parallel) POSIX file system
  - Any workload with high performance (incl. throughput, databases, small files)
- Can be deployed in containers, on kubelet hosts.
  Linearly scalable performance.
- Fully fault-tolerant, split-brain safe

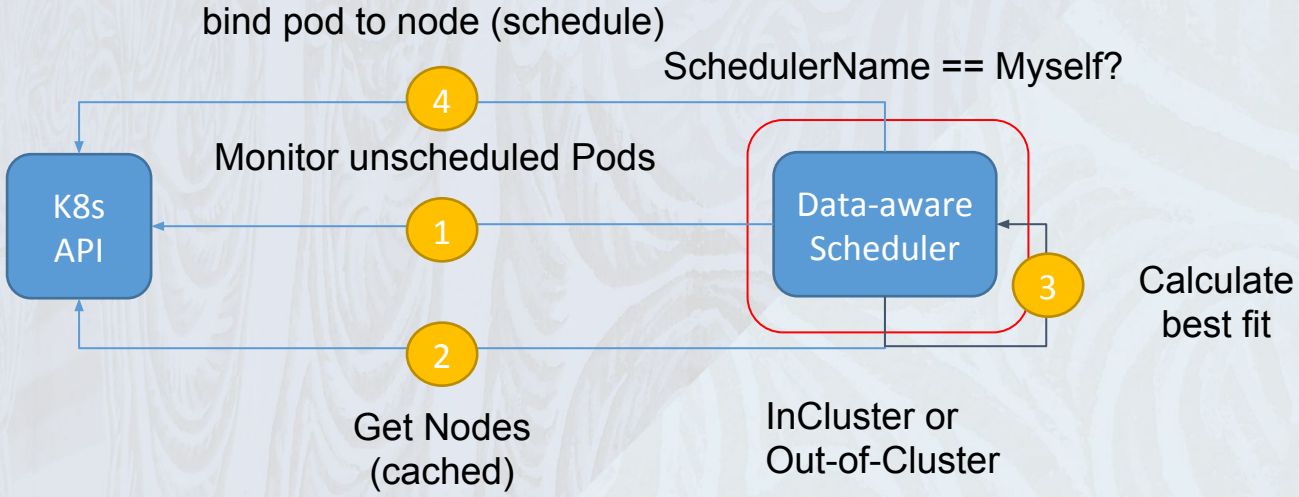
Quobyte - Architecture



Quobyte - Placement

- Metadata servers make placement decisions against policies
  - file level
  - tiering, isolation, …
  - keep stripes of files on disks of same machine => enable local read
  - allow preferring writes to local storage servers => enable local write
- Locality information can be retrieved per file
  - that’s where the schedule hooks in

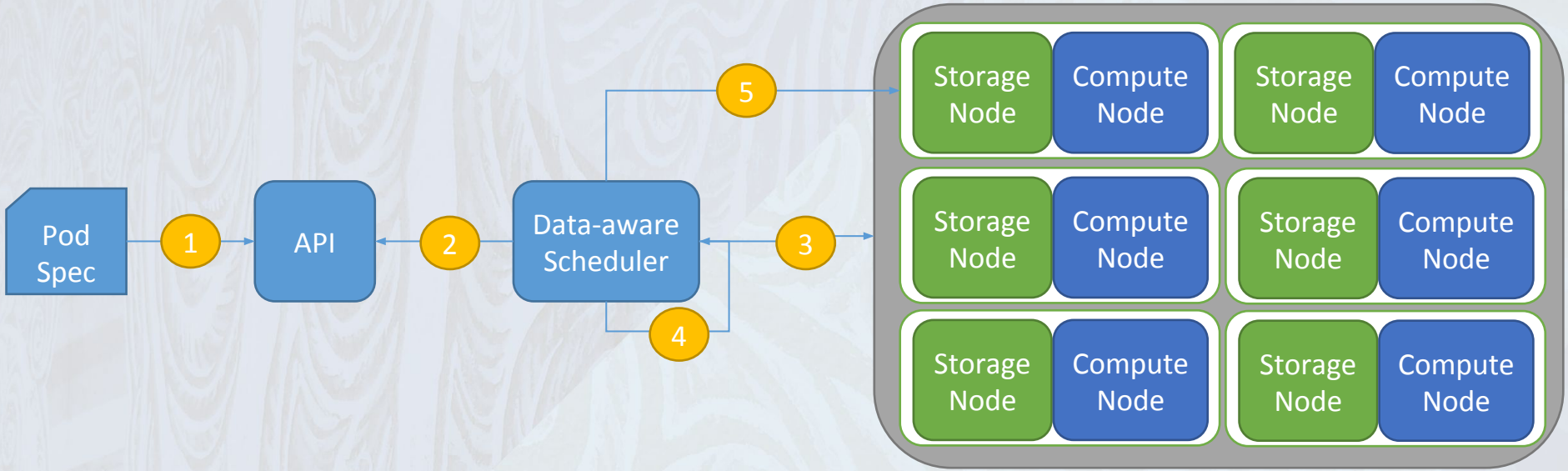
Running multiple schedulers



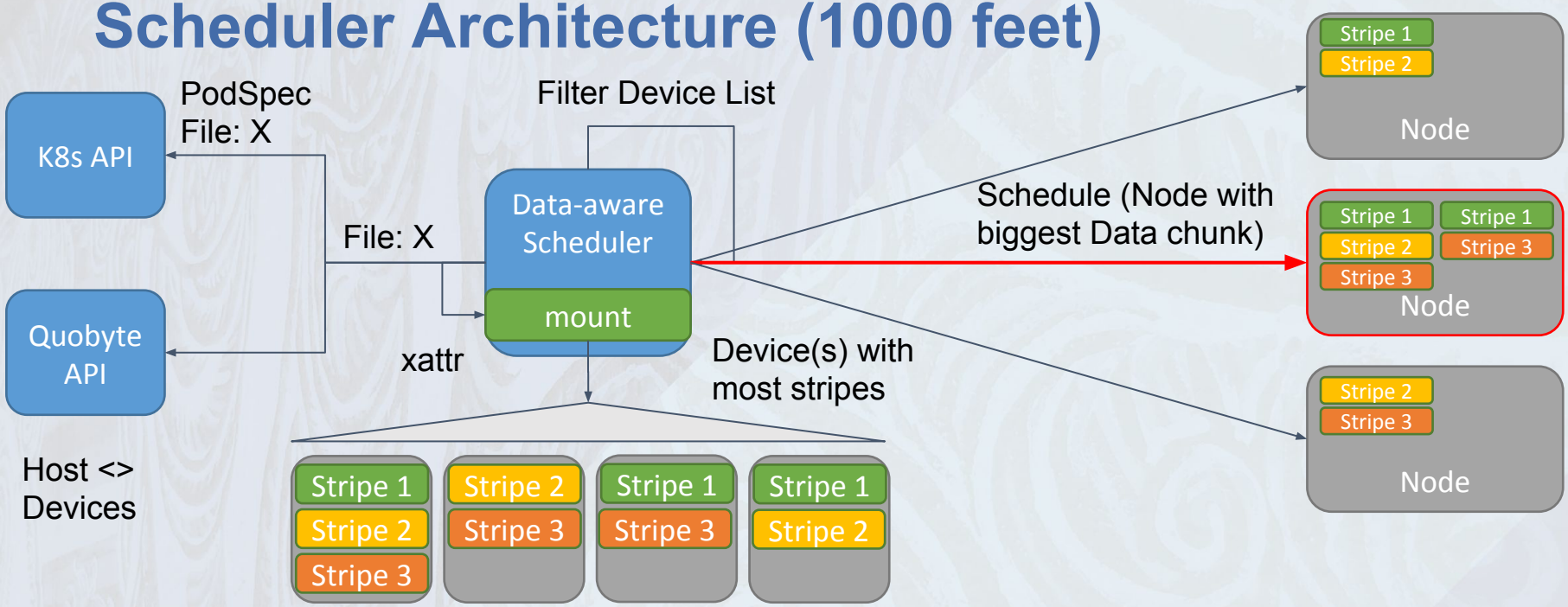
Scheduling data-aware (file-based)

- Specify wanted Data
- Lookup Data Placement
- Remapping if Storage runs in Containers
- Schedule Pod

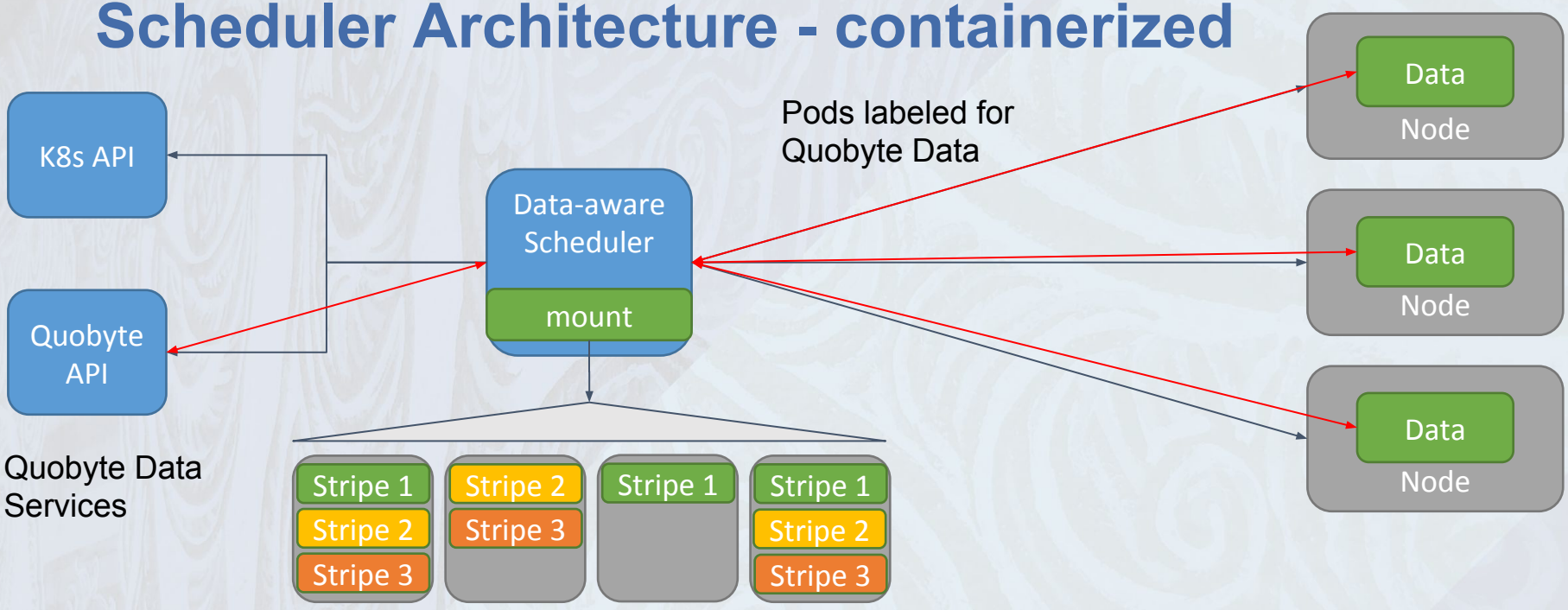
Scheduler Architecture (4000 feet)



Scheduler Architecture (1000 feet)



Scheduler Architecture - containerized



BigData - Analogy

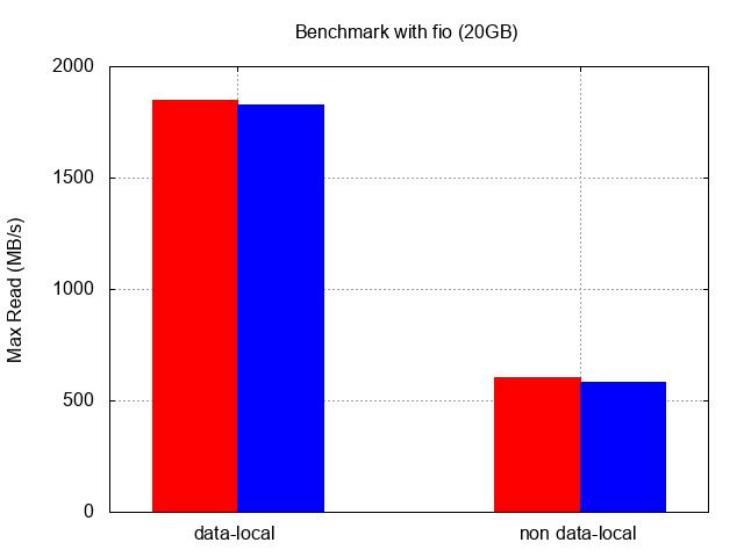
```java
public static void main(String[] args) throws Exception {
    JobConf conf = new JobConf(WordCount.class);
    conf.setJobName("wordcount");

    conf.setOutputKeyClass(Text.class);
    conf.setOutputValueClass(IntWritable.class);

    conf.setMapperClass(Map.class);
    conf.setCombinerClass(Reduce.class);
    conf.setReducerClass(Reduce.class);

    conf.setInputFormat(TextInputFormat.class);
    conf.setOutputFormat(TextOutputFormat.class);

    FileInputFormat.setInputPaths(conf, new Path(args[0]));
    FileOutputFormat.setOutputPath(conf, new Path(args[1]));

    JobClient.runJob(conf);
}
```

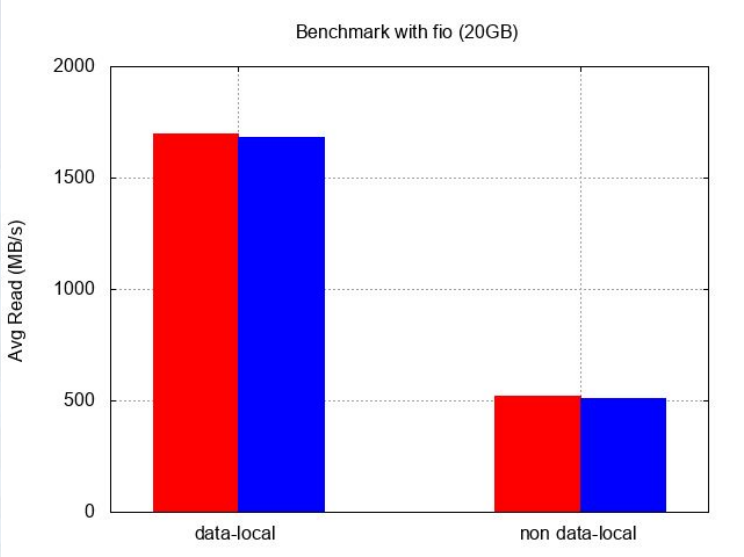
```yaml
apiVersion: extensions/v1beta1
kind: Deployment
metadata:
  labels:
    app: db
  name: db
spec:
  replic
  templ
    meta
      annotati
        "scheduler.alpha.ku
        "scheduler.alpha.quobyte.com
      labels:
        app: db
      name: db
    spec:
      containers:
        - name: db
          image: "mysql:5.7"
          volumeMounts:
          - mountPath: /var/lib/mysql
            name: mysqlVolume
      volumes:
      - name: mysqlVolume
        quobyte:
          registry: registry:7861
          volume: MySQLVolume
```

Kubernetes v1.5

Demo?

Benchmarks (SSD, Seq. Read, BK=1M)

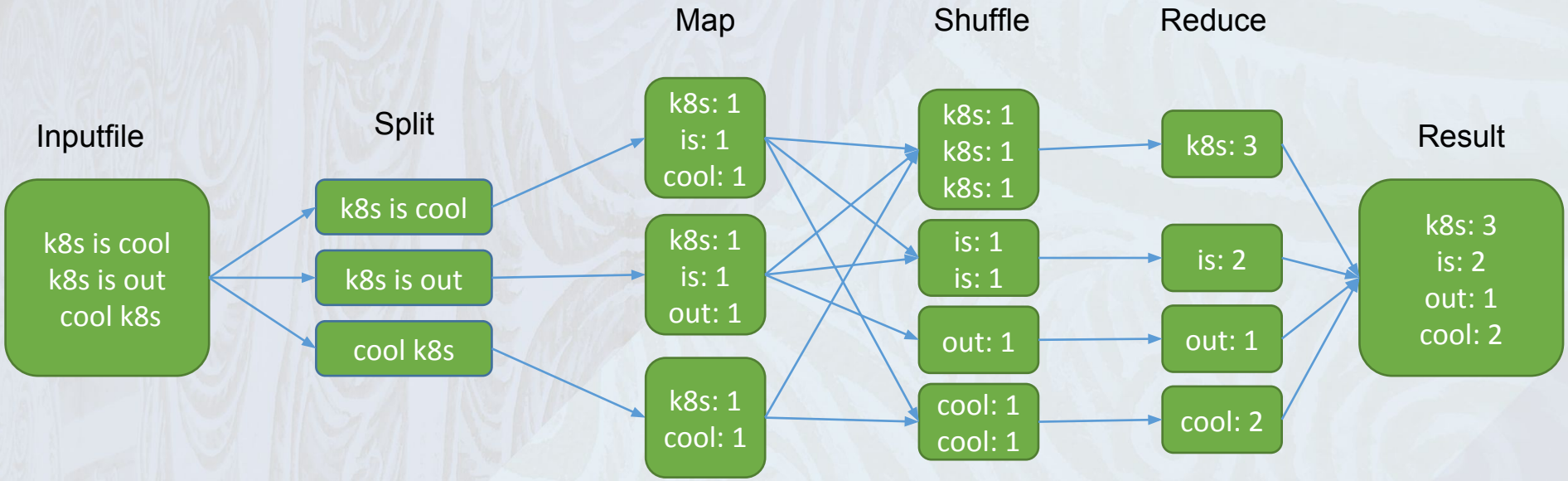




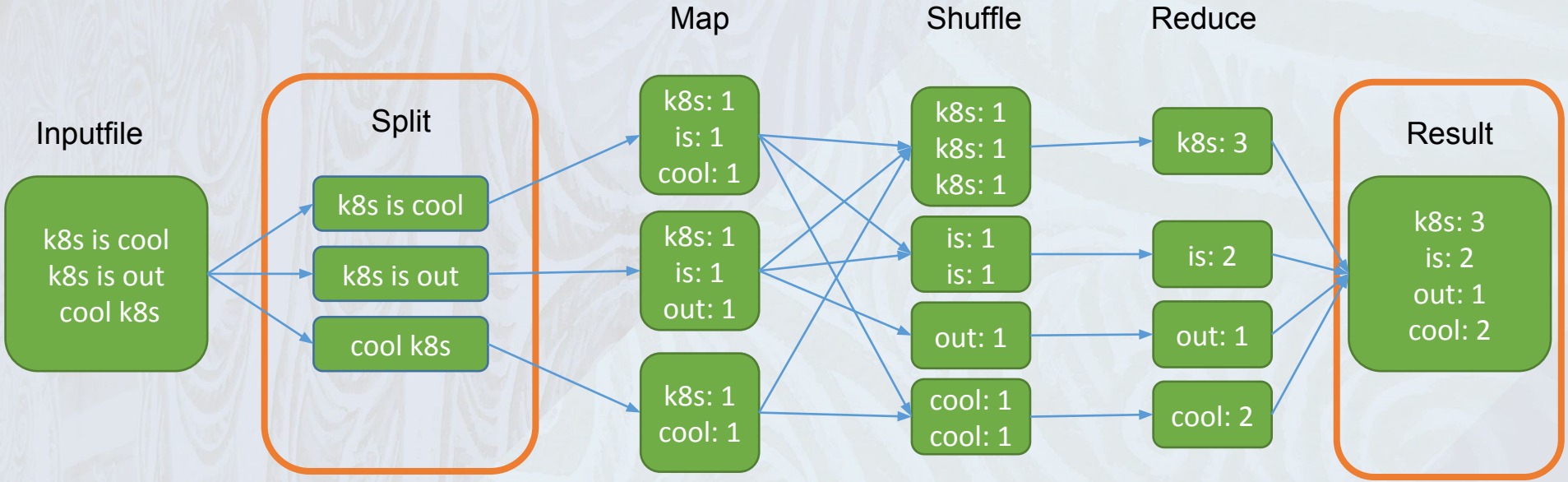
Big Data and Kubernetes

- Make use of HDFS
  - as protocol
  - as Filesystem
- In progress
- https://github.com/apache-spark-on-k8s/spark
  - dynamic scheduling
  - better integration
  - data locality? (https://github.com/apache-spark-on-k8s/spark/issues/206)

Map Reduce


Inputfile
Split
Map
Shuffle
Reduce
Result
k8s is cool
k8s is out
cool k8s
k8s is cool
k8s is out
cool k8s
k8s: 1
is: 1
cool: 1
k8s: 1
is: 1
out: 1
k8s: 1
cool: 1
k8s: 1
k8s: 1
k8s: 1
is: 1
is: 1
out: 1
cool: 1
cool: 1
k8s: 3
is: 2
out: 1
cool: 2
k8s: 3
is: 2
out: 1
cool: 2

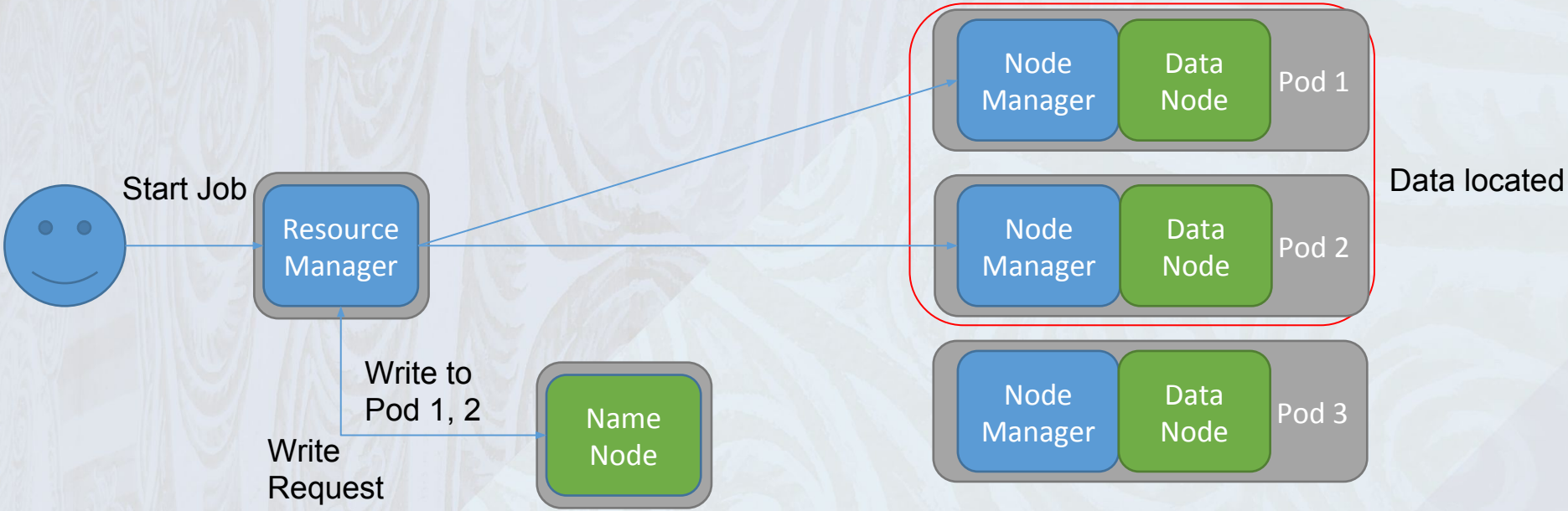
Map Reduce (data-locality)



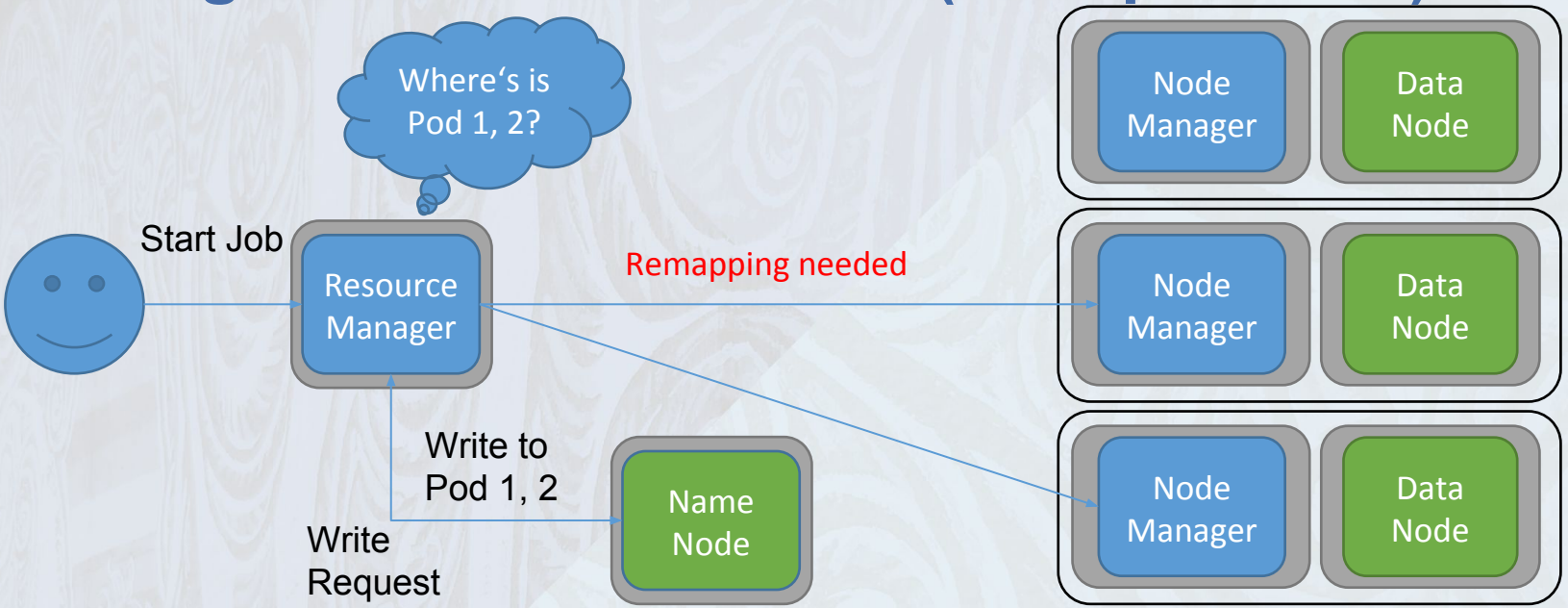
Requirements for data-locality

- Local storage (not S3, GCS etc.)
- Definition of “local”
- Topology awareness
- Data Node Name == Node Manager Name
- Scheduler Configuration

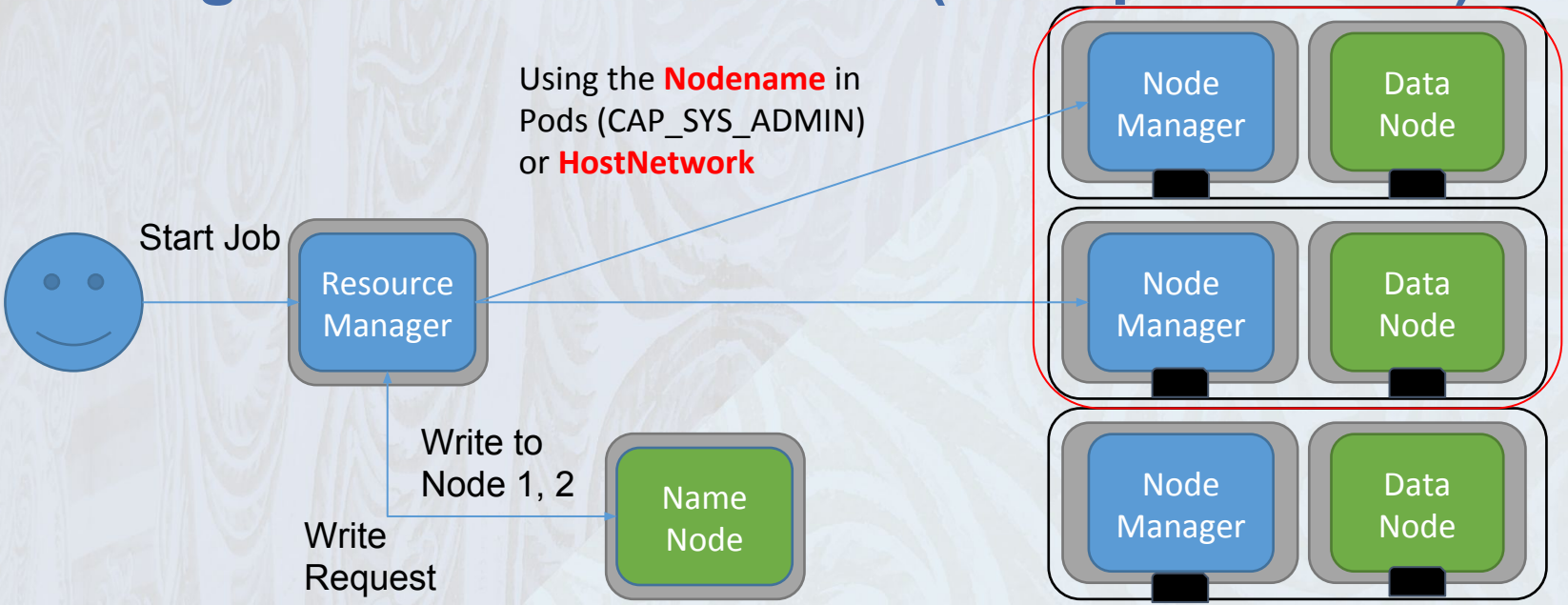
Big Data on Kubernetes (one Pod)



Big Data on Kubernetes (multiple Pods)



Big Data on Kubernetes (multiple Pods 2)



Summary (Big Data and data-locality)

	One Pod	Multiple Pods	Multiple Pods (hostname)
Data locality	✓	✗	✓
Independent scaling	✗	✓	(✓)
Unprivileged	✓	✓	✗

Out-look

- Data locality is possible (even for non Big-Data apps)
  - Workarounds are needed
  - Do we really need pure data locality?
- Data-locality and Big Data on Kubernetes
  - https://github.com/apache-spark-on-k8s/spark/issues/128
  - https://github.com/apache-spark-on-k8s/kubernetes-HDFS
  - Rack-local often good enough

Links

- https://github.com/johscheuer/data-aware-scheduler
- https://github.com/inovex/quobyte-kubernetes-operator
- https://github.com/inovex/kubernetes-demo

Vielen Dank

Johannes Scheuermann
Cloud Platform Engineer

inovex GmbH
Ludwig-Erhard-Allee 6
76131 Karlsruhe

jscheuermann@inovex.de
0173 3181058