Spark on Elastic Mesos, for Enterprise Use Cases

Spark Summit, 2013-12-02

Paco Nathan
http://mesosphere.io
@pacoid

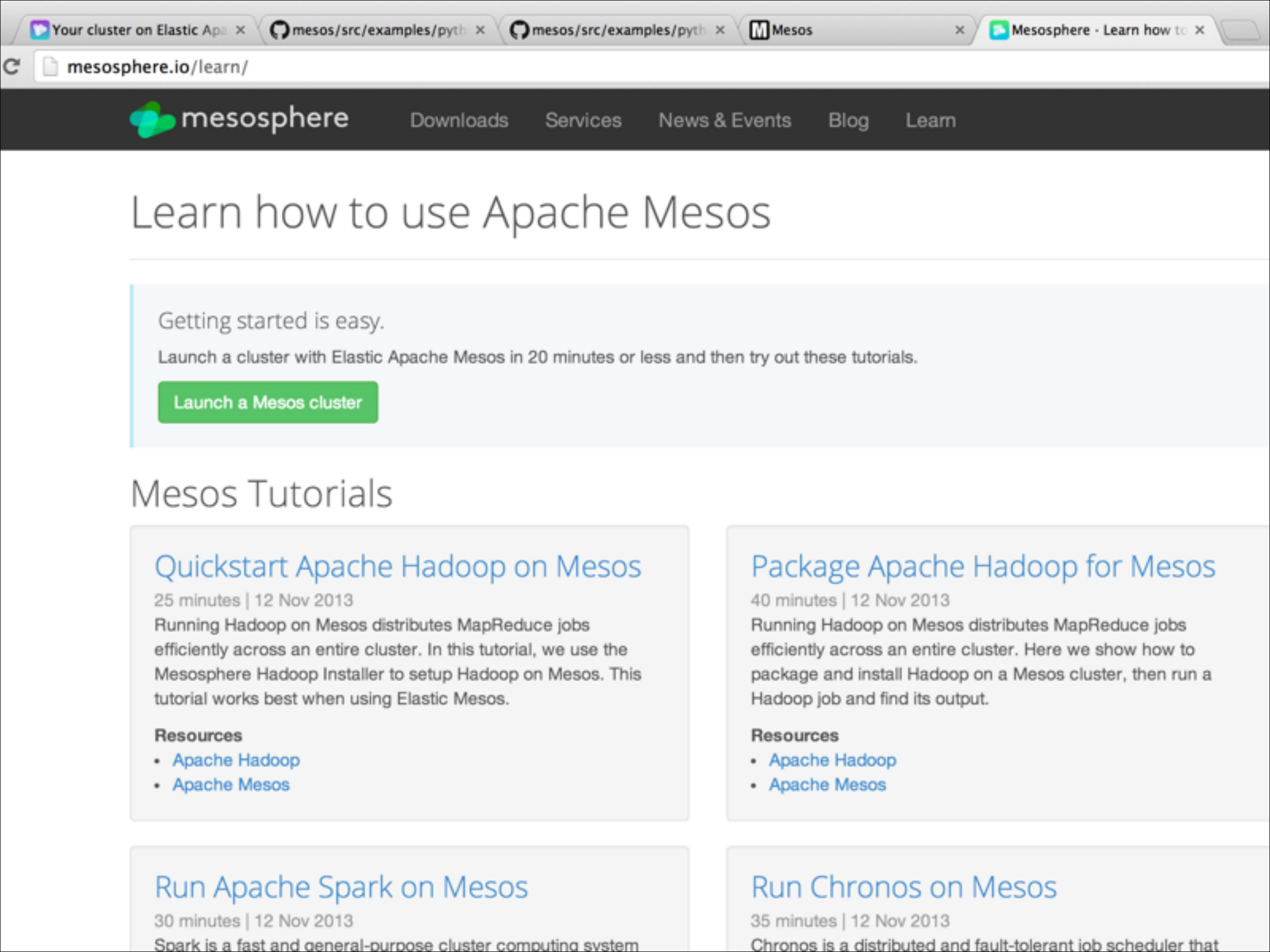
powered by mesosphere

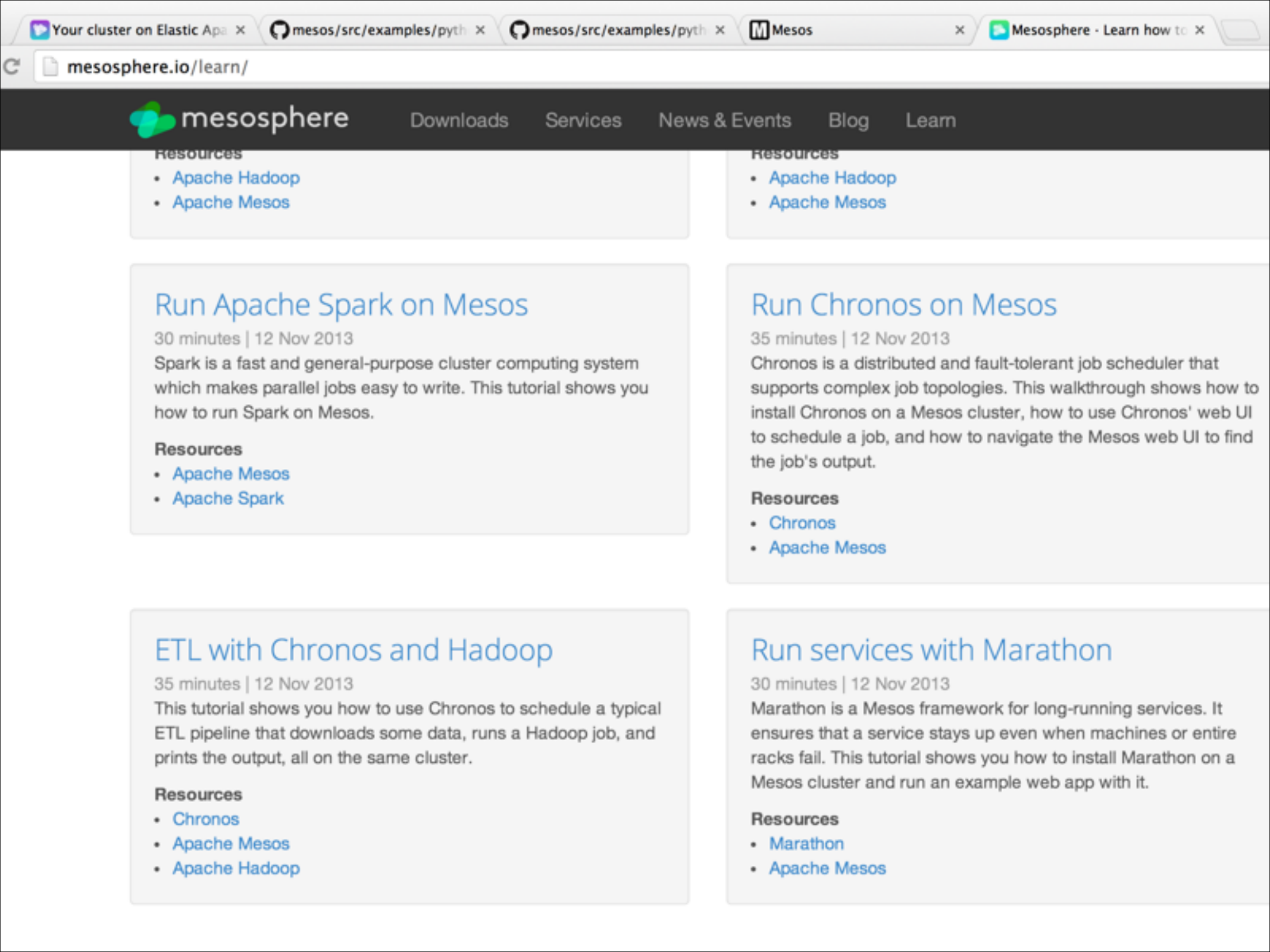
Spark Summit, 2013-12-02

- Learn how to use Mesos
- Spark on Mesos
- PMML for Spark (WIP)

powered by mesosphere

http://mesosphere.io/learn


Your cluster on Elastic Apa
mesos/src/examples/pyth
mesos/src/examples/pyth
Mesos
Mesosphere - Learn how to
mesosphere.io/learn/
mesosphere
Downloads
Services
News & Events
Blog
Learn
Learn how to use Apache Mesos
Getting started is easy.
Launch a cluster with Elastic Apache Mesos in 20 minutes or less and then try out these tutorials.
Launch a Mesos cluster
Mesos Tutorials
Quickstart Apache Hadoop on Mesos
25 minutes | 12 Nov 2013
Running Hadoop on Mesos distributes MapReduce jobs efficiently across an entire cluster. In this tutorial, we use the Mesosphere Hadoop Installer to setup Hadoop on Mesos. This tutorial works best when using Elastic Mesos.
Resources
Apache Hadoop
Apache Mesos
Package Apache Hadoop for Mesos
40 minutes | 12 Nov 2013
Running Hadoop on Mesos distributes MapReduce jobs efficiently across an entire cluster. Here we show how to package and install Hadoop on a Mesos cluster, then run a Hadoop job and find its output.
Resources
Apache Hadoop
Apache Mesos
Run Apache Spark on Mesos
30 minutes | 12 Nov 2013
Spark is a fast and general-purpose cluster computing system
Run Chronos on Mesos
35 minutes | 12 Nov 2013
Chronos is a distributed and fault-tolerant job scheduler that


Your cluster on Elastic Apa
mesos/src/examples/pyth
mesos/src/examples/pyth
Mesos
Mesosphere - Learn how to
mesosphere.io/learn/
mesosphere
Downloads
Services
News & Events
Blog
Learn
Resources
Apache Hadoop
Apache Mesos
Resources
Apache Hadoop
Apache Mesos
Run Apache Spark on Mesos
30 minutes | 12 Nov 2013
Spark is a fast and general-purpose cluster computing system which makes parallel jobs easy to write. This tutorial shows you how to run Spark on Mesos.
Resources
Apache Mesos
Apache Spark
Run Chronos on Mesos
35 minutes | 12 Nov 2013
Chronos is a distributed and fault-tolerant job scheduler that supports complex job topologies. This walkthrough shows how to install Chronos on a Mesos cluster, how to use Chronos' web UI to schedule a job, and how to navigate the Mesos web UI to find the job's output.
Resources
Chronos
Apache Mesos
ETL with Chronos and Hadoop
35 minutes | 12 Nov 2013
This tutorial shows you how to use Chronos to schedule a typical ETL pipeline that downloads some data, runs a Hadoop job, and prints the output, all on the same cluster.
Resources
Chronos
Apache Mesos
Apache Hadoop
Run services with Marathon
30 minutes | 12 Nov 2013
Marathon is a Mesos framework for long-running services. It ensures that a service stays up even when machines or entire racks fail. This tutorial shows you how to install Marathon on a Mesos cluster and run an example web app with it.
Resources
Marathon
Apache Mesos

Why use Mesos?

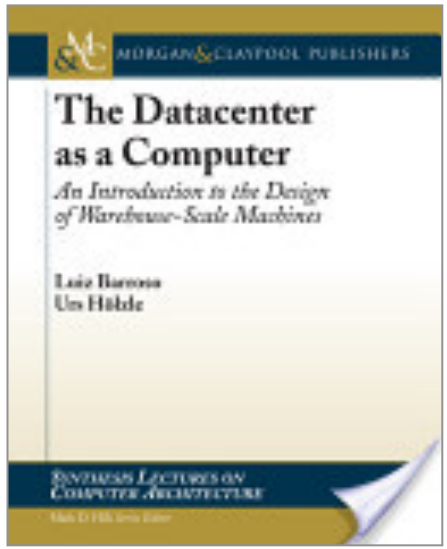
"Return of the Borg"

Google has been doing *datacenter computing* for years,
to address the complexities of large-scale data workflows:

- leveraging the modern kernel: isolation in lieu of VMs
- "most (>80%) jobs are batch jobs, but the majority of resources (55–80%) are allocated to service jobs"
- mixed workloads, multi-tenancy
- relatively high utilization rates
- JVM? not so much…
- reality: scheduling batch is simple; scheduling services is hard/expensive

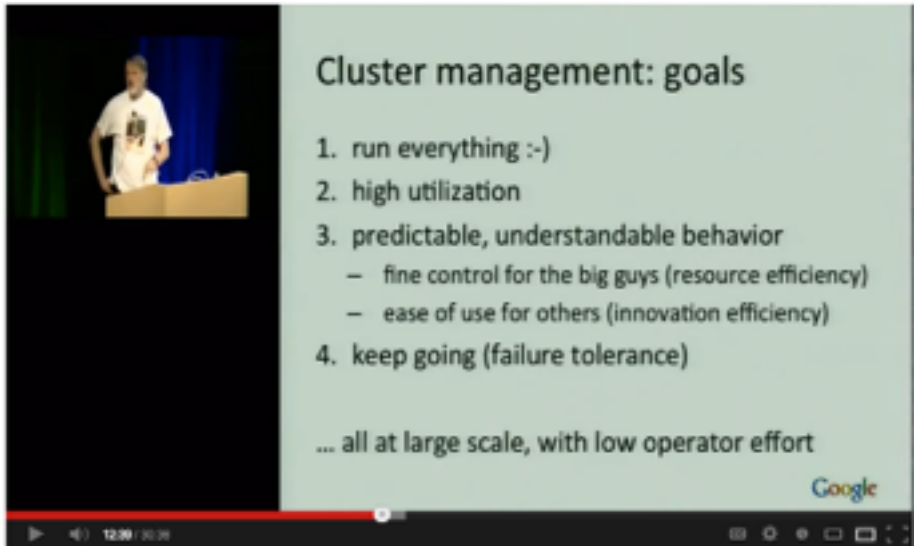
"Return of the Borg"

Return of the Borg: How Twitter Rebuilt Google's Secret Weapon
Cade Metz
wired.com/wiredenterprise/2013/03/google-borg-twitter-mesos



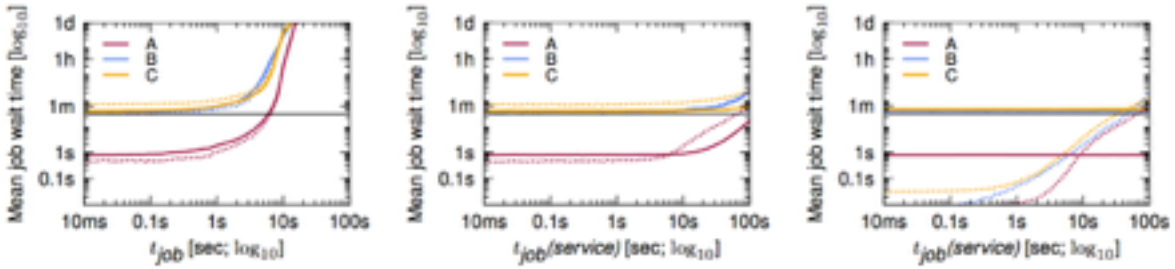
The Datacenter as a Computer: An Introduction to the Design of Warehouse-Scale Machines
Luiz André Barroso, Urs Hölzle
research.google.com/pubs/pub35290.html

2011 GAFS Omega
John Wilkes, et al.
youtu.be/0ZFMlO98Jkc



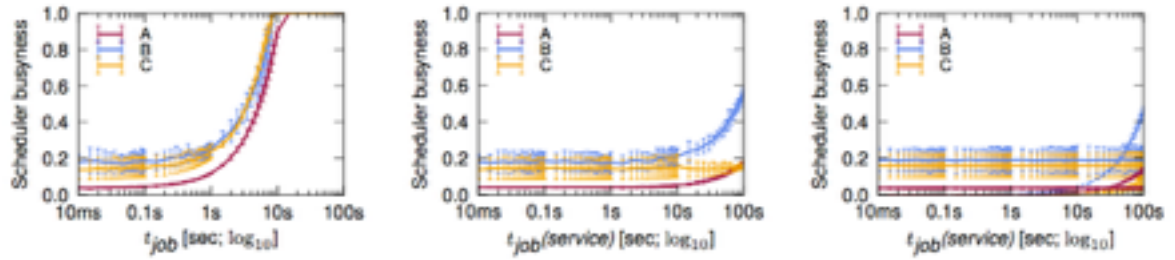
“Return of the Borg”

Omega: flexible, scalable schedulers for large compute clusters
Malte Schwarzkopf, Andy Konwinski, Michael Abd-El-Malek, John Wilkes
eurosys2013.tudos.org/wp-content/uploads/2013/paper/Schwarzkopf.pdf



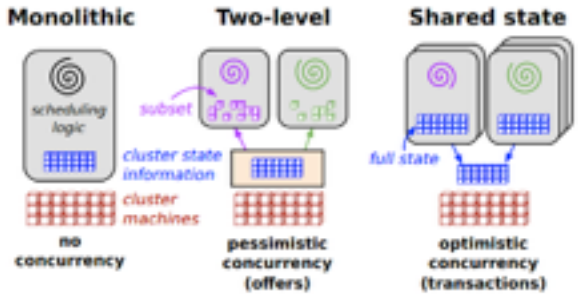
(a) Single-path. (b) Multi-path. (c) Shared state.

Figure 5: Schedulers' job wait time, as a function of t_{job} in the monolithic single-path case, $t_{job}(service)$ in the monolithic multi-path and shared-state cases. The SLO (horizontal bar) is 30s.



(a) Single-path. (b) Multi-path. (c) Shared state.

Figure 6: Schedulers' busyness, as a function of t_{job} in the monolithic single-path case, $t_{job}(service)$ in the monolithic multi-path and shared-state cases. The value is the median daily busyness over the 7-day experiment, and error bars are one ± median absolute deviation (MAD), i.e. the median deviation from the median value, a robust estimator of typical value dispersion.



Google describes the business case…

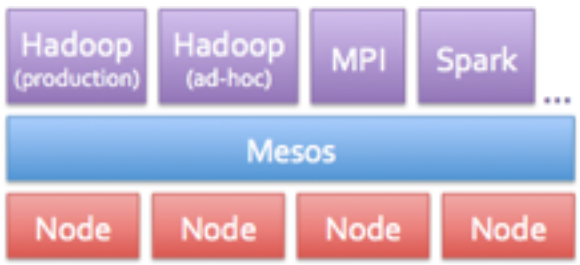
Taming Latency Variability
Jeff Dean
plus.google.com/u/0/+ResearchatGoogle/posts/C1dPhQhcDRv

Mesos – definitions

a common substrate for cluster computing

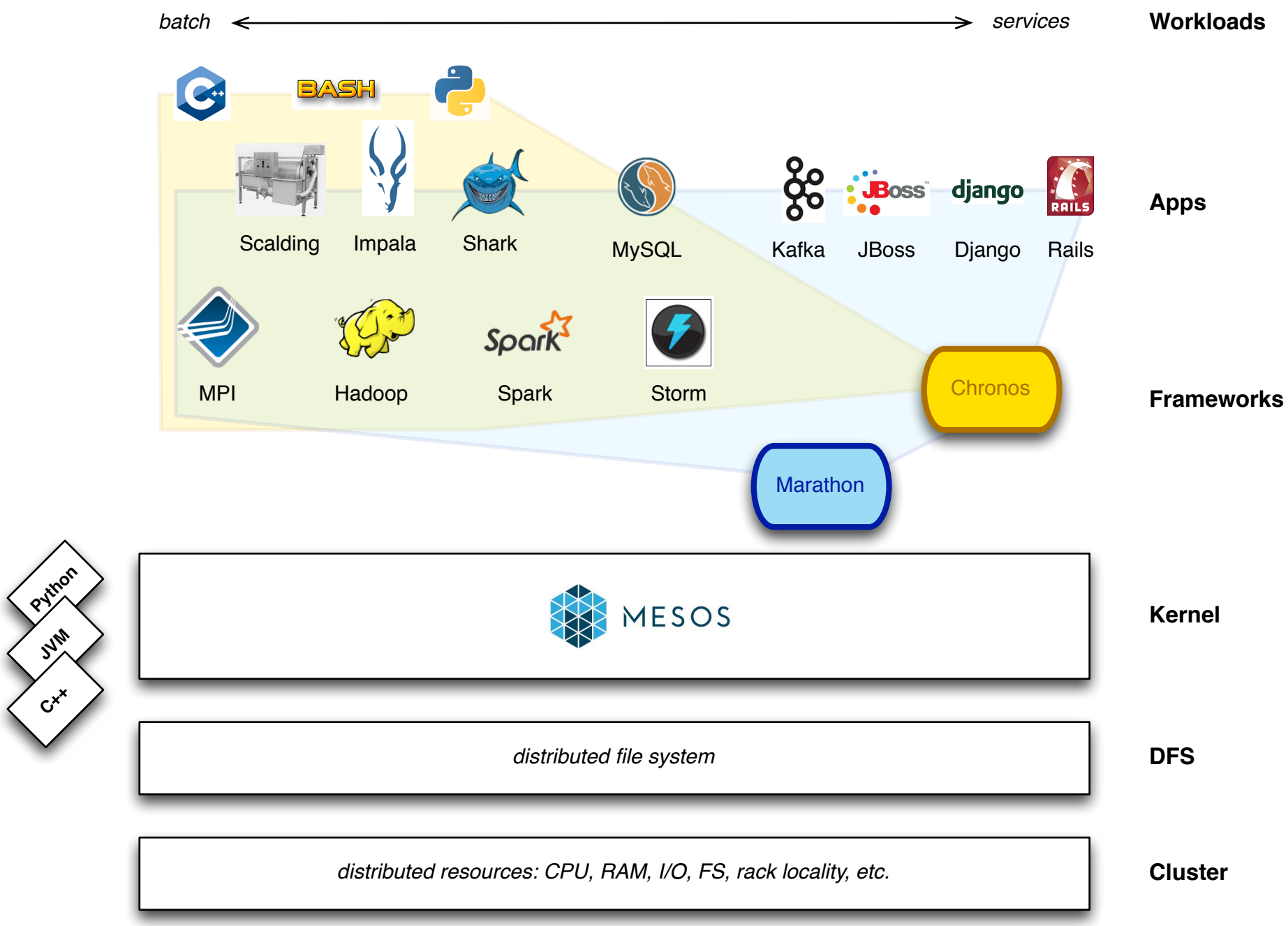
mesos.apache.org

heterogenous assets in your datacenter or cloud
made available as a homogenous set of resources



- top-level Apache project
- scalability to 10,000s of nodes
- obviates the need for virtual machines
- isolation (pluggable) for CPU, RAM, I/O, FS, etc.
- fault-tolerant leader election based on Zookeeper
- APIs in C++, Java, Python
- web UI for inspecting cluster state
- available for Linux, OpenSolaris, Mac OSX

Mesos – architecture



Production Deployments (public)

vimeo

MediaCrossing™
BRIDGING THE MARKET™

xogito
[kä-gi-tō]

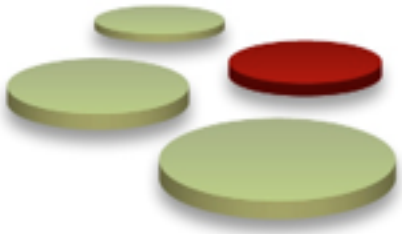
HubSpot

sharethrough

device scape™

iQIYI 爱奇艺

CLOUD PHYSICS



OpenTable®

airbnb™
Travel like a human.

CATEGORIZE

Case Study: Twitter (bare metal / on premise)

"Mesos is the cornerstone of our elastic compute infrastructure – it's how we build all our new services and is critical for Twitter's continued success at scale. It's one of the primary keys to our data center efficiency."

Chris Fry, SVP Engineering

blog.twitter.com/2013/mesos-graduates-from-apache-incubation

wired.com/gadgetlab/2013/11/qa-with-chris-fry/

- key services run in production: analytics, typeahead, ads
- Twitter engineers rely on Mesos to build all new services
- instead of thinking about static machines, engineers think about resources like CPU, memory and disk
- allows services to scale and leverage a shared pool of servers across datacenters efficiently
- reduces the time between prototyping and launching

Resources

Apache Mesos Project
mesos.apache.org

Twitter
@ApacheMesos

Mesosphere
mesosphere.io

Tutorials
mesosphere.io/learn

Documentation
mesos.apache.org/documentation

2011 USENIX Research Paper
usenix.org/legacy/event/nsdi11/tech/full_papers/Hindman_new.pdf

Collected Notes/Archives
goo.gl/jPtTP

Spark Summit, 2013-12-02

- Learn how to use Mesos
- **Spark on Mesos**
- PMML for Spark (WIP)

powered by mesosphere

http://elastic.mesosphere.io

Elastic Apache Mesos

https://elastic.mesosphere.io

Launch an Apache Mesos Cluster in 3 2 1

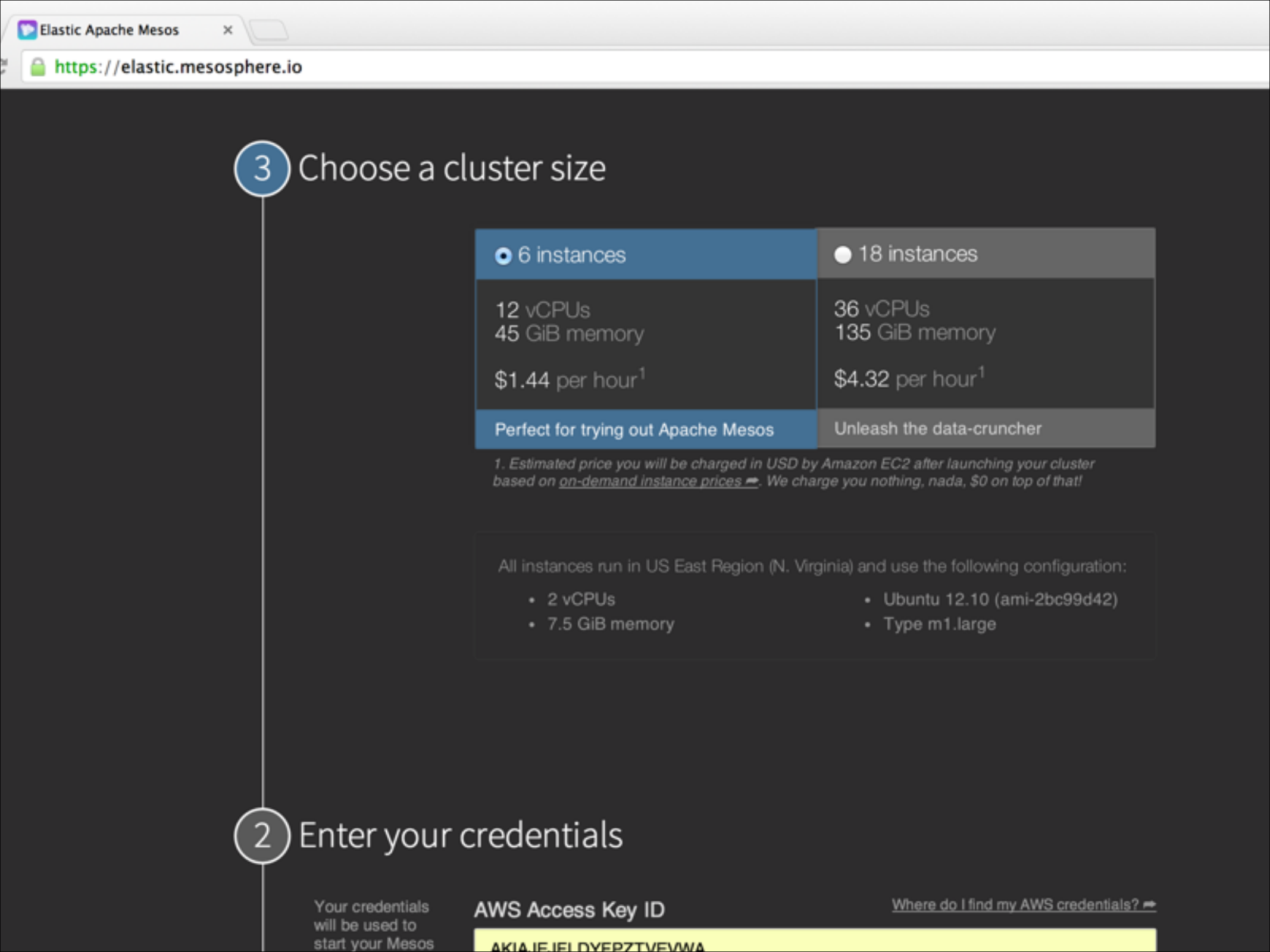
Elastic Apache Mesos is a web service that automates the creation of Apache Mesos clusters on Amazon Elastic Compute Cloud (EC2). It provisions EC2 instances, installs dependencies including Apache ZooKeeper and HDFS, and delivers you a cluster with all the services running.

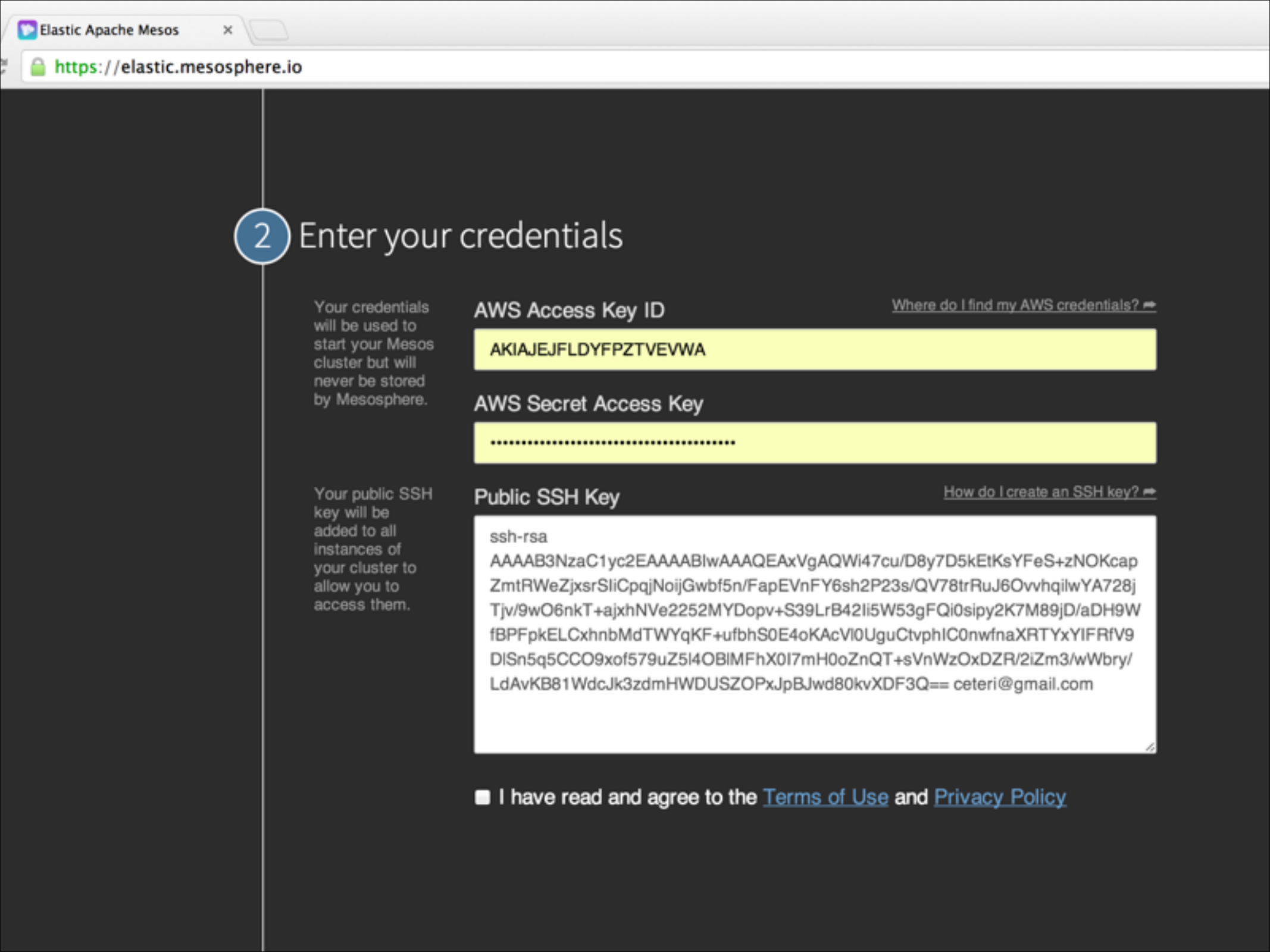
Mesos allows you to easily share compute resources and data between frameworks like Apache Hadoop and Apache Spark.

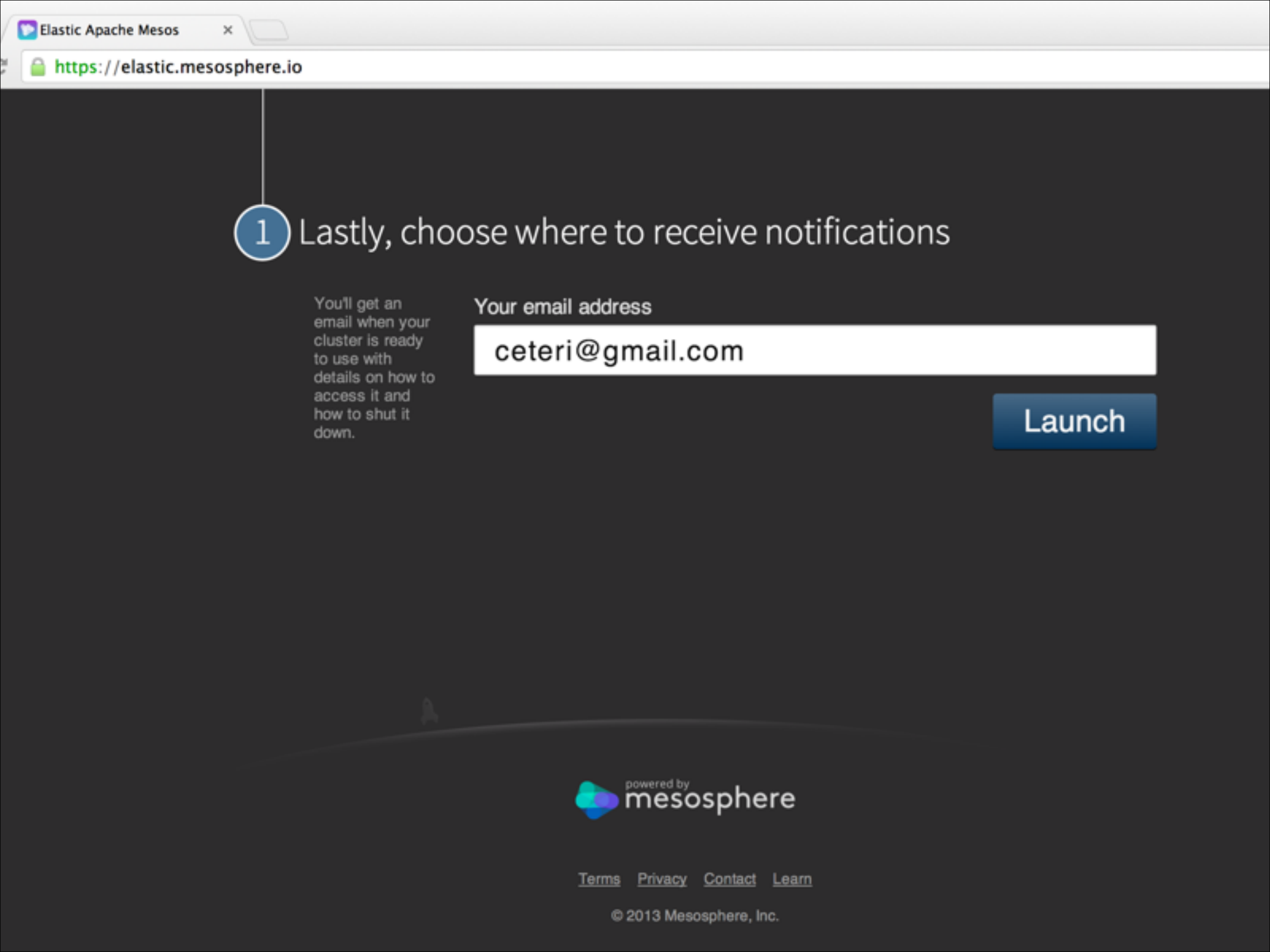
You just pay for your EC2 instances; Elastic Apache Mesos costs you nothing, nada, zilch on top of that.

3 Choose a cluster size

● 6 instances	● 18 instances
12 vCPUs 45 GiB memory	36 vCPUs 135 GiB memory
$1.44 per hour[1]	$4.32 per hour[1]


Elastic Apache Mesos
https://elastic.mesosphere.io
3 Choose a cluster size
6 instances
18 instances
12 vCPUs
45 GiB memory
$1.44 per hour[1]
36 vCPUs
135 GiB memory
$4.32 per hour[1]
Perfect for trying out Apache Mesos
Unleash the data-cruncher
1. Estimated price you will be charged in USD by Amazon EC2 after launching your cluster based on on-demand instance prices. We charge you nothing, nada, $0 on top of that!
All instances run in US East Region (N. Virginia) and use the following configuration:
2 vCPUs
7.5 GiB memory
Ubuntu 12.10 (ami-2bc99d42)
Type m1.large
2 Enter your credentials
Your credentials will be used to start your Mesos
AWS Access Key ID
Where do I find my AWS credentials?


Elastic Apache Mesos
https://elastic.mesosphere.io
2
Enter your credentials
Your credentials will be used to start your Mesos cluster but will never be stored by Mesosphere.
AWS Access Key ID
Where do I find my AWS credentials?
AKIAJEJFLDYFPZTVEVWA
AWS Secret Access Key
Your public SSH key will be added to all instances of your cluster to allow you to access them.
Public SSH Key
How do I create an SSH key?
ssh-rsa
AAAAB3NzaC1yc2EAAAABIwAAAQEAxVgAQWi47cu/D8y7D5kEtKsYFeS+zNOKcap
ZmtRWeZjxsrSIiCpqjNoijGwbf5n/FapEVnFY6sh2P23s/QV78trRuJ6OvvhqilwYA728j
Tjv/9wO6nkT+ajxhNVe2252MYDopv+S39LrB42Ii5W53gFQi0sipy2K7M89jD/aDH9W
fBPFpkELCxhnbMdTWYqKF+ufbhS0E4oKAcVl0UguCtvphIC0nwfnaXRTYxYIFRfV9
DlSn5q5CCO9xof579uZ5l4OBlMFhX0I7mH0oZnQT+sVnWzOxDZR/2iZm3/wWbry/
LdAvKB81WdcJk3zdmHWDUSZOPxJpBJwd80kvXDF3Q== ceteri@gmail.com
I have read and agree to the Terms of Use and Privacy Policy


Elastic Apache Mesos
https://elastic.mesosphere.io
1
Lastly, choose where to receive notifications
You'll get an email when your cluster is ready to use with details on how to access it and how to shut it down.
Your email address
ceteri@gmail.com
Launch
powered by mesosphere
Terms
Privacy
Contact
Learn
© 2013 Mesosphere, Inc.

More

Success! Your Apache Mesos cluster is ready

Inbox x

The Mesosphere Team support@mesosphere.io via mail4.wdc04.manc 3:20 PM (2 minutes ago)

to me

Great news,

Your Apache Mesos cluster is up and running!

View running frameworks and tasks in your Mesos UI:
http://54.235.3.97:5050

To get started with Hadoop on Mesos, visit the Hadoop on Mesos quickstart tutorial.
For more advanced Hadoop use, visit the Package Hadoop for Mesos tutorial.
For fast in-memory number crunching, try the Apache Spark on Mesos tutorial.
To run repeating jobs via Chronos, visit the Chronos on Mesos tutorial.
To learn how to build ETL pipelines, try ETL with Chronos and Hadoop.
And to run your long-lived services, visit the Marathon tutorial.

The IP addresses of your Mesos master nodes are: 54.235.3.97 54.204.193.157 54.204.133.148
Your Mesos slave nodes are: 54.234.5.237 54.221.134.235 54.227.20.164

You can connect to your instances via ssh using the "ubuntu" user.

Shutdown your cluster on Elastic Apache Mesos:
https://elastic.mesosphere.io/clusters/54.235.3.97

View running instances on AWS:
https://console.aws.amazon.com/ec2/v2/home?region=us-east-1

Have fun,
The Mesosphere Team
http://mesosphere.io/

More

Success! Your Apache Mesos cluster is ready

Inbox x

The Mesosphere Team support@mesosphere.io via mail4.wdc04.manc 3:20 PM (2 minutes ago)

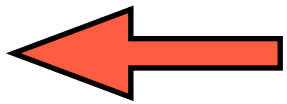
to me

Great news,

Your Apache Mesos cluster is up and running!

View running frameworks and tasks in your Mesos UI:
http://54.235.3.97:5050

To get started with Hadoop on Mesos, visit the Hadoop on Mesos quickstart tutorial.
For more advanced Hadoop use, visit the Package Hadoop for Mesos tutorial.
For fast in-memory number crunching, try the Apache Spark on Mesos tutorial.
To run repeating jobs via Chronos, visit the Chronos on Mesos tutorial.
To learn how to build ETL pipelines, try ETL with Chronos and Hadoop.
And to run your long-lived services, visit the Marathon tutorial.



The IP addresses of your Mesos master nodes are: 54.235.3.97 54.204.193.157 54.204.133.148
Your Mesos slave nodes are: 54.234.5.237 54.221.134.235 54.227.20.164

You can connect to your instances via ssh using the "ubuntu" user.

Shutdown your cluster on Elastic Apache Mesos:
https://elastic.mesosphere.io/clusters/54.235.3.97

View running instances on AWS:
https://console.aws.amazon.com/ec2/v2/home?region=us-east-1

Have fun,
The Mesosphere Team
http://mesosphere.io/

```
bash-3.2$ ssh -l ubuntu 54.226.218.168
Welcome to Ubuntu 12.10 (GNU/Linux 3.5.0-41-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

  System information as of Sat Nov  9 00:24:13 UTC 2013

  System load:  0.0                 Processes:           74

  Usage of /:   16.1% of 7.87GB   Users logged in:     0

  Memory usage: 2%                  IP address for eth0: 10.34.165.84

  Swap usage:   0%

Last login: Sat Nov  9 00:11:02 2013 from 107-202-144-131.example.com
```

step 1: ssh to master

```
ubuntu:~$ sudo aptitude install git
The following NEW packages will be installed:

  git git-man{a} liberror-perl{a}

0 package
Need to g
Do you wa
Get: 1 ht
 kB]
Get: 2 ht
1 [634 kB
Get: 3 ht
[6,165 kB
Fetched 6
Selecting
(Reading
Unpacking
Selecting
Unpacking
Selecting
Unpacking
Processin
Setting u
Setting u
Se
```

```
ubuntu:~$ sudo aptitude -y install openjdk-7-jdk
The following NEW packages will be installed:

  acl{a} at-spi2-core{a} colord{a} cpp{a} cpp-4.7{a} dbus-x11{a} dconf-gsett
service{a} fontconfig{a}
  gconf-service{a} gconf-service-backend{a} gconf2{a} gconf2-common{a} gvfs{
-daemons{a} gvfs-libs{a}
  hicolor-icon-theme{a} libasound2{a} libasyncns0{a} libatasmart4{a} libatk-
wrapper-java{a}
  libatk-wrapper-java-jni{a} libatk1.0-0{a} libatk1.0-data{a} libatspi2.0-0{
ibbonobo2-0{a}
  libbonobo2-common{a} libcairo-gobject2{a} libcairo2{a} libcanberra0{a} lib
a} libdconf1{a}
  libdrm-nouveau2{a} libexif12{a} libflac8{a} libfontenc1{a} libgconf-2-4{a}
-xpm{a}
  libgdk-pixbuf2.0-0{a} libgdk-pixbuf2.0-common{a} libgif4{a} libgl1-mesa-dr
} libglapi-mesa{a}
  libgnome2-0{a} libgnome2-bin{a} libgnome2-common{a} libgnomevfs2-0{a} libg
gphoto2-2{a}
  libgphoto2-l10n{a} libgphoto2-port0{a} libatk-3-0{a} libatk-3-bin{a} libat
```

step 2: install git, jdk-7

```
ubuntu:~$ wget http://spark-project.org/download/spark-0.8.0-incubating-bin-cdh4.tgz
--2013-11-09 00:29:49--  http://spark-project.org/download/spark-0.8.0-incubating-bin-
Resolving spark-project.org (spark-project.org)... 128.32.37.248
Connecting to spark-project.org (spark-project.org)|128.32.37.248|:80... connected.
HTTP request sent, awaiting response... 307 Temporary Redirect
Location: http://d3kbcqa49mib13.cloudfront.net/spark-0.8.0-incubating-bin-cdh4.tgz [f
--2013-11-09 00:29:50--  http://d3kbcqa49mib13.cloudfront.net/spark-0.8.0-incubating-
Resolving d3kbcqa49mib13.cloudfront.net (d3kbcqa49mib13.cloudfront.net)... 54.240.160
7.89, 54.230.17.120, ...
Connecting to d3kbcqa49mib13.cloudfront.net (d3kbcqa49mib13.cloudfront.net)|54.240.16
onnected.
HTTP request sent, awaiting response... 200 OK
Length: 140612059 (134M) [application/x-compressed]
Saving to: `spark-0.8.0-incubating-bin-cdh4.tgz'

100%[======================================================================
12,059 15.1M/s   in 8.4s


2013-11-09 00:29:58 (15.9 MB/s) - `spark-0.8.0-incubating-bin-cdh4.tgz' saved [14061
]

ubuntu:~$ tar xzf spark-0.8.0-incubating-bin-cdh4.tgz
ubuntu:~$ cd spark-0.8.0-incubating-bin-cdh4/
```

step 3: download spark

```
ubuntu:~/spark-0.8.0-incubating-bin-cdh4$ SPARK_HADOOP_VERSION=2.0.0-mr1-cdh4.4.0 sbt/
sembly
Getting net.java.dev.jna jna 3.2.3 ...
downloading http://repo1.maven.org/maven2/net/java/dev/jna/jna/3.2.3/jna-3.2.3.jar ..

        [SUCCESSFUL ] net.java.dev.jna#jna;3.2.3!jna.jar (113ms)

:: retrieving :: org.scala-sbt#boot-jna

        confs: [default]

        1 artifacts copied, 0 already retrieved (838kB/39ms)

Getting org.scala-sbt sbt 0.12.4 ...



[info] Checking every *.class/*.jar file's SHA-1.
[info] SHA-1: 1e99bae998cb96697f95cf8d8b44370b6a7bae71
[info] Packaging /home/ubuntu/spark-0.8.0-incubating-bin-cdh4/examples/target/scala-2
amples-assembly-0.8.0-incubating.jar ...
[info] Done packaging.
[success] Total time: 806 s, completed Nov 9, 2013 12:46:08 AM
```

step 4: sbt clean assembly

```
ubuntu:~/spark-0.8.0-incubating-bin-cdh4$ ./make-distribution.sh --hadoop 2.0.0-mr1-c
...
[success] Total time: 69 s, completed Nov 9, 2013 12:55:03 AM


ubuntu:~/spark-0.8.0-incubating-bin-cdh4$ mv dist spark-0.8.0-2.0.0-mr1-cdh4.4.0
ubuntu:~/spark-0.8.0-incubating-bin-cdh4$ tar czf spark-0.8.0-2.0.0-mr1-cdh4.4.0.tgz
.0.0-mr1-cdh4.4.0
```

```
ubuntu:~/spark-0.8.0-incubating-bin-cdh4$ hadoop fs -mkdir /tmp
ubuntu:~/spark-0.8.0-incubating-bin-cdh4$ hadoop fs -put spark-0.8.0-2.0.0-mr1-cdh4
```

```
ubuntu:~/spark-0.8.0-incubating-bin-cdh4$ cd conf/
ubuntu:~/spark-0.8.0-incubating-bin-cdh4/conf$ cp spark-env.sh.template spark-env.sh
ubuntu:~/spark-0.8.0-incubating-bin-cdh4/conf$ vim spark-env.sh
ubuntu:~/spark-0.8.0-incubating-bin-cdh4/conf$ cat spark-env.sh
#!/usr/bin/env bash
export MESOS_NATIVE_LIBRARY=/usr/local/lib/libmesos.so
export SPARK_EXECUTOR_URI=hdfs://54.204.196.36/tmp/spark-0.8.0-2.0.0-mr1-cdh4.4.0.tgz
export MASTER=zk://54.226.218.168:2181/mesos
ubuntu:~/spark-0.8.0-incubating-bin-cdh4/conf$ cd ..
```

step 5: make distro, cp to hdfs, set env

```
ubuntu:~/spark-0.8.0-incubating-bin-cdh4$ ./spark-shell
Welcome to

      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /___/ .__/\_,_/_/ /_/\_\   version 0.8.0
      /_/

Using Scala version 2.9.3 (OpenJDK 64-Bit Server VM, Java 1.7.0_25)
Initializing interpreter...
13/11/09 17:21:23 INFO server.Server: jetty-7.x.y-SNAPSHOT
13/11/09 17:21:23 INFO server.AbstractConnector: Started SocketConnector@0.0.0.0:4667
Creating SparkContext...
13/11/09 17:21:33 INFO slf4j.Slf4jEventHandler: Slf4jEventHandler started
13/11/09 17:21:33 INFO spark.SparkEnv: Registering BlockManagerMaster
13/11/09 17:21:33 INFO storage.MemoryStore: MemoryStore started with capacity 323.9 M
13/11/09 17:21:33 INFO storage.DiskStore: Created local directory at /tmp/spark-local-
33-fea7
13/11/09 17:21:33 INFO network.ConnectionManager: Bound socket to port 53657 with id =
anagerId(ec2-54-234-234-236.compute-1.amazonaws.com,53657)
13/11/09 17:21:33 INFO storage.BlockManagerMaster: Trying to register BlockManager
13/11/09 17:21:33 INFO storage.BlockManagerMaster: Registered BlockManager
13/11/09 17:21:33 INFO server.Server: jetty-7.x.y-SNAPSHOT
13/11/09 17:21:33 INFO server.AbstractConnector: Started SocketConnector@0.0.0.0:5699
13/
99
13/11/09 17:21:33 INFO spark.SparkEnv: Registering MapOutputTracker
```

et voilà!

Spark Summit, 2013-12-02

- Learn how to use Mesos
- Spark on Mesos
- **PMML for Spark (WIP)**

powered by mesosphere

PMML – standard

- established XML standard for predictive model markup
- organized by Data Mining Group (DMG), since 1997
  http://dmg.org/
- members: *IBM*, *SAS*, *Visa*, *Equifax*, *Microstrategy*, *Microsoft*, etc.
- PMML concepts for *metadata*, *ensembles*, etc., translate directly into tuple-based workflows

"PMML is the leading standard for statistical and data mining models and supported by over 20 vendors and organizations. With PMML, it is easy to develop a model on one system using one application and deploy the model on another system using another application."

wikipedia.org/wiki/Predictive_Model_Markup_Language

PMML – create a model in R

```
## train a RandomForest model

f <- as.formula("as.factor(label) ~ .")
fit <- randomForest(f, data_train, ntree=50)

## test the model on the holdout test set

print(fit$importance)
print(fit)

predicted <- predict(fit, data)
data$predicted <- predicted
confuse <- table(pred = predicted, true = data[,1])
print(confuse)

## export predicted labels to TSV

write.table(data, file=paste(dat_folder, "sample.tsv", sep="/"),
  quote=FALSE, sep="\t", row.names=FALSE)

## export RF model to PMML

saveXML(pmml(fit), file=paste(dat_folder, "sample.rf.xml", sep="/"))
```

PMML – analytics workflow metadata


```
<?xml version="1.0"?>
<PMML version="4.0" xmlns="http://www.dmg.org/PMML-4_0"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xsi:schemaLocation="http://www.dmg.org/PMML-4_0
 http://www.dmg.org/v4-0/pmml-4-0.xsd">
 <Header copyright="Copyright (c)2012 Concurrent, Inc." description="Random Forest Tree Model">
  <Extension name="user" value="ceteri" extender="Rattle/PMML"/>
  <Application name="Rattle/PMML" version="1.2.30"/>
  <Timestamp>2012-10-22 19:39:28</Timestamp>
 </Header>
 <DataDictionary numberOfFields="4">
  <DataField name="label" optype="categorical" dataType="string">
   <Value value="0"/>
   <Value value="1"/>
  </DataField>
  <DataField name="var0" optype="continuous" dataType="double"/>
  <DataField name="var1" optype="continuous" dataType="double"/>
  <DataField name="var2" optype="continuous" dataType="double"/>
 </DataDictionary>
 <MiningModel modelName="randomForest_Model" functionName="classification">
  <MiningSchema>
   <MiningField name="label" usageType="predicted"/>
   <MiningField name="var0" usageType="active"/>
   <MiningField name="var1" usageType="active"/>
   <MiningField name="var2" usageType="active"/>
  </MiningSchema>
  <Segmentation multipleModelMethod="majorityVote">
   <Segment id="1">
    <True/>
    <TreeModel modelName="randomForest_Model" functionName="classification" algorithmName="randomForest"
splitCharacteristic="binarySplit">
```

PMML – vendor coverage

PMML – scope

- Association Rules: *AssociationModel* element
- Cluster Models: *ClusteringModel* element
- Decision Trees: *TreeModel* element
- Naïve Bayes Classifiers: *NaiveBayesModel* element
- Neural Networks: *NeuralNetwork* element
- Regression: *RegressionModel* and *GeneralRegressionModel* elements
- Rulesets: *RuleSetModel* element
- Sequences: *SequenceModel* element
- Support Vector Machines: *SupportVectorMachineModel* element
- Text Models: *TextModel* element
- Time Series: *TimeSeriesModel* element

…XML extensions

…plus ensembles, chaining, etc.

Python stack:

https://code.google.com/p/augustus/

Cascading/Hadoop/JVM:

https://github.com/Cascading/pattern

Why integration into Spark, not MLbase?

- focus on workflow integration,
  not model selection
- offload legacy systems
- better integration/migration with other
  distributed frameworks
- audited/regulated industries pose restrictions
- current needs in Enterprise use cases;
  is MLbase ready for that today?

Join us!

Data Day Texas, Austin
Jan 11
2014.datadaytexas.com
(incl. Mesos talk+BOF)

O'Reilly Strata, Santa Clara
Feb 11-13
strataconf.com/strata2014
(incl. Mesos talk+BOF+workshop)

powered by mesosphere