Running Apache Spark on Mesos

Timothy Chen
tim@mesosphere.io

mesosphere

Geekbang

极客邦科技

全球领先的技术人学习和交流平台

InfoQ

专注中高端技术
人员的社区媒体

EGO EXTRA GEEKS' ORGANIZATION NETWORKS

高端技术人员
学习型社交网络

StuQ

实践驱动的IT职业
学习和服务平台

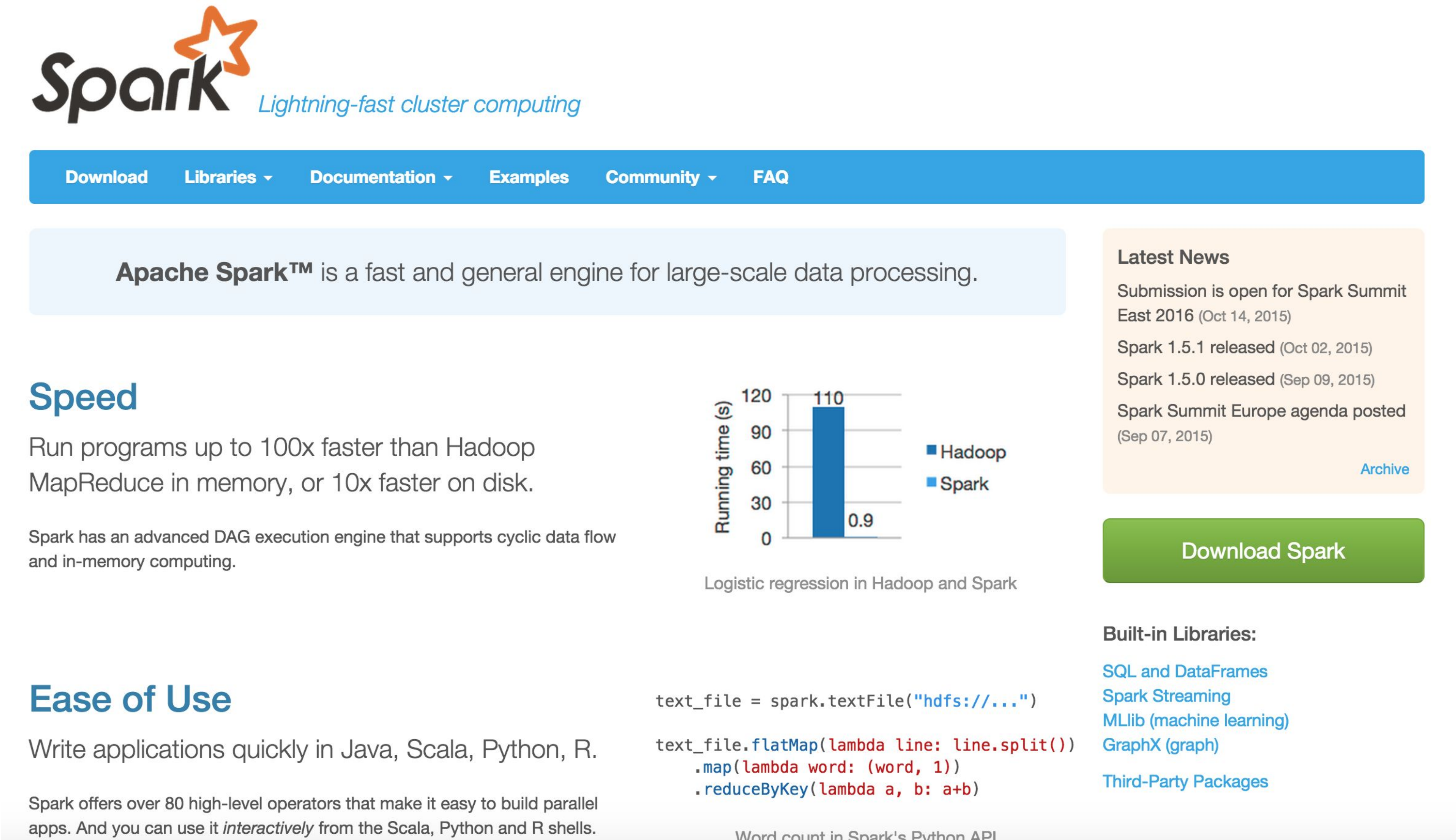
扫我，码上开启新世界

Geekbang

InfoQ | EGO EXTRA GEEKS' ORGANIZATION NETWORKS | StuQ

About me:

- Distributed Systems Architect @ Mesosphere
  - Lead Containerization engineering
- Apache Mesos, Drill PMC / Committer
- Maintain Apache Spark Mesos Schedulers



Apache Spark™ is a fast and general engine for large-scale data processing.

Speed

Run programs up to 100x faster than Hadoop MapReduce in memory, or 10x faster on disk.

Spark has an advanced DAG execution engine that supports cyclic data flow and in-memory computing.

Logistic regression in Hadoop and Spark

Ease of Use

Write applications quickly in Java, Scala, Python, R.

Spark offers over 80 high-level operators that make it easy to build parallel apps. And you can use it *interactively* from the Scala, Python and R shells.

```
text_file = spark.textFile("hdfs://...")

text_file.flatMap(lambda line: line.split())
    .map(lambda word: (word, 1))
    .reduceByKey(lambda a, b: a+b)
```

Word count in Spark's Python API

Latest News

Submission is open for Spark Summit East 2016 (Oct 14, 2015)

Spark 1.5.1 released (Oct 02, 2015)

Spark 1.5.0 released (Sep 09, 2015)

Spark Summit Europe agenda posted (Sep 07, 2015)

Archive

Download Spark

Built-in Libraries:

SQL and DataFrames
Spark Streaming
MLlib (machine learning)
GraphX (graph)

Third-Party Packages

Mesos: A Platform for Fine-Grained Resource Sharing in the Data Center

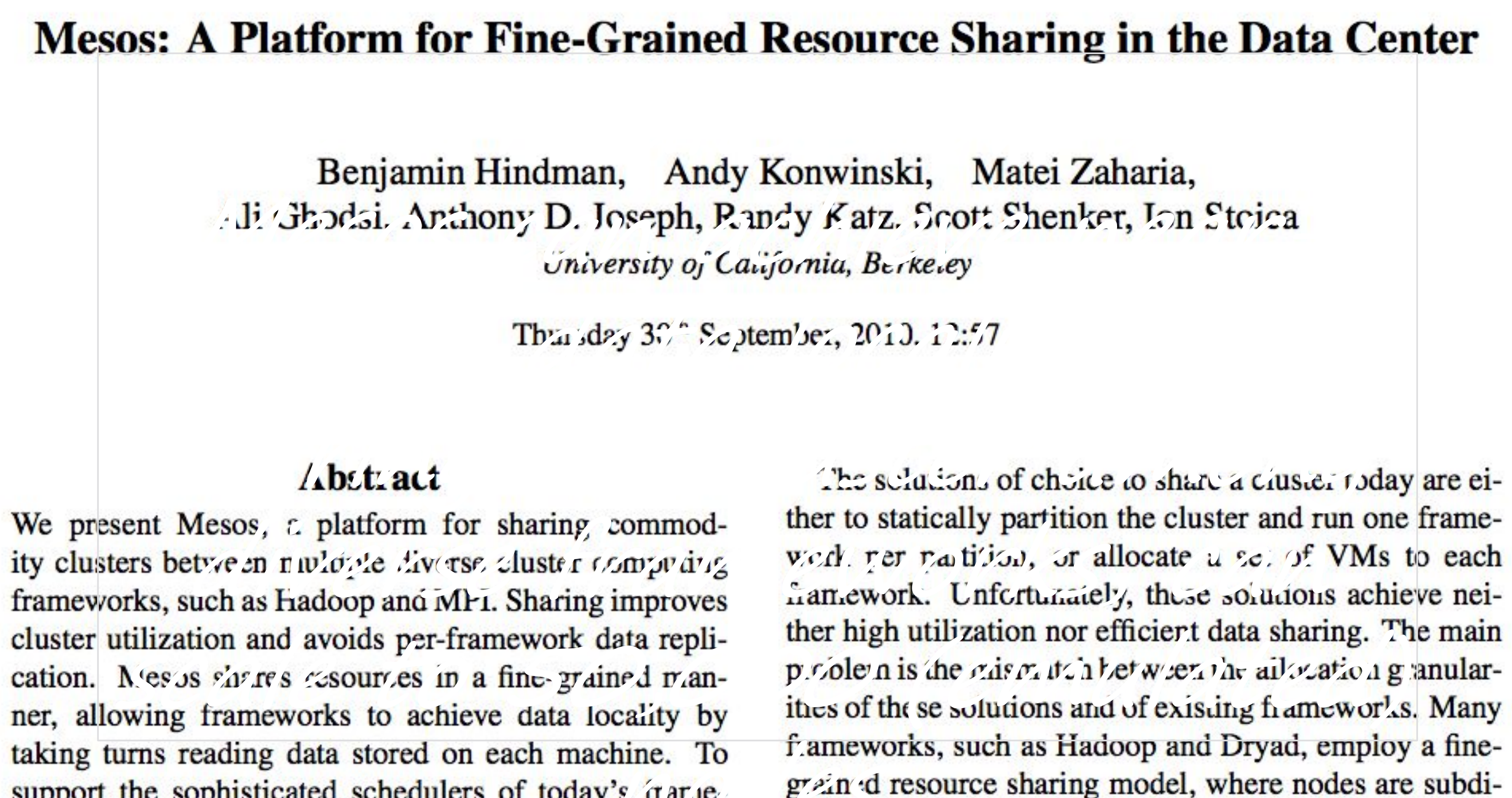
Benjamin Hindman, Andy Konwinski, Matei Zaharia,
Ali Ghodsi, Anthony D. Joseph, Randy Katz, Scott Shenker, Ion Stoica
University of California, Berkeley

Thursday 30th September, 2010, 12:57

Abstract

We present Mesos, a platform for sharing commodity clusters between multiple diverse cluster computing frameworks, such as Hadoop and MPI. Sharing improves cluster utilization and avoids per-framework data repli-

The solutions of choice to share a cluster today are either to statically partition the cluster and run one framework per partition, or allocate a set of VMs to each framework. Unfortunately, these solutions achieve neither high utilization nor efficient data sharing. The main



Mesos: A Platform for Fine-Grained Resource Sharing in the Data Center

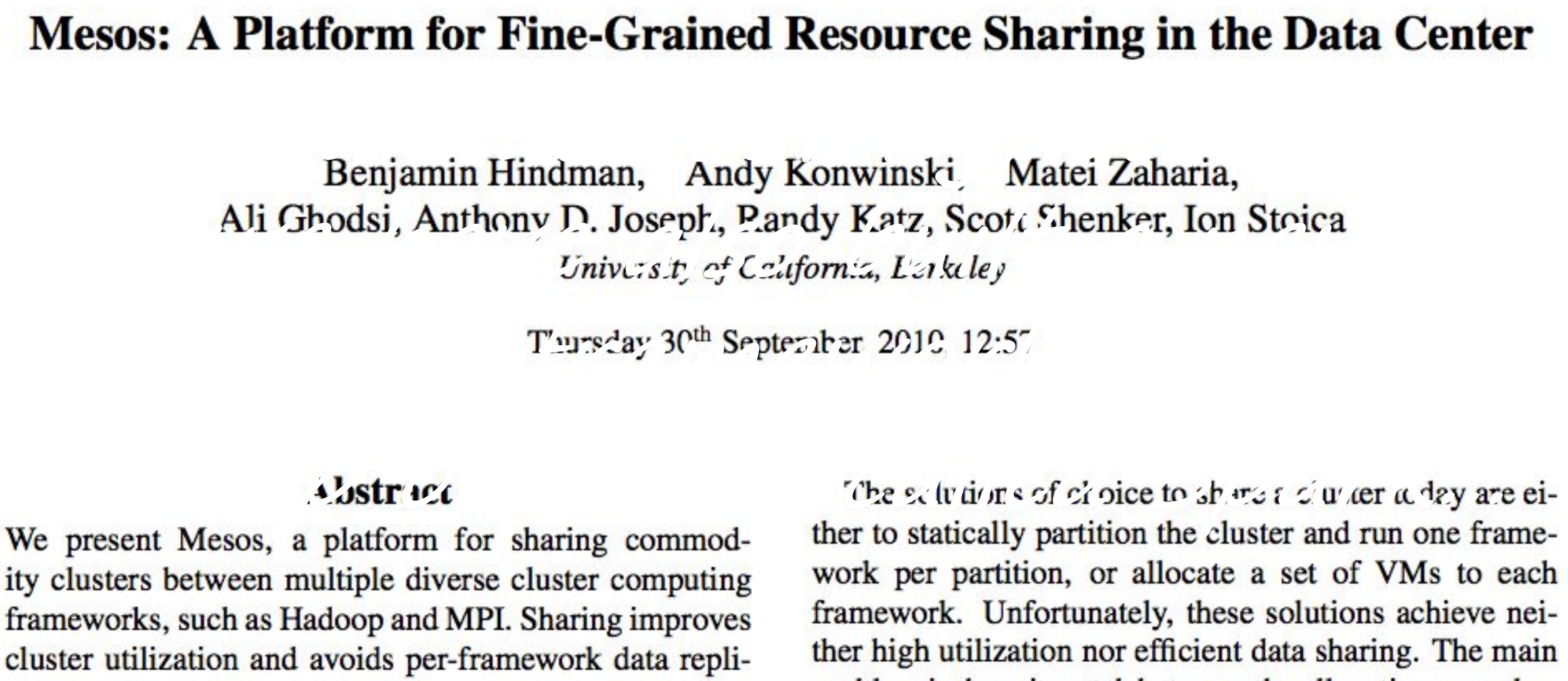
Benjamin Hindman, Andy Konwinski, Matei Zaharia,
Ali Ghodsi, Anthony D. Joseph, Randy Katz, Scott Shenker, Ion Stoica
University of California, Berkeley

Thursday 30th September, 2010, 12:57

Abstract

We present Mesos, a platform for sharing commodity clusters between multiple diverse cluster computing frameworks, such as Hadoop and MPI. Sharing improves cluster utilization and avoids per-framework data replication. Mesos shares resources in a fine-grained manner, allowing frameworks to achieve data locality by taking turns reading data stored on each machine. To support the sophisticated schedulers of today's frame-

The solutions of choice to share a cluster today are either to statically partition the cluster and run one framework per partition, or allocate a set of VMs to each framework. Unfortunately, these solutions achieve neither high utilization nor efficient data sharing. The main problem is the mismatch between the allocation granularities of these solutions and of existing frameworks. Many frameworks, such as Hadoop and Dryad, employ a fine-grained resource sharing model, where nodes are subdi-



Mesos: A Platform for Fine-Grained Resource Sharing in the Data Center

Benjamin Hindman, Andy Konwinski, Matei Zaharia,
Ali Ghodsi, Anthony D. Joseph, Randy Katz, Scott Shenker, Ion Stoica
University of California, Berkeley

Thursday 30th September 2010, 12:57

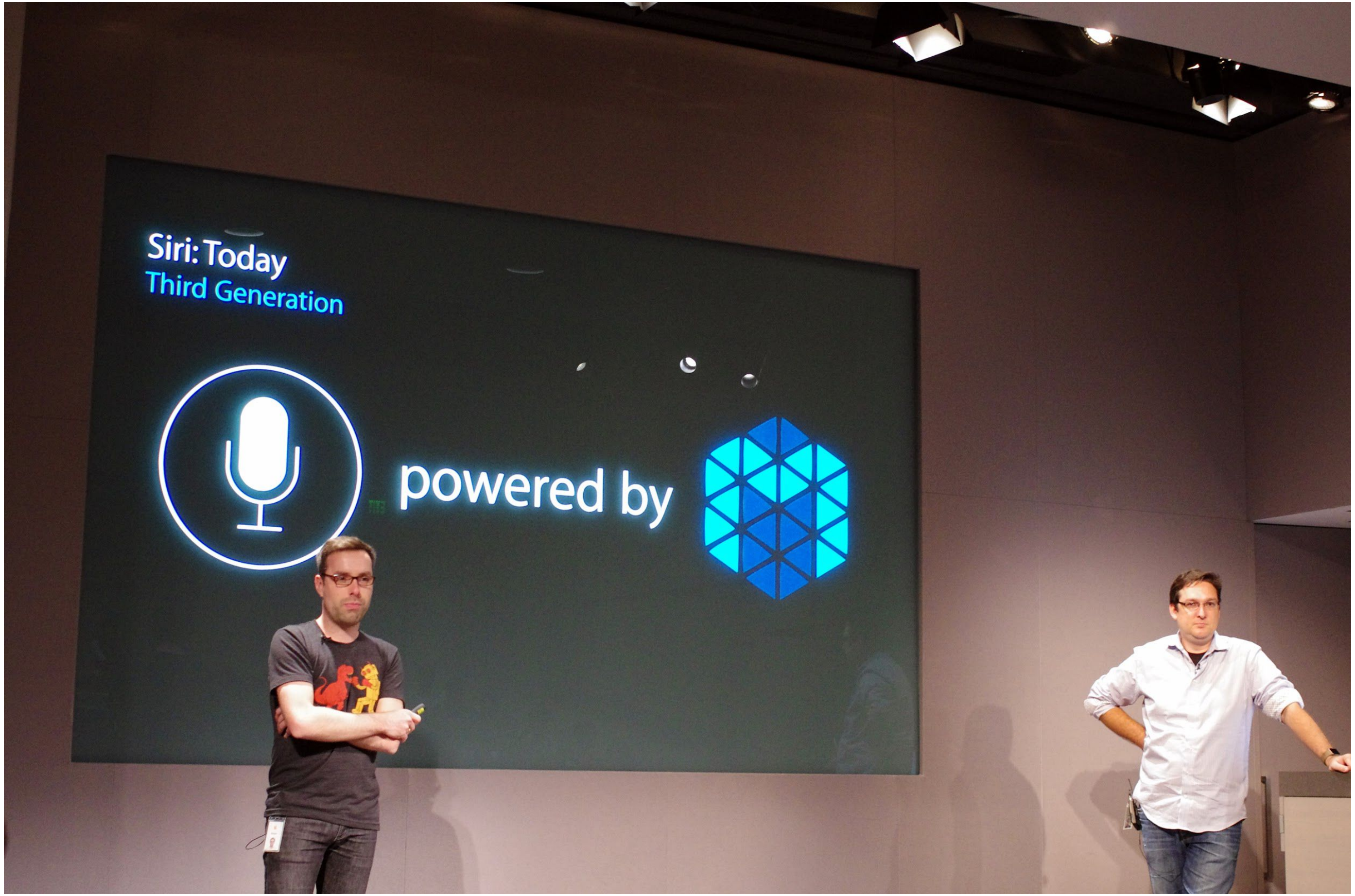
Abstract

We present Mesos, a platform for sharing commodity clusters between multiple diverse cluster computing frameworks, such as Hadoop and MPI. Sharing improves cluster utilization and avoids per-framework data repli-

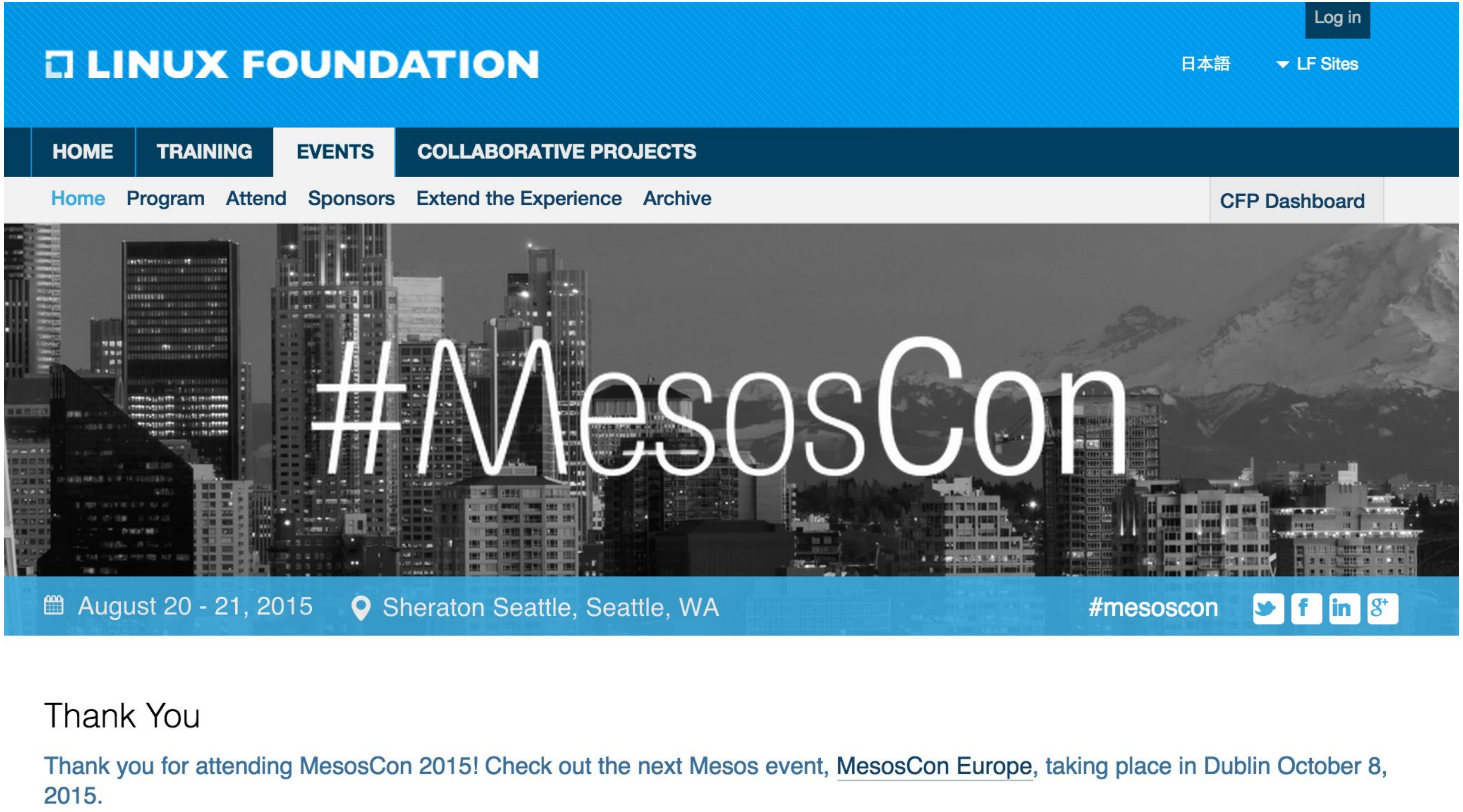
The solutions of choice to share a cluster today are either to statically partition the cluster and run one framework per partition, or allocate a set of VMs to each framework. Unfortunately, these solutions achieve neither high utilization nor efficient data sharing. The main

Apache Mesos

airbnb
ebay
verizon
Make progress every day
shopify
HubSpot
ignidata
igniting business with data
xogito
...radical thinking...
PayPal tm
DueDil
Atlassian
CLOUD PHYSICS
device scape
mesosphere
yelp
灵雀云
alauda.cn
数人云



Siri: Today
Third Generation
powered by

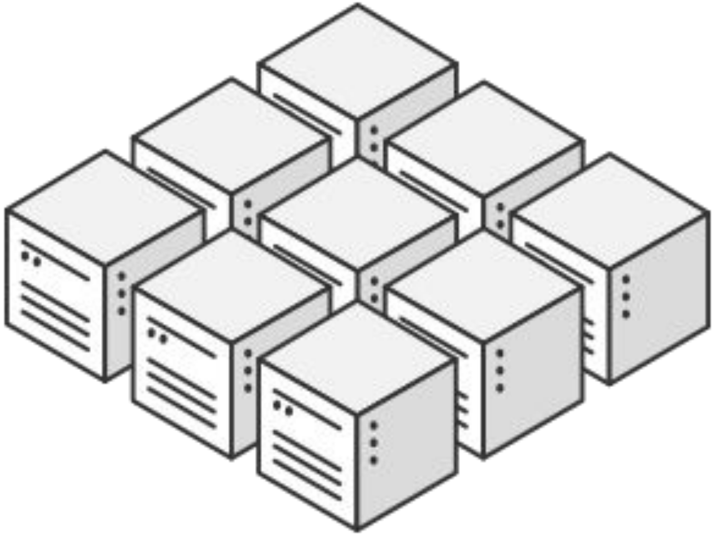


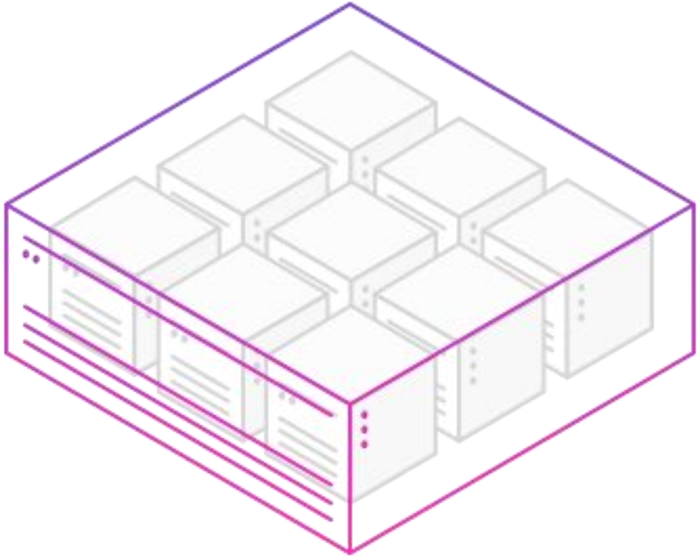
August 20 - 21, 2015

Sheraton Seattle, Seattle, WA

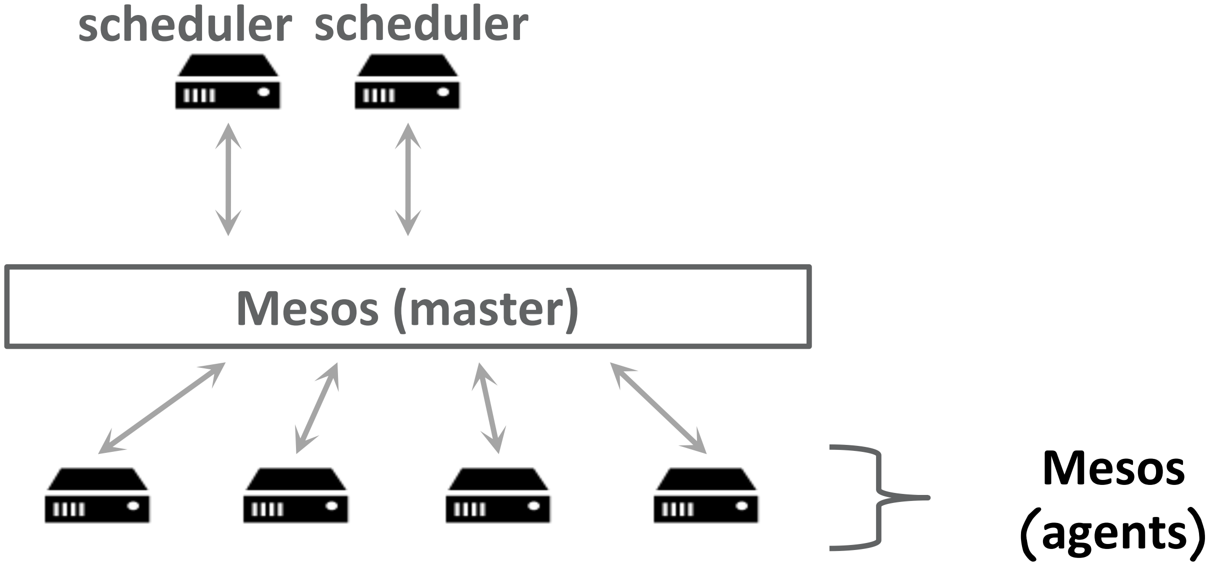
Thank You

Thank you for attending MesosCon 2015! Check out the next Mesos event, MesosCon Europe, taking place in Dublin October 8, 2015.





Mesos: level of indirection



Mesos

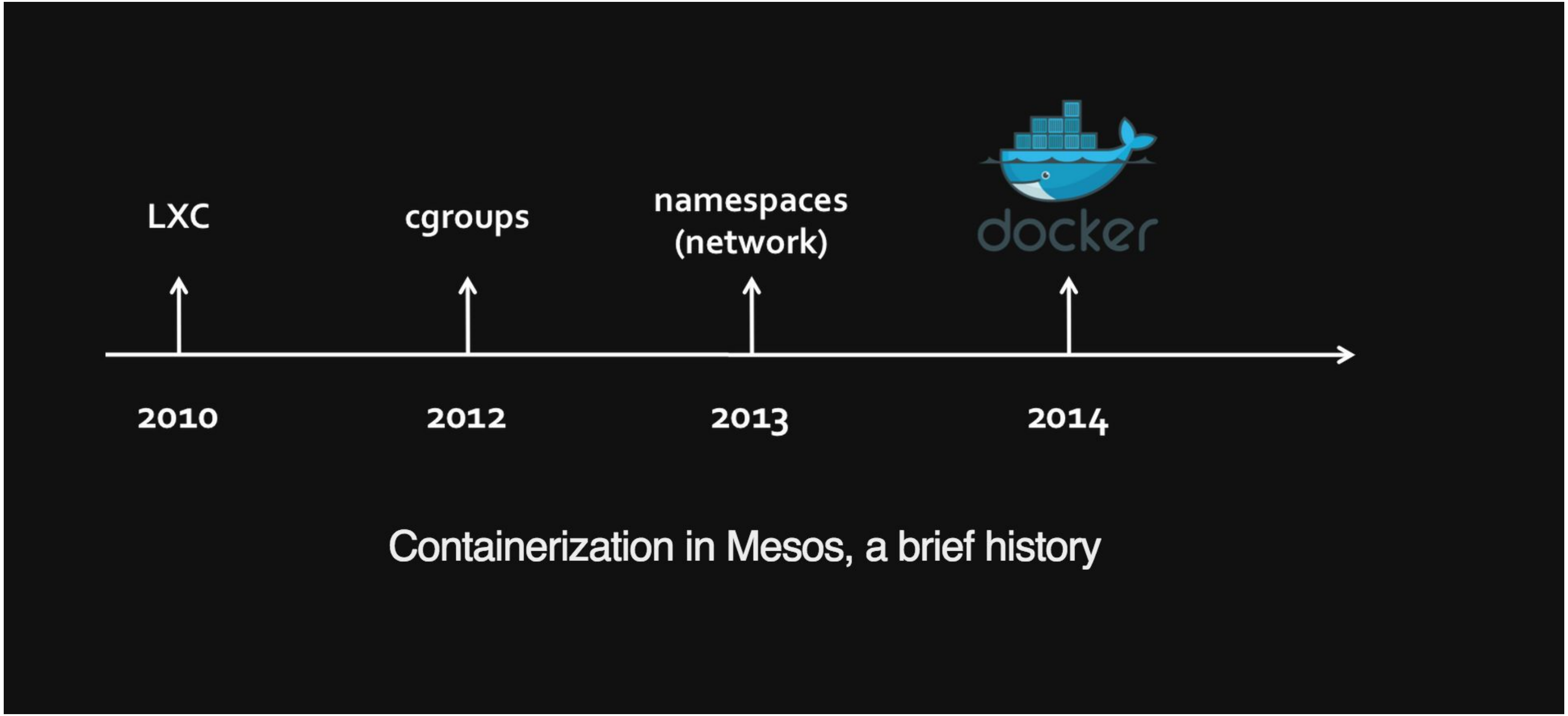
Improve utilization by sharing cluster

Support multiple frameworks with weighted DRF and roles

Allow Isolation among frameworks and jobs

Simplified Operations and Development

Mesos Community Frameworks & Tools



LXC

2010

cgroups

2012

namespaces
(network)

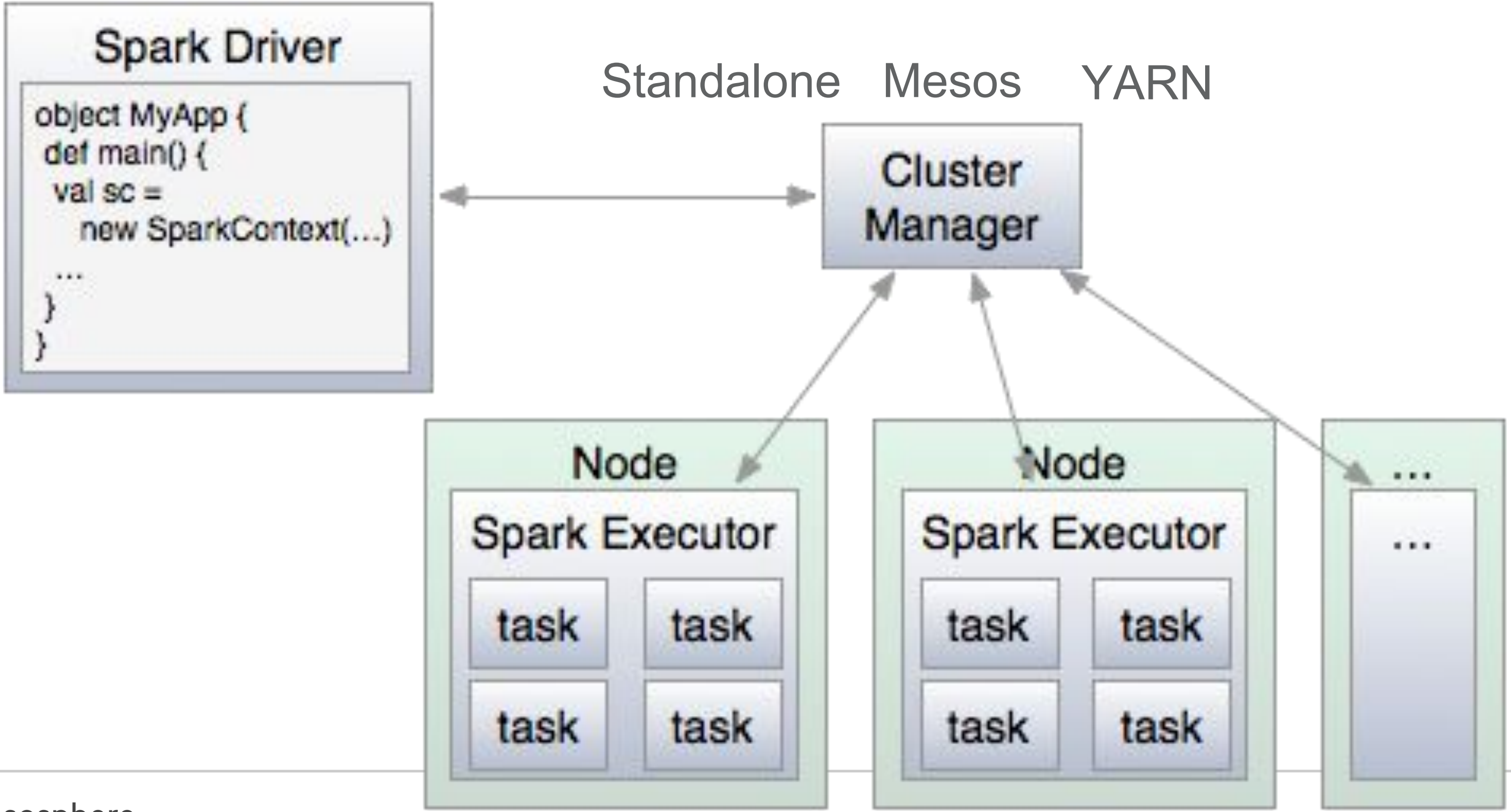
2013

docker

2014

Containerization in Mesos, a brief history

Spark Cluster Abstraction



Spark Driver

```
object MyApp {
 def main() {
  val sc =
    new SparkContext(...)
  ...
 }
}
```

Standalone Mesos YARN

Cluster Manager

Node

Spark Executor

task task

task task

Node

Spark Executor

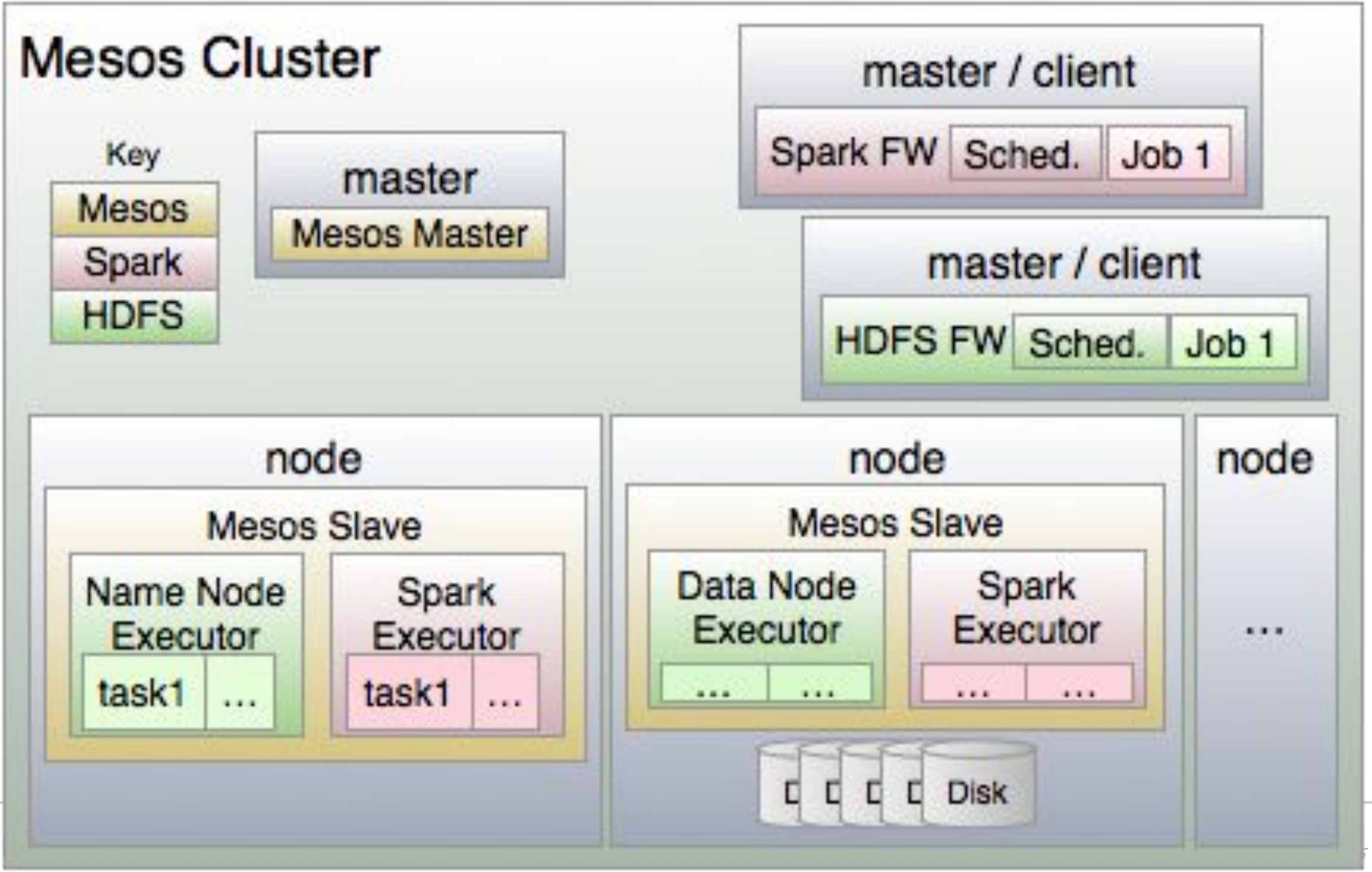
task task

task task

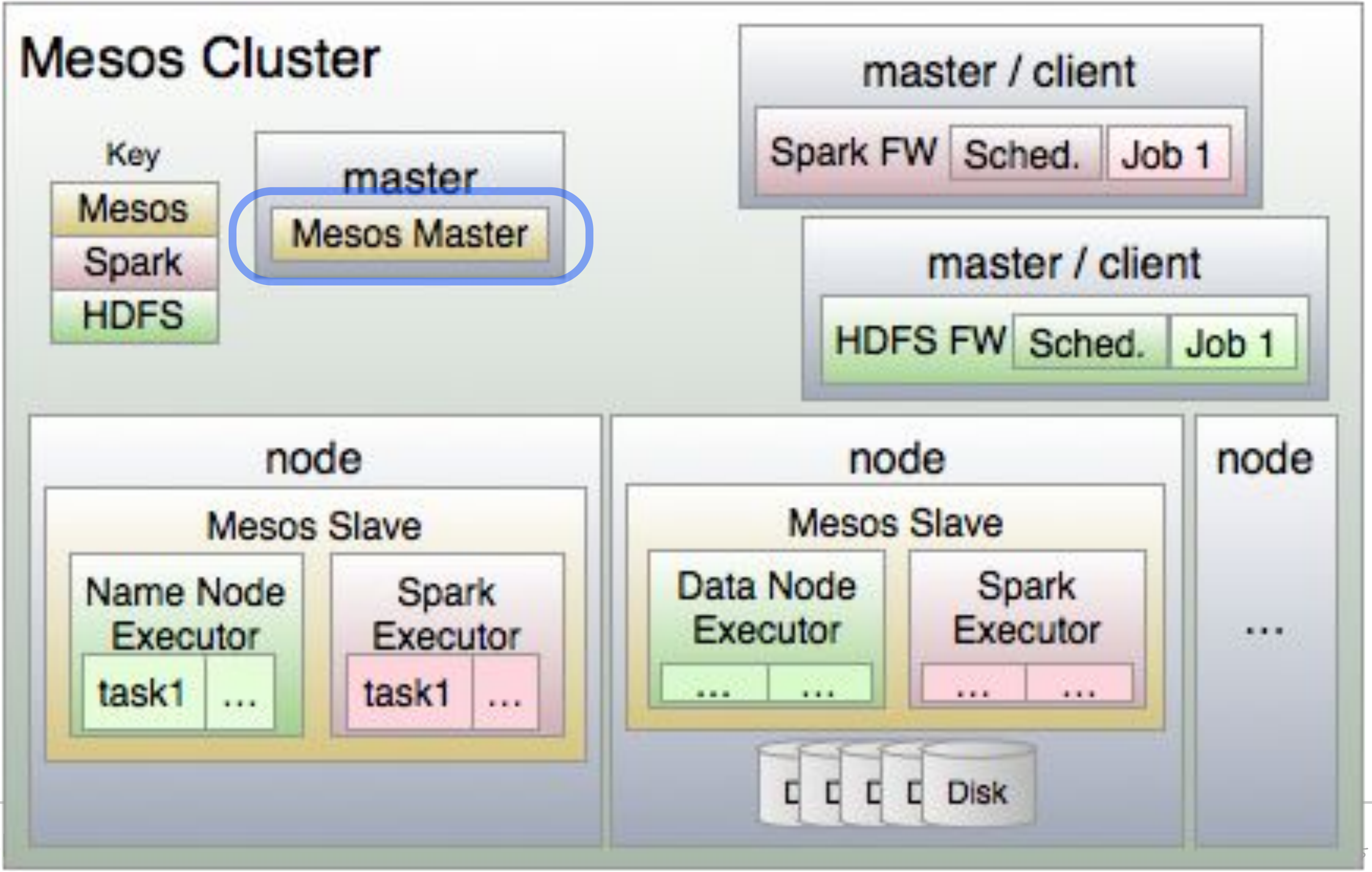
...

...

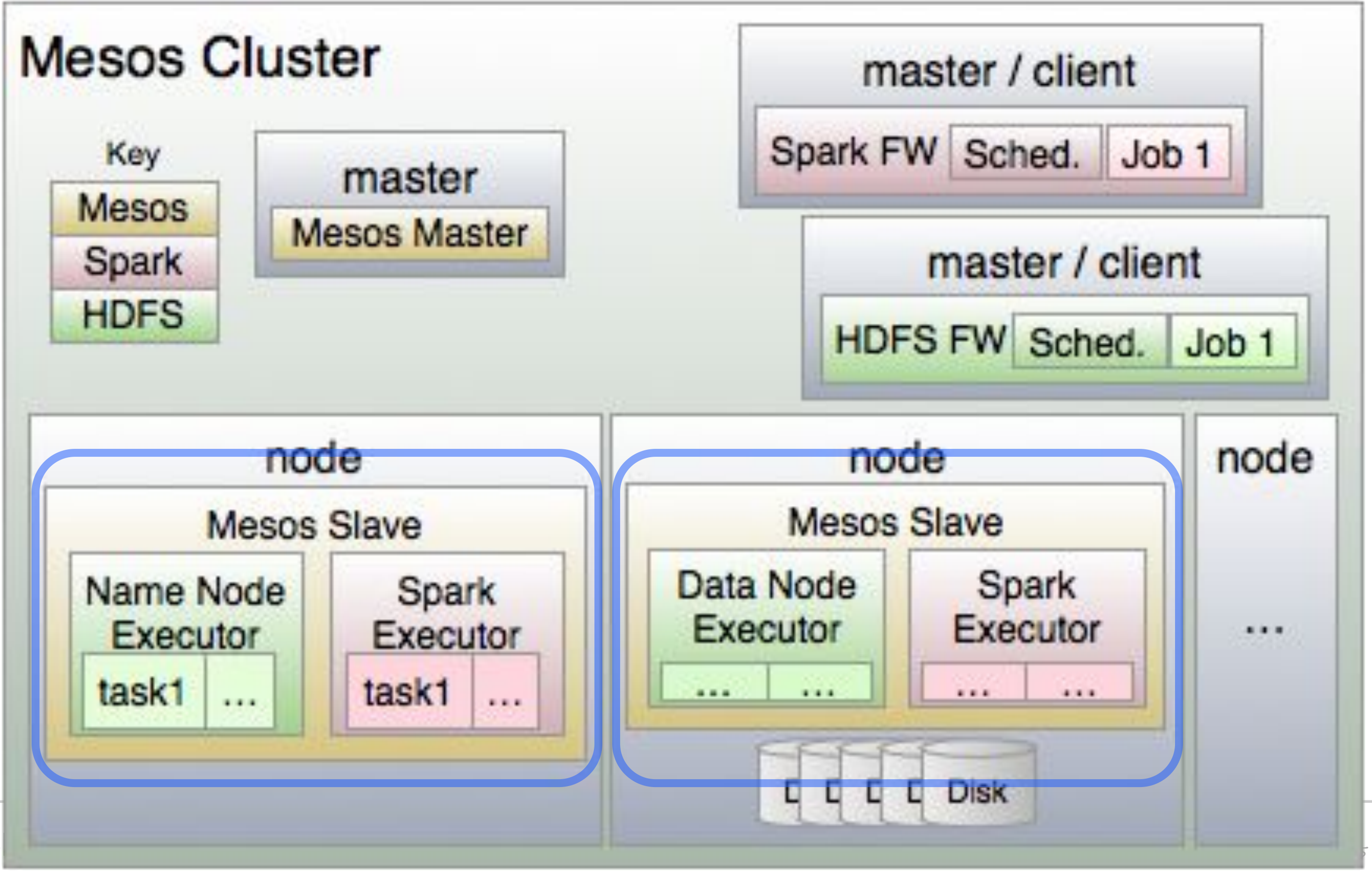
mesosphere


Mesos Cluster
Key
Mesos
Spark
HDFS
master
Mesos Master
master / client
Spark FW
Sched.
Job 1
master / client
HDFS FW
Sched.
Job 1
node
Mesos Slave
Name Node Executor
task1
...
Spark Executor
task1
...
node
Mesos Slave
Data Node Executor
...
...
Spark Executor
...
...
D
D
D
D
Disk
node
...

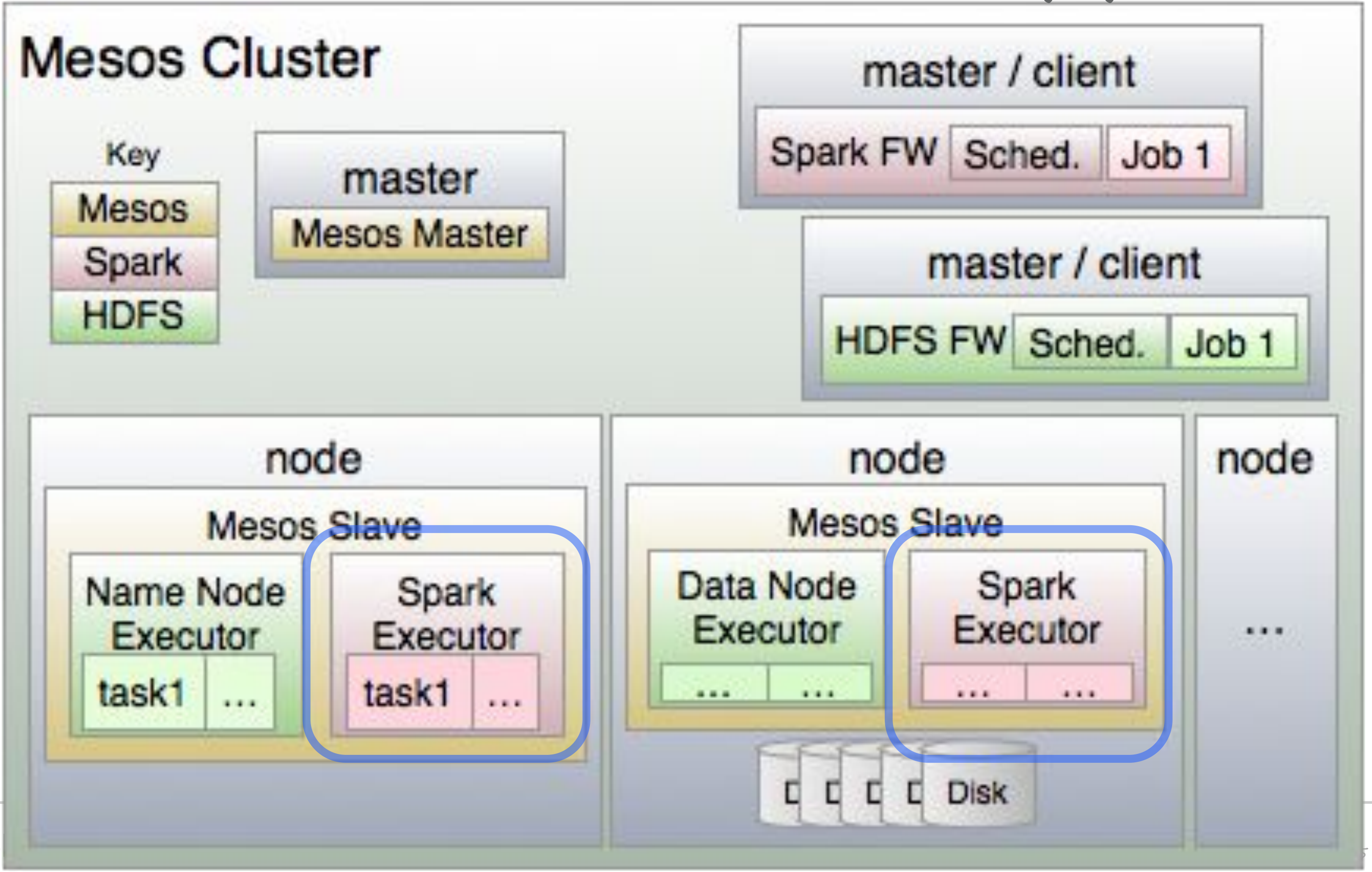
Mesos Master



Mesos Agents



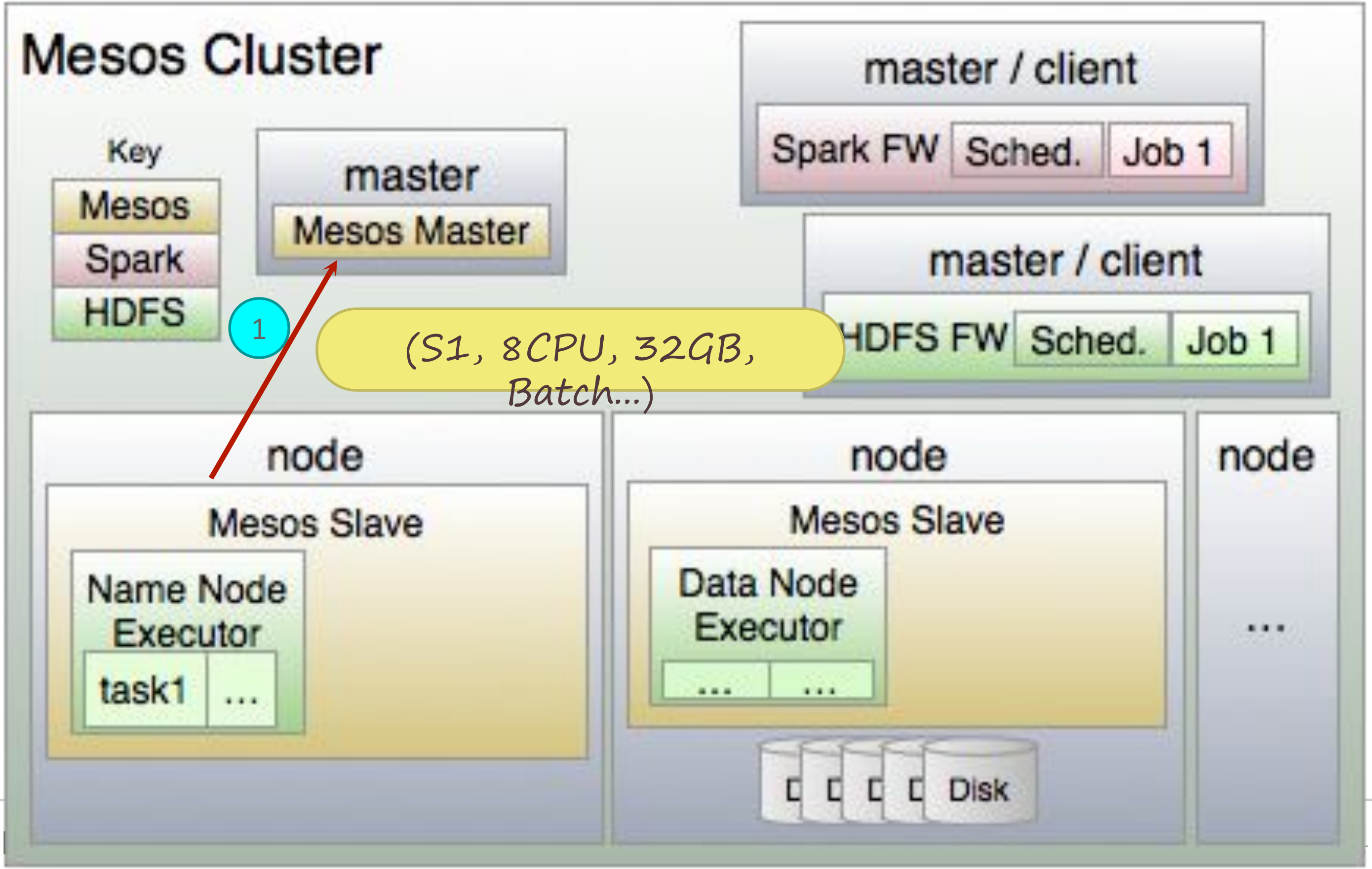
Mesos Executors (Apps)



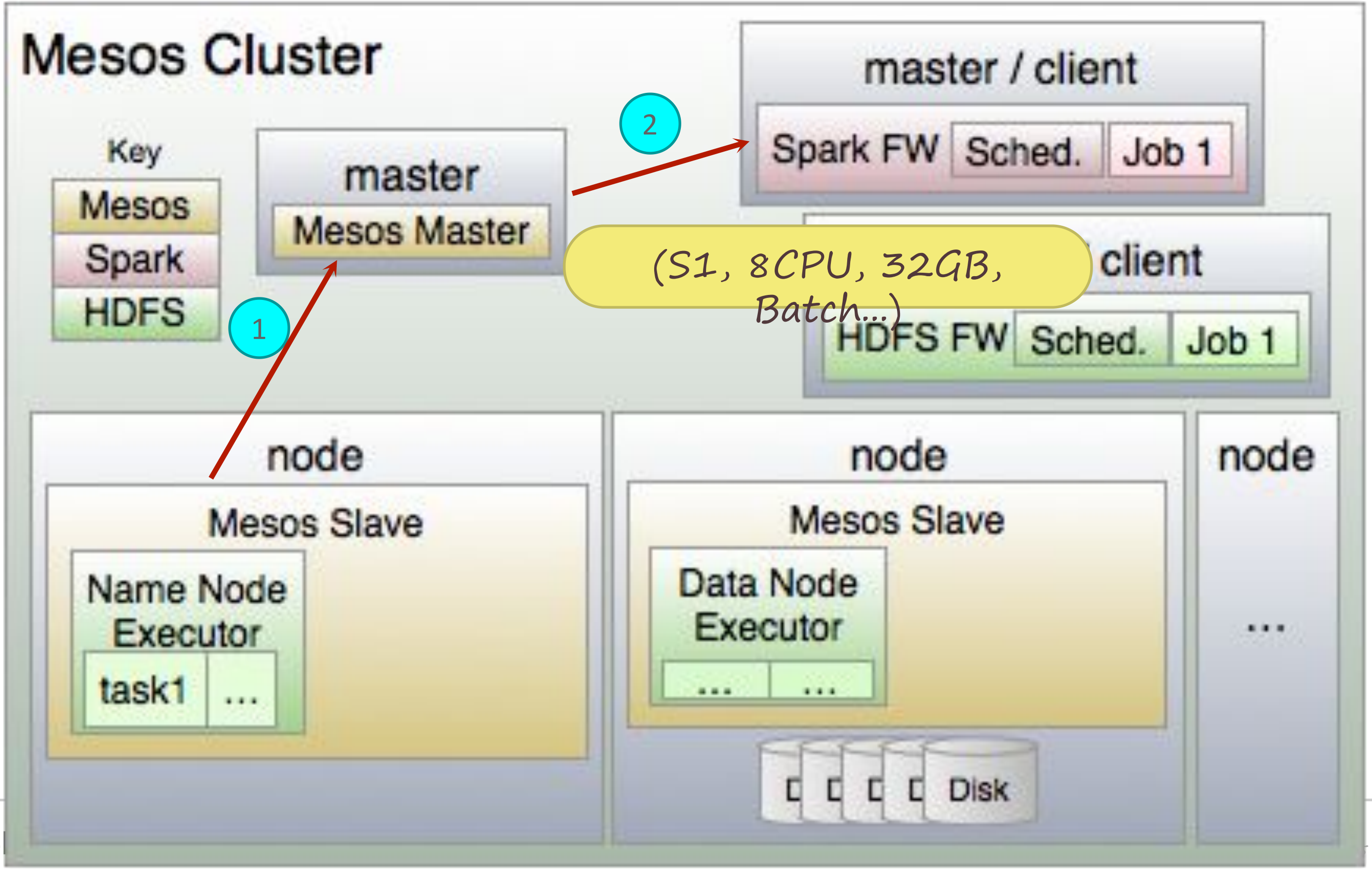
Resources are offered.
They can be refused.

Two-Level Scheduling

Mesos Slaves

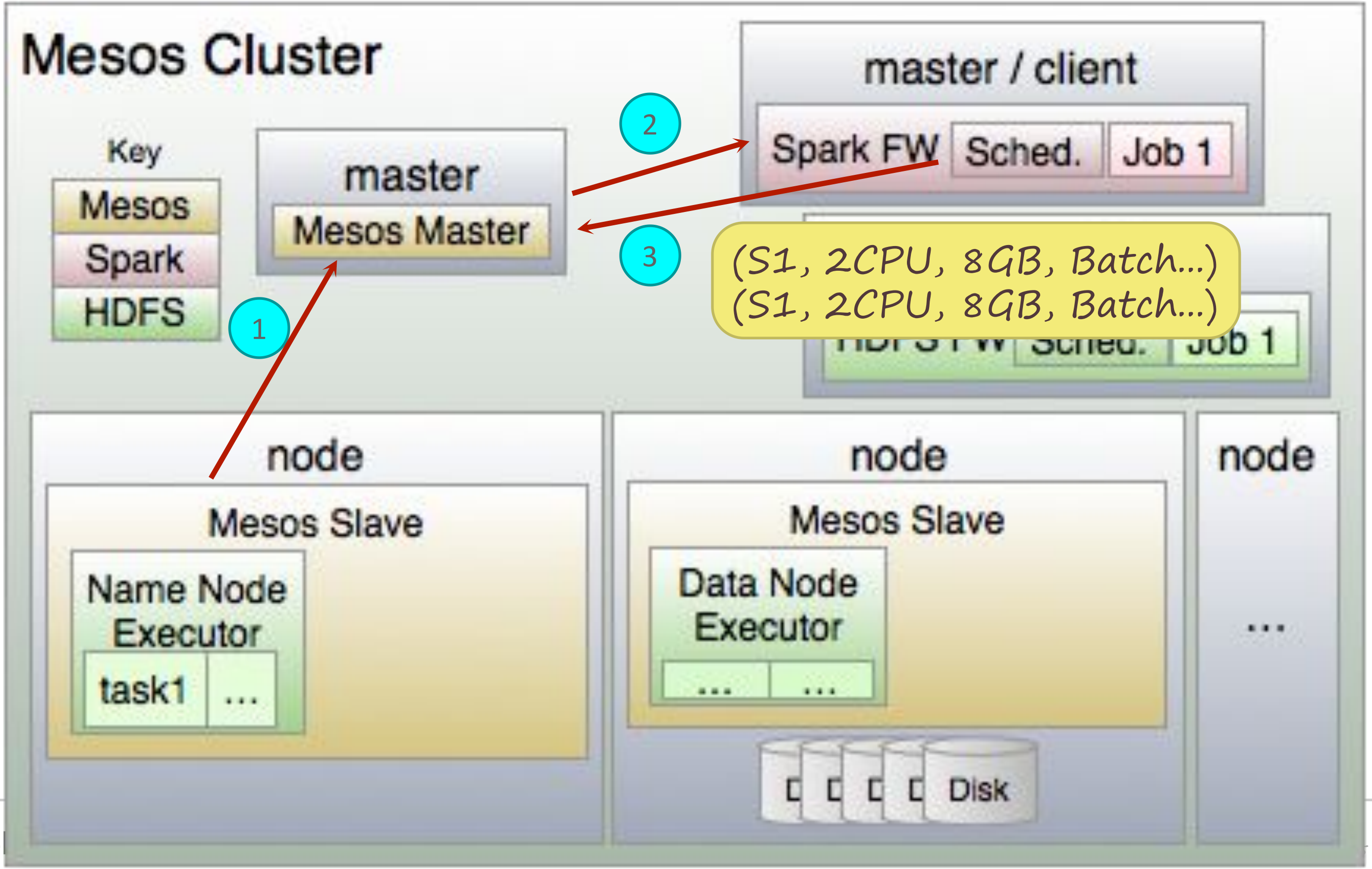


Mesos Slaves



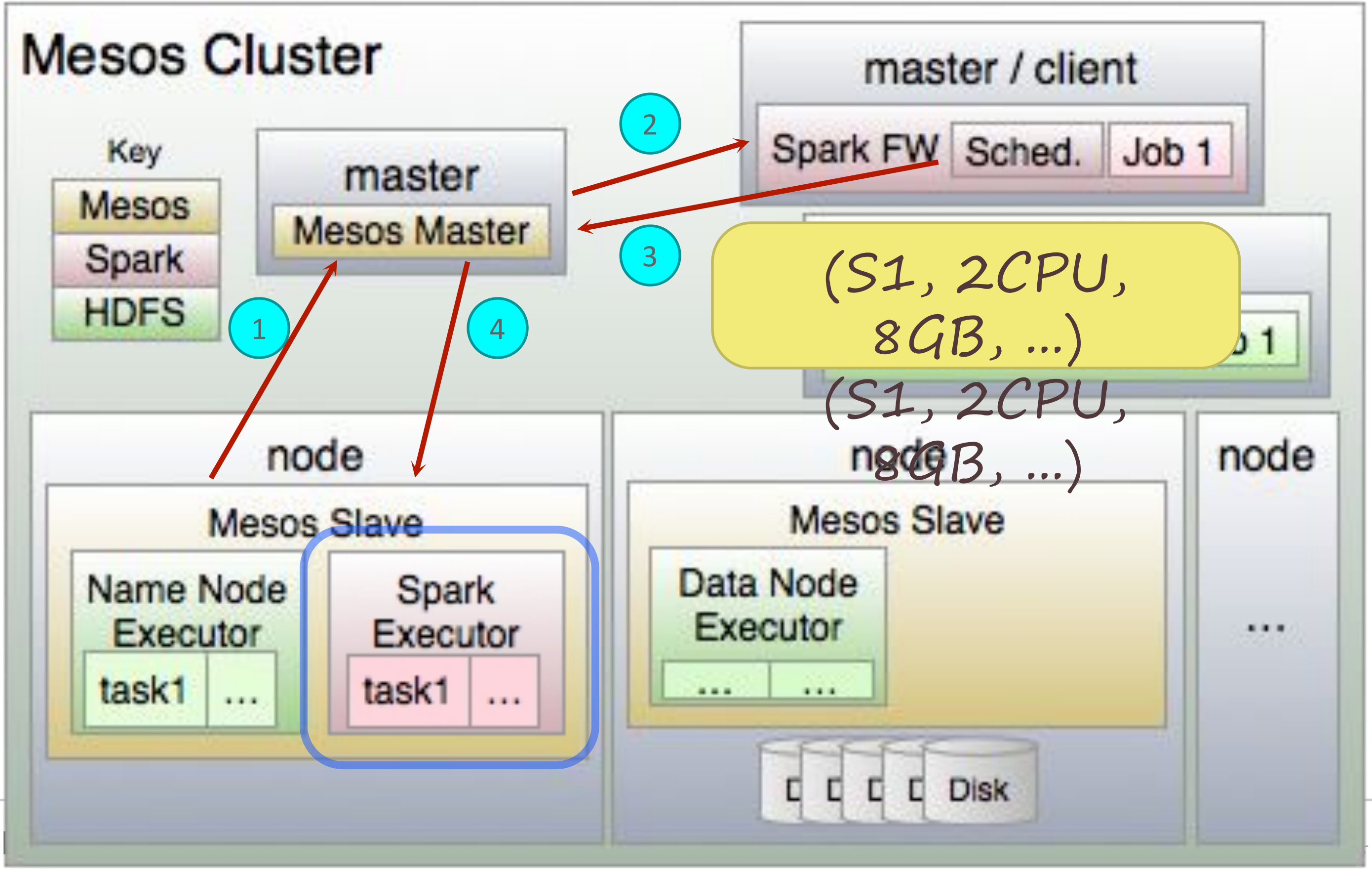
(S1, 8CPU, 32GB, Batch...)

Mesos Slaves

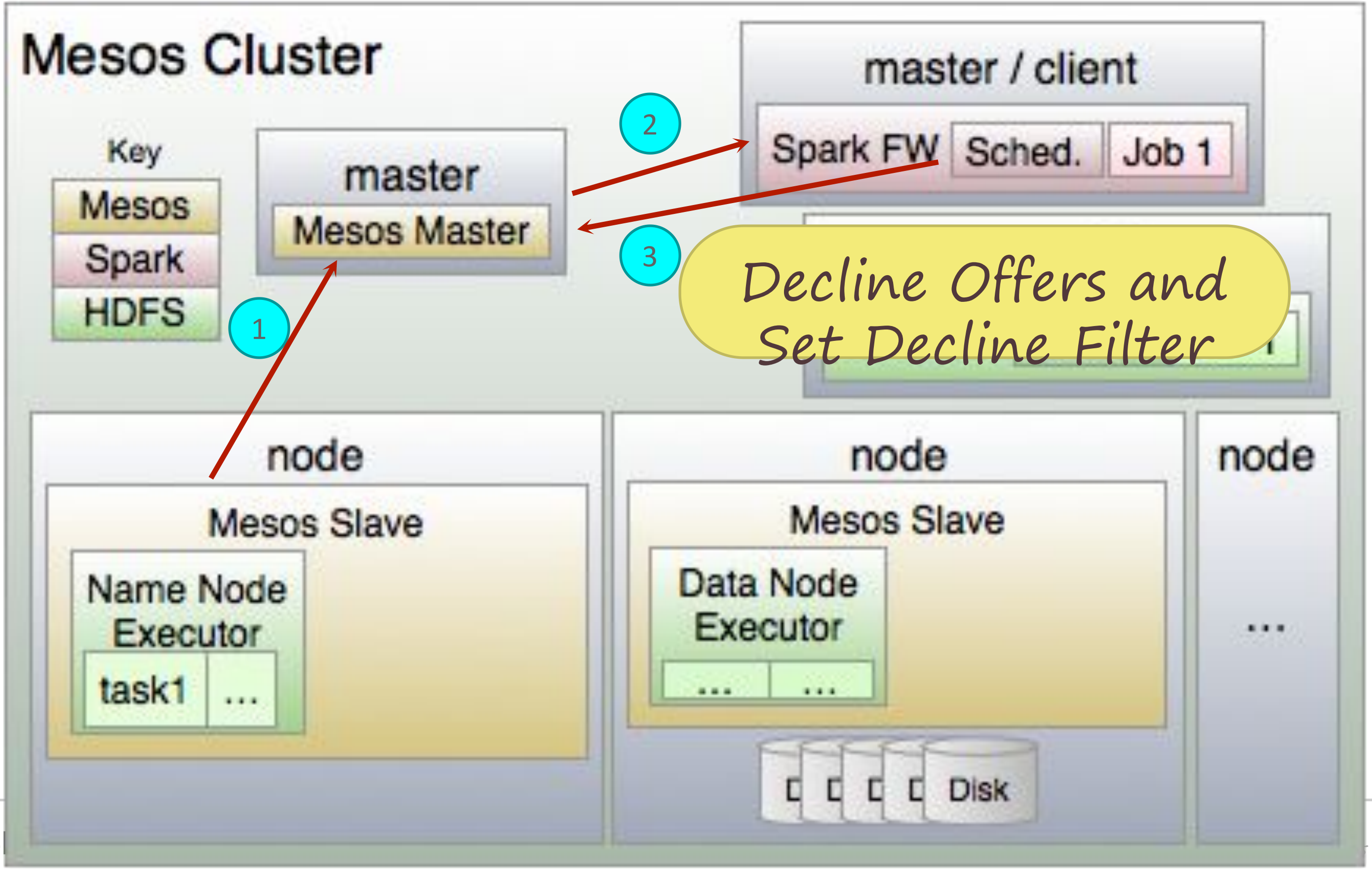


(S1, 2CPU, 8GB, Batch…)
(S1, 2CPU, 8GB, Batch…)

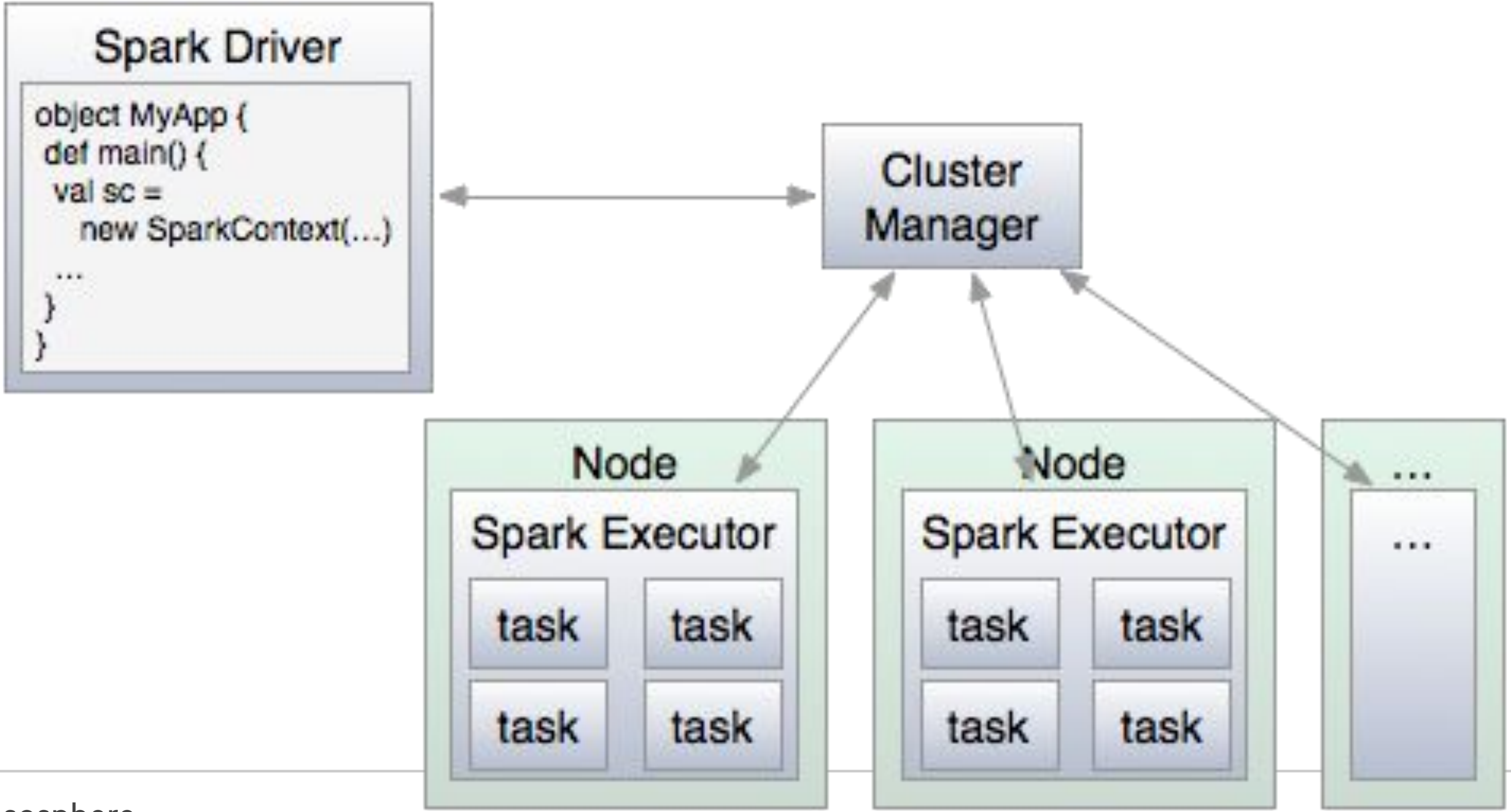
Mesos Slaves



Mesos Slaves



Spark Driver



```
object MyApp {
 def main() {
  val sc =
    new SparkContext(...)
   ...
 }
}
```

mesosphere

Deploying Spark for Mesos

Download on each task

- spark.mesos.executor.uri=http://1.1.1.1/spark-1.5.1-bin.tar.gz

Pre-deploy on each node

- spark.executor.home=/root/spark/

Docker images

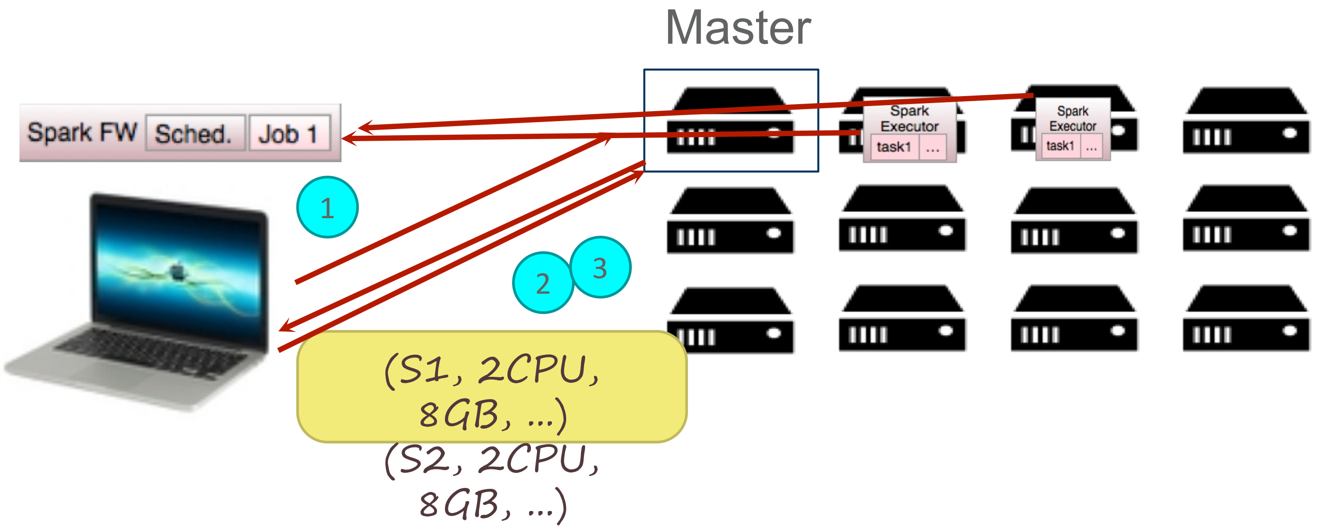
- spark.mesos.executor.docker.image=mesosphere/spark:1.5.1

Spark on Mesos Deploy modes

Client mode vs Cluster mode

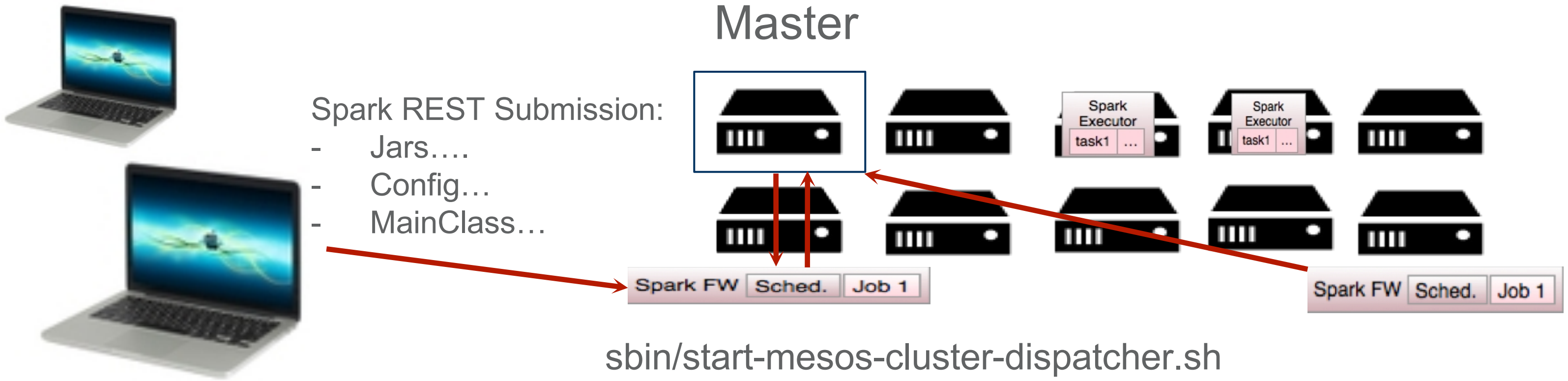
Client mode

spark-submit.sh –deploy-mode client –master mesos://…….



Cluster Mode

spark-submit.sh –deploy-mode cluster –master mesos://…….



Master

Spark REST Submission:

- Jars….
- Config…
- MainClass…

sbin/start-mesos-cluster-dispatcher.sh

Spark on Mesos Run modes

Coarse-grain mode

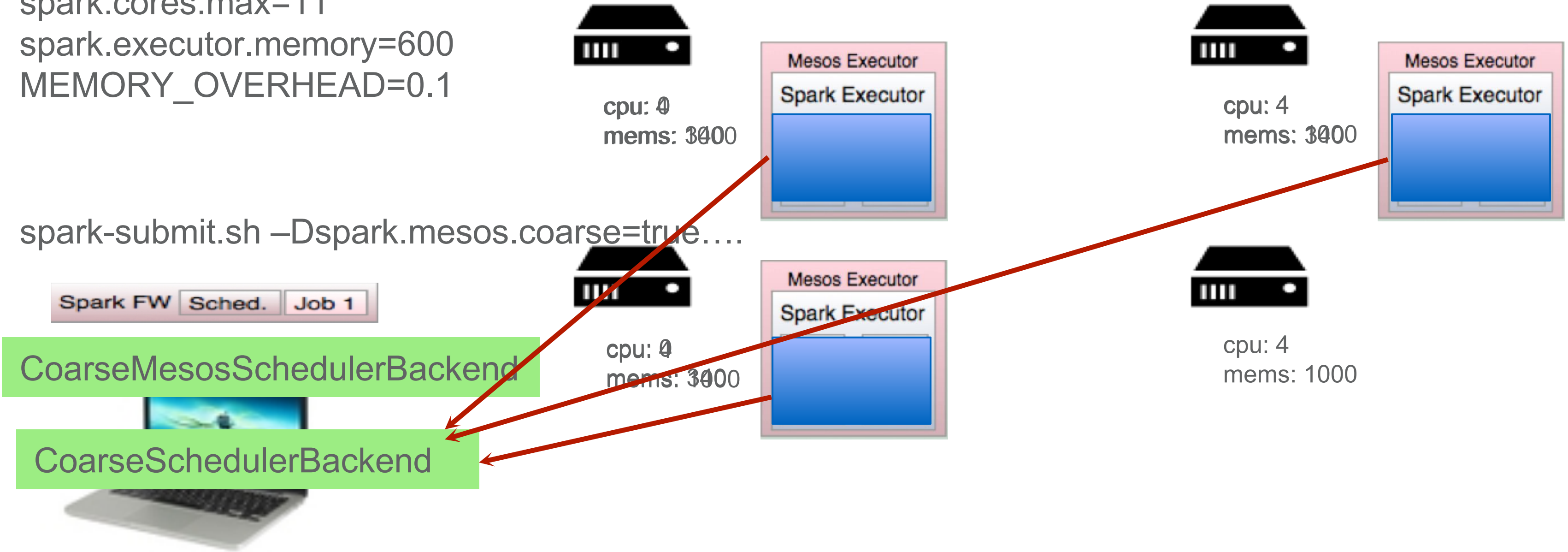
vs

Fine-grain mode

Mesos Coarse Grained Mode

spark.cores.max=11
spark.executor.memory=600
MEMORY_OVERHEAD=0.1

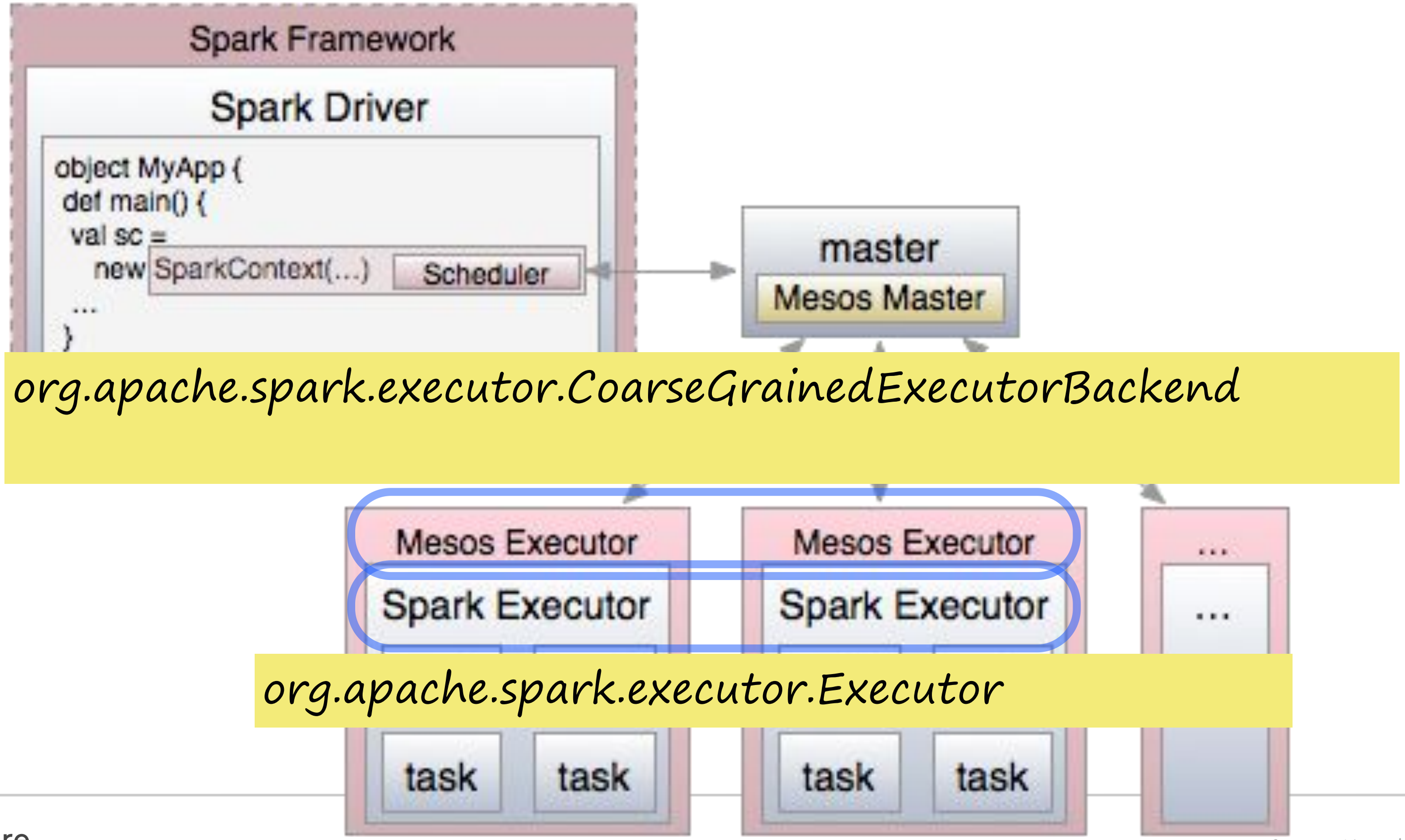
spark-submit.sh –Dspark.mesos.coarse=true….



mesosphere

© 2015 Mesosphere, Inc.

Mesos Coarse Grained Mode

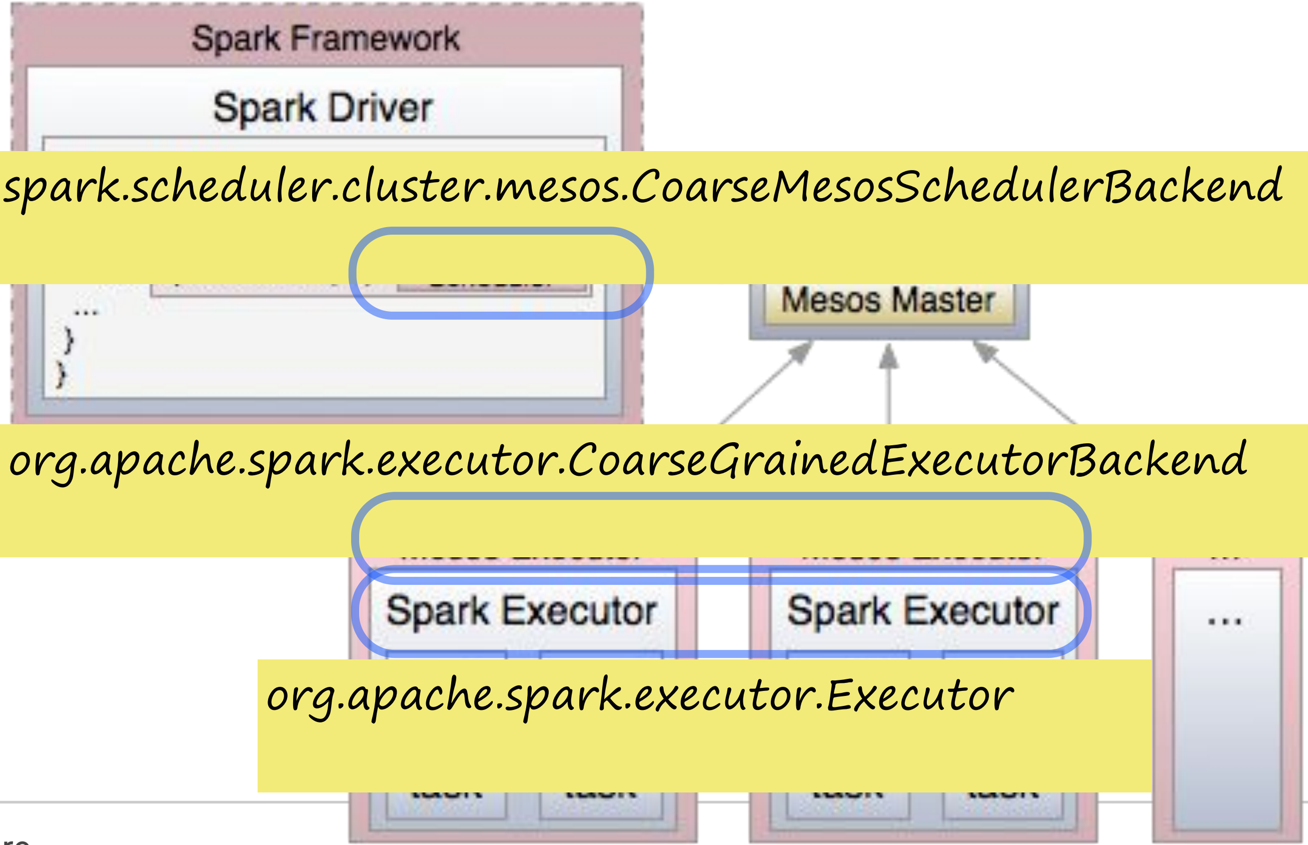


org.apache.spark.executor.CoarseGrainedExecutorBackend

org.apache.spark.executor.Executor

mesosphere

Mesos Coarse Grained Mode



org.apache.spark.scheduler.cluster.mesos.CoarseMesosSchedulerBackend

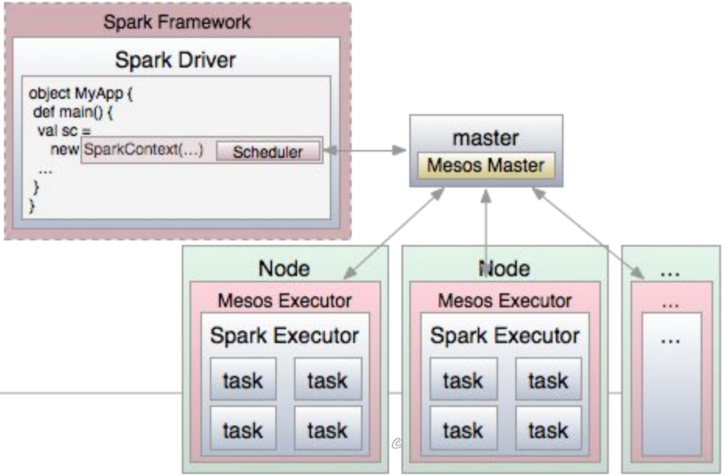
org.apache.spark.executor.CoarseGrainedExecutorBackend

org.apache.spark.executor.Executor

Mesos Coarse Grained Mode

One Mesos and one Spark executor for the job’s lifetime.

Tasks are spawned by Spark itself.



```
object MyApp {
 def main() {
  val sc =
    new SparkContext(...)  Scheduler
  ...
 }
}
```

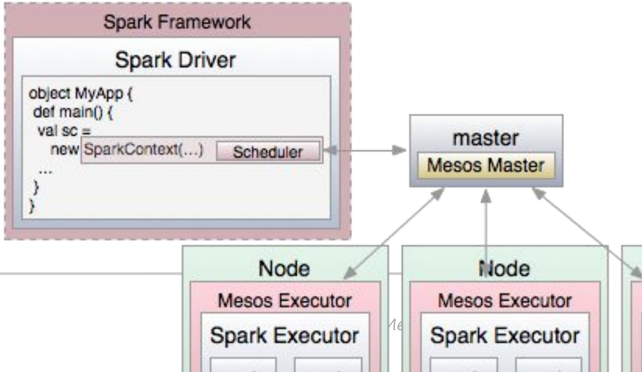
Mesos Coarse Grained Mode

Fast startup for tasks:

- Better for interactive sessions.

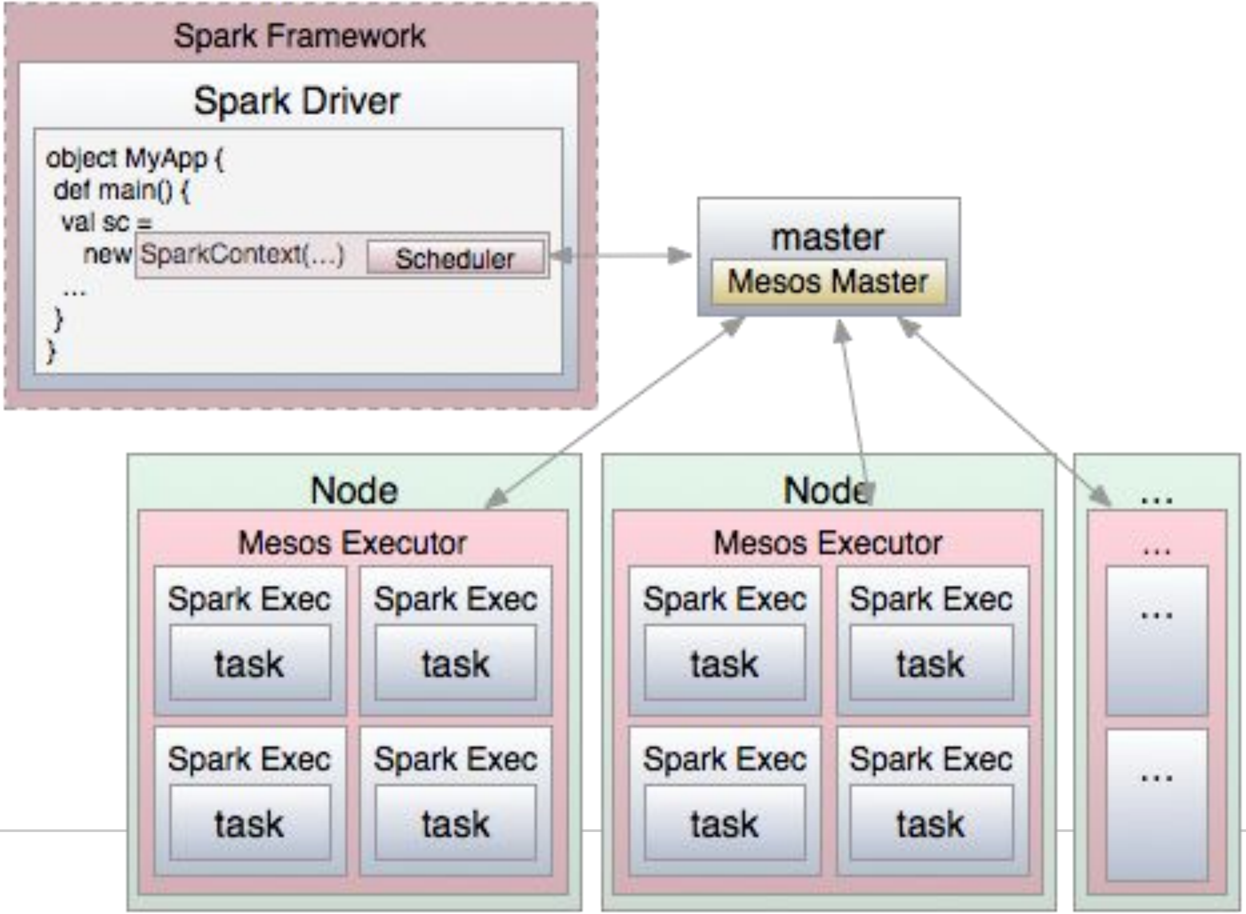
But resources locked up in larger Mesos task.

- Except when using dynamic allocation



```
object MyApp {
 def main() {
  val sc =
   new SparkContext(...)
   ...
 }
}
```

Mesos Fine Grained Mode



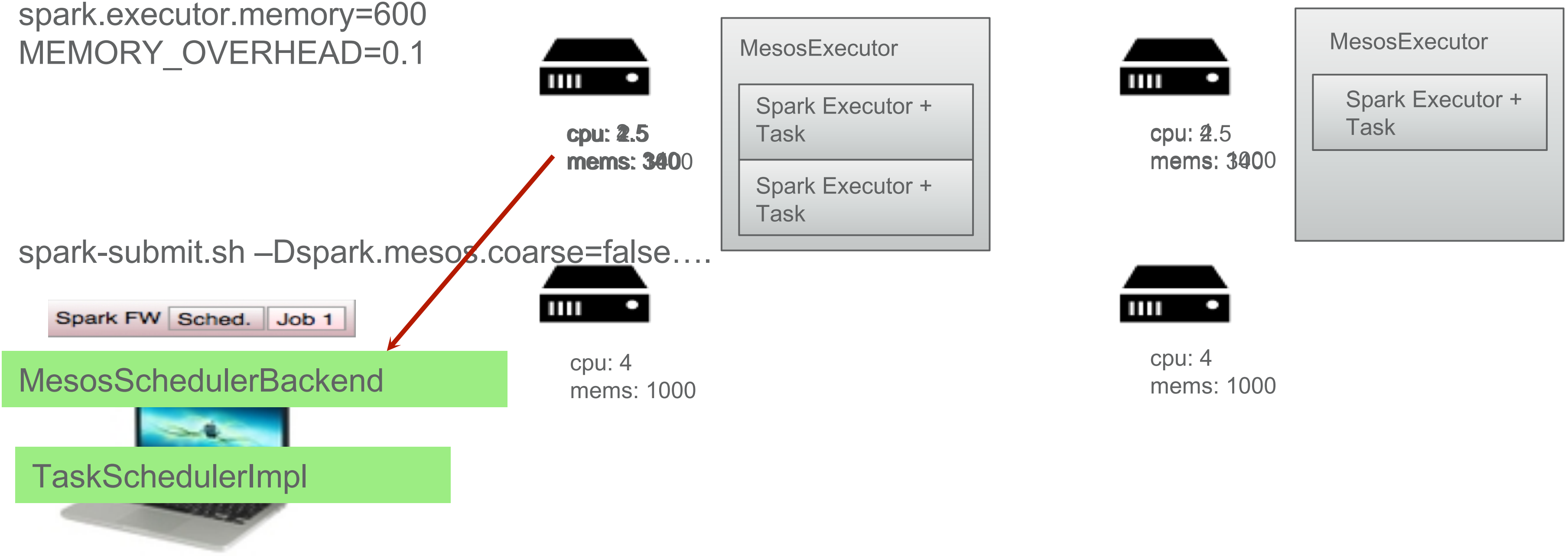
```
object MyApp {
 def main() {
  val sc =
     new SparkContext(...)
  ...
 }
}
```

mesosphere

Mesos Fine Grained Mode

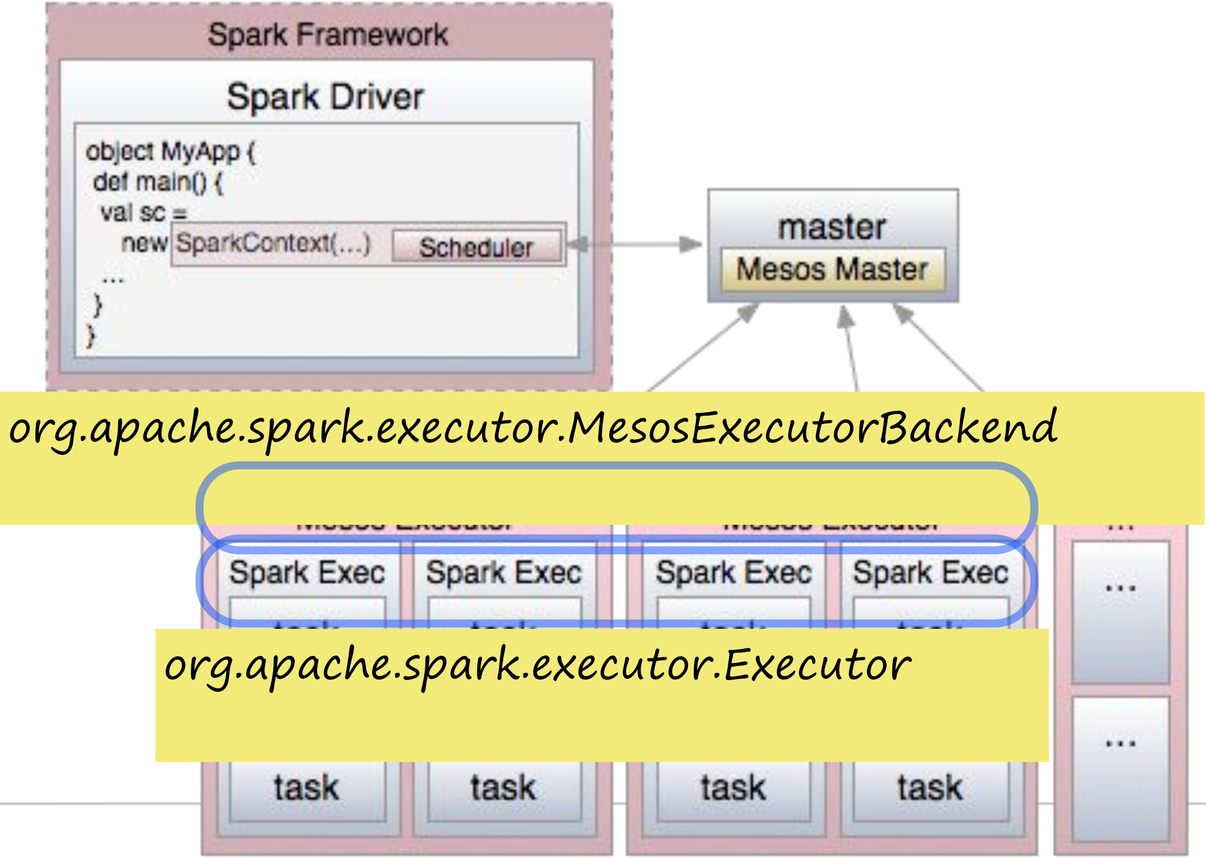
spark.tasks.cpu=1
spark.mesos.mesosExecutor.cores=0.5
spark.executor.memory=600
MEMORY_OVERHEAD=0.1

spark-submit.sh –Dspark.mesos.coarse=false….



mesosphere

Mesos Fine Grained Mode

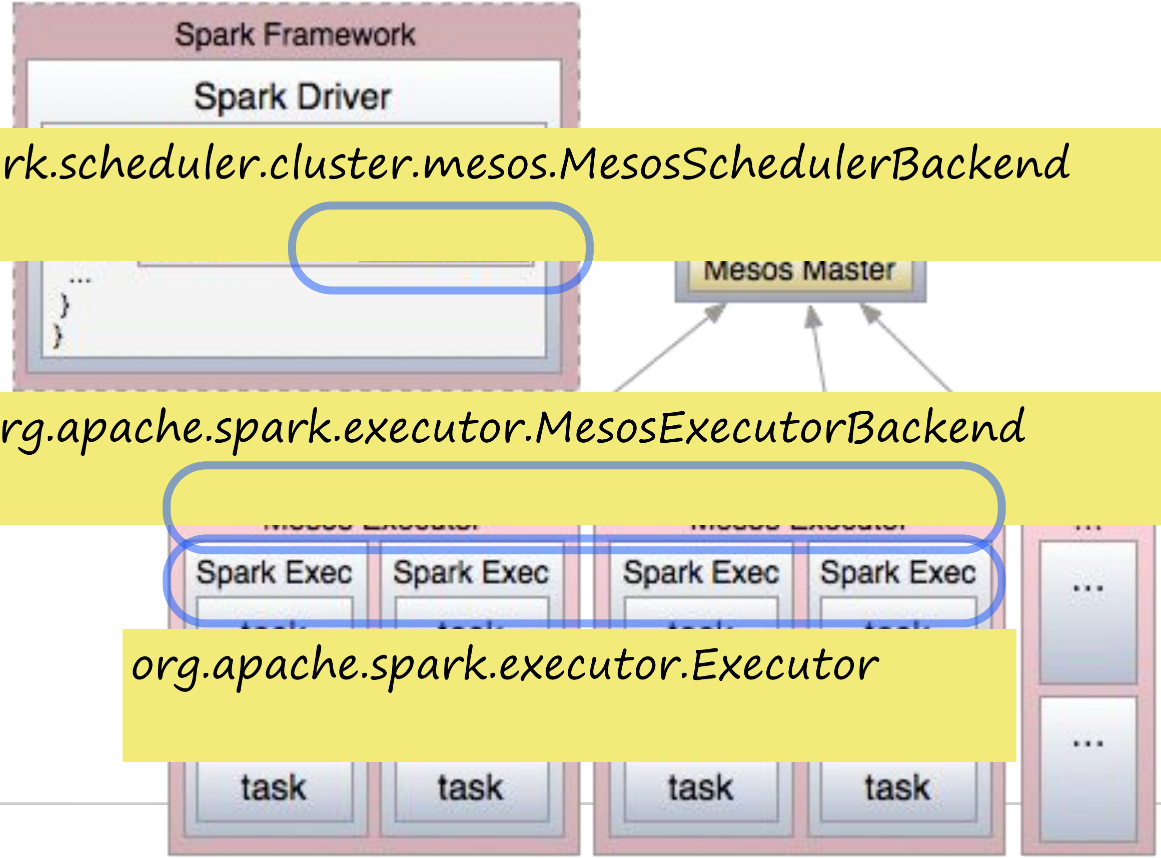


```
object MyApp {
 def main() {
  val sc =
    new SparkContext(...)
   ...
 }
}
```

org.apache.spark.executor.MesosExecutorBackend

org.apache.spark.executor.Executor

Mesos Fine Grained Mode



org.apache.spark.scheduler.cluster.mesos.MesosSchedulerBackend

org.apache.spark.executor.MesosExecutorBackend

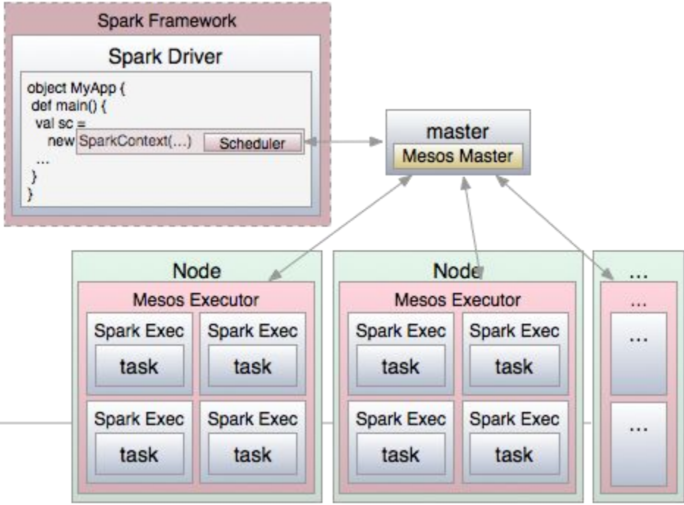
org.apache.spark.executor.Executor

mesosphere

Mesos Fine Grained Mode

One Mesos task per Spark executor.

Spark tasks are spawned as threads.



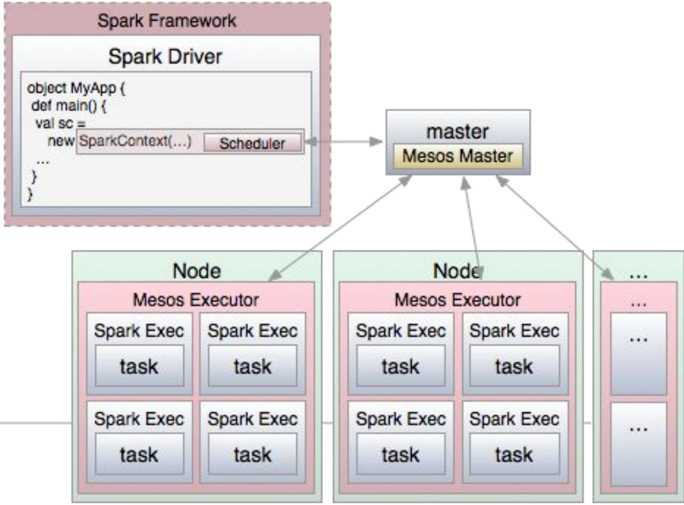
```
object MyApp {
 def main() {
  val sc =
    new SparkContext(...)
   ...
  }
}
```

Mesos Fine Grained Mode

Better resource utilization.

Slower startup for tasks:

- Fine for batch and relatively static streaming.



```
object MyApp {
 def main() {
  val sc =
    new SparkContext(...)
  ...
 }
}
```

Fine & Coarse Grain Mode

Cluster Mode

Docker Support

Constraints / Attributes

Dynamic Allocation

Framework Authentication / Roles

What’s coming next for Spark on Mesos?

Kerberos Authentication

Automated Mesos integration testing

More controls to tune coarse grain scheduler

Preferred location data hinting with dynamic allocation

Support different strategies (binpacking, spread, etc)

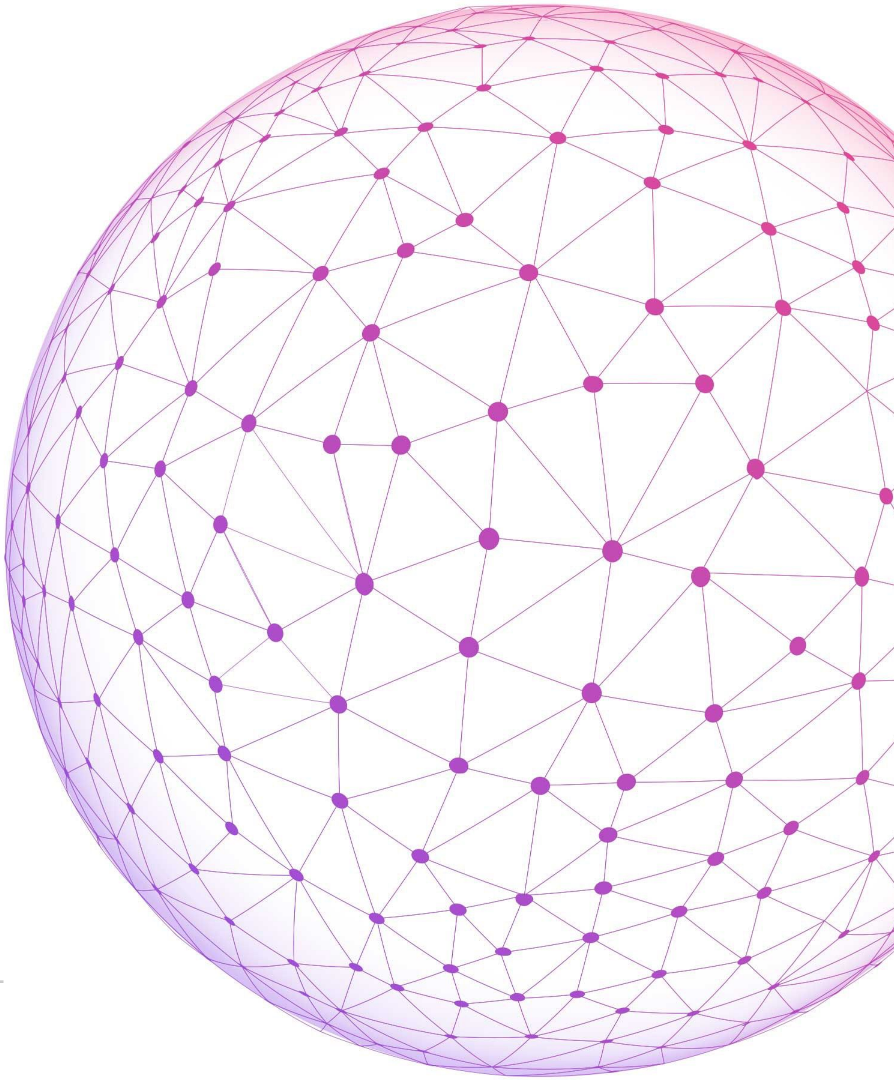
Support Spark shell over cluster mode

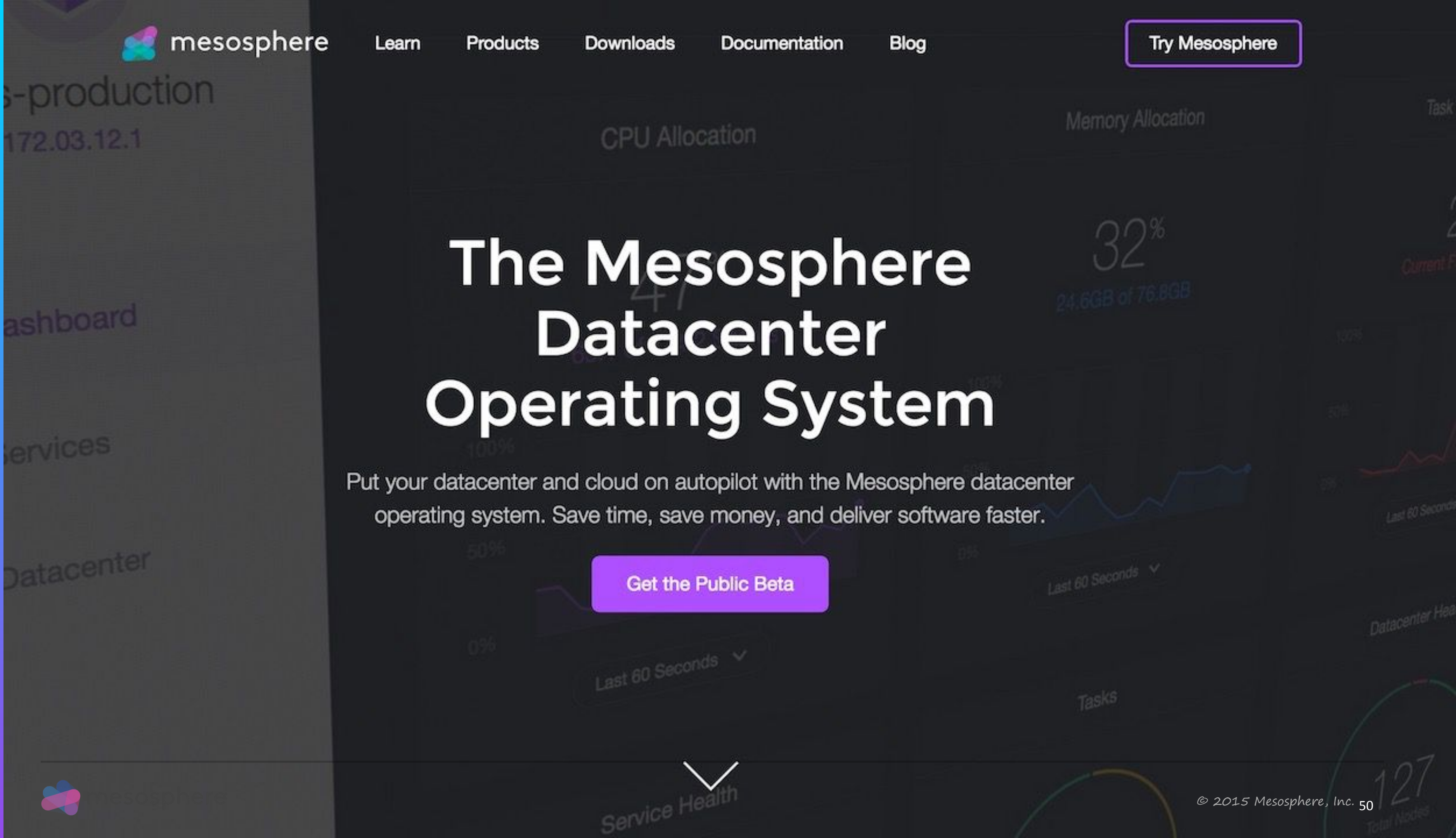
More….

Spark on Mesos

spark.apache.org/docs/latest/running-on-mesos.html

THE DATACENTER IS THE NEW SERVER.





The Mesosphere Datacenter Operating System

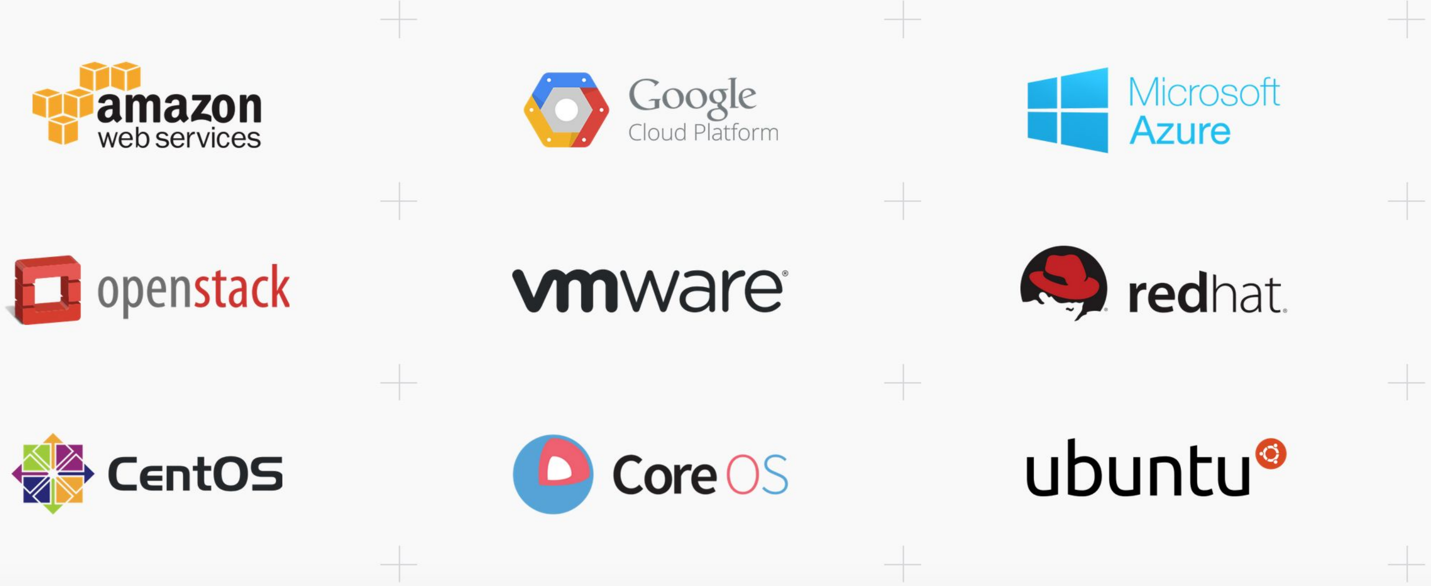
Put your datacenter and cloud on autopilot with the Mesosphere datacenter operating system. Save time, save money, and deliver software faster.

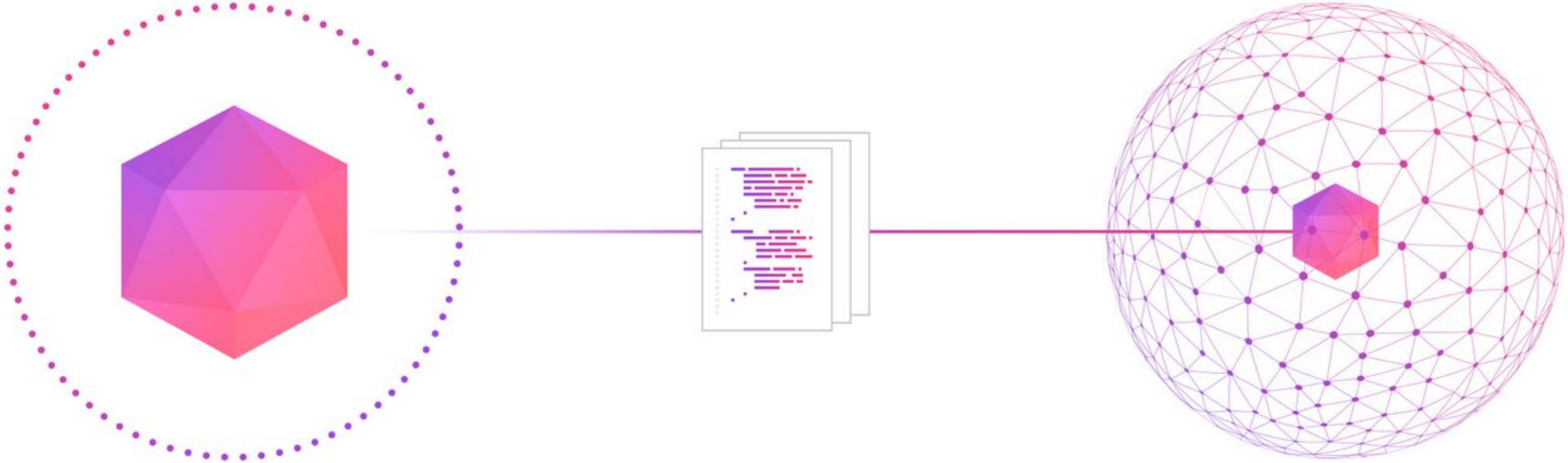
OVER 40 SERVICES MADE FOR DCOS.

DCOS enables single-command installation of services like Hadoop, Spark, Cassandra, Jenkins, Kafka and MemSQL from the DCOS public repository.

WORKS WHERE YOU WORK.

Install Mesosphere DCOS on any public cloud or in your own private datacenter—even a hybrid environment—whether virtualized or on bare metal. Create a consistent user experience and move your workloads with ease.





Mesosphere Universe

What's Next for Mesos?

mesosphere

Oversubscription

Networking

Master Reservations

Optimistic Offers

Isolations

More….

Thanks!

Come and talk to us!
P.S., we're hiring!

mesosphere