amazon web services

Masterclass

Elastic MapReduce

Ian Massingham – Technical Evangelist
@IanMmmm

Masterclass

A technical deep dive beyond the basics

Help educate you on how to get the best from AWS technologies

Show you how things work and how to get things done

Broaden your knowledge in ~45 mins

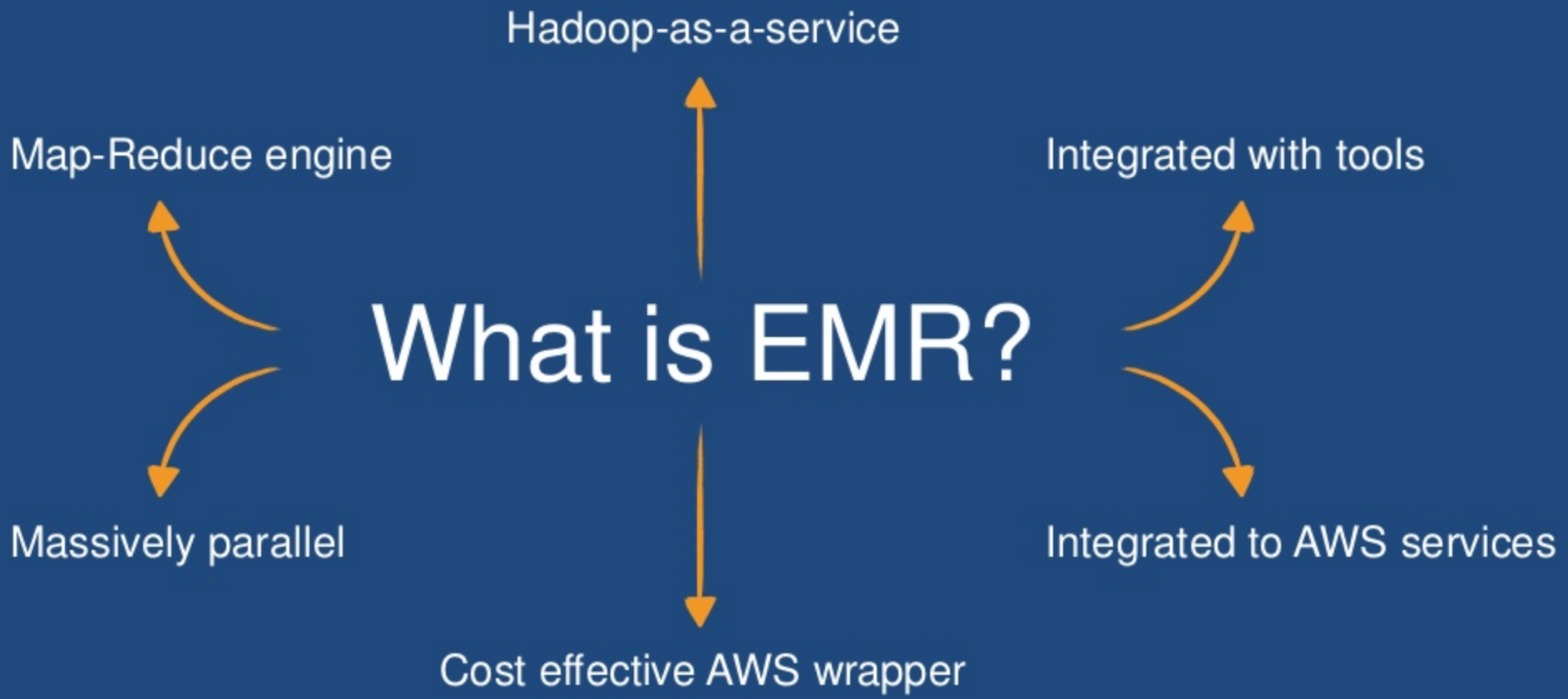
Amazon Elastic MapReduce

A key tool in the toolbox to help with ‘Big Data’ challenges

Makes possible analytics processes previously not feasible

Cost effective when leveraged with EC2 spot market

Broad ecosystem of tools to handle specific use cases


Hadoop-as-a-service
Map-Reduce engine
Integrated with tools
What is EMR?
Massively parallel
Integrated to AWS services
Cost effective AWS wrapper

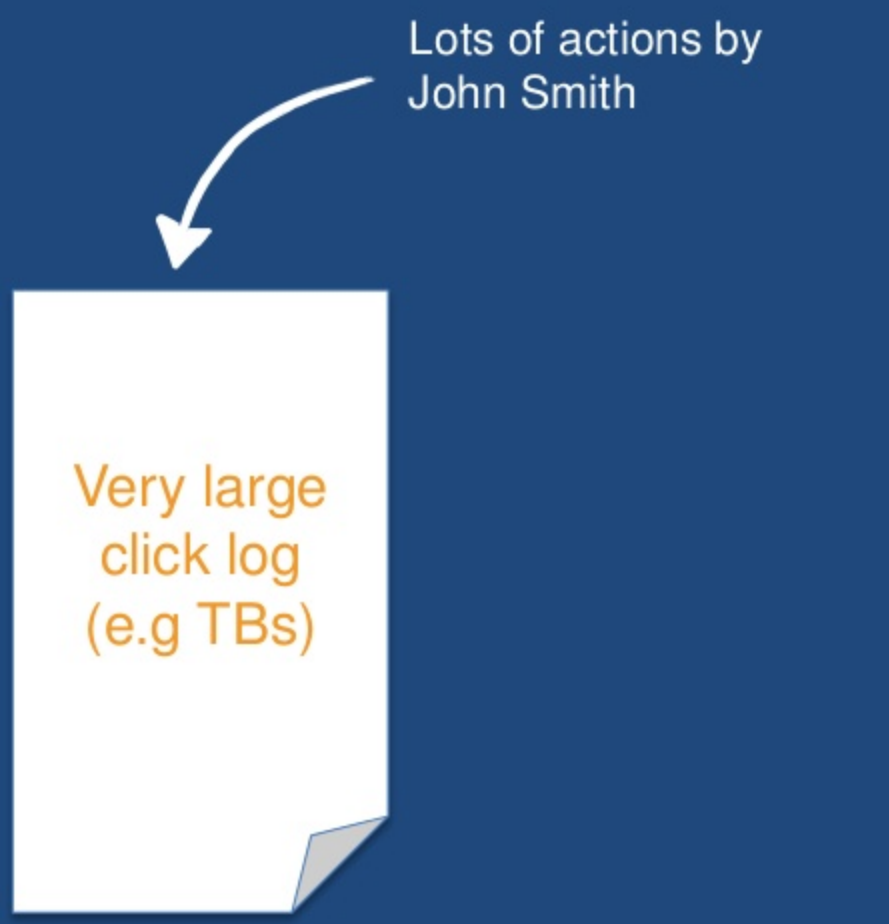
A framework ◀

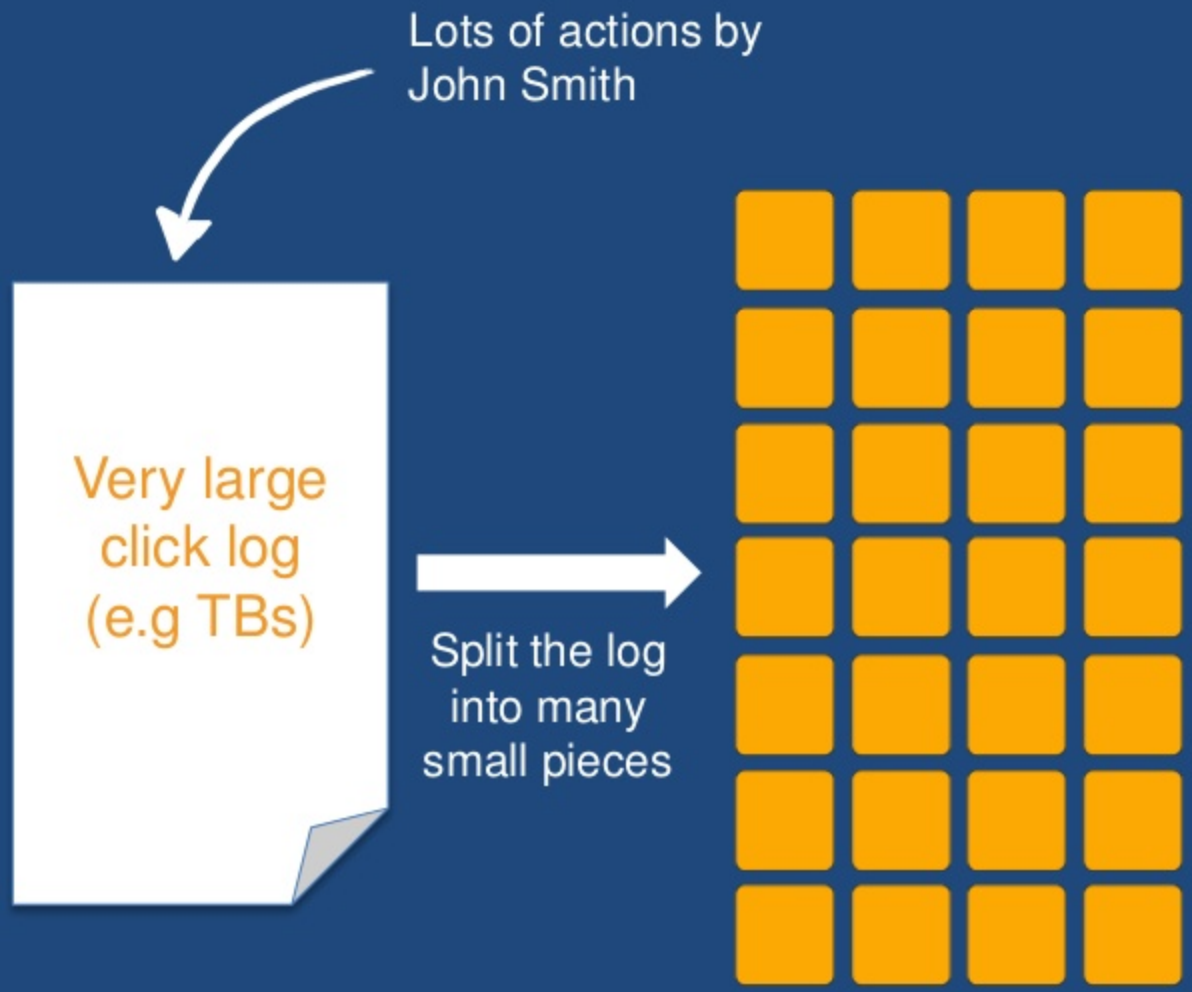
Splits data into pieces ◀

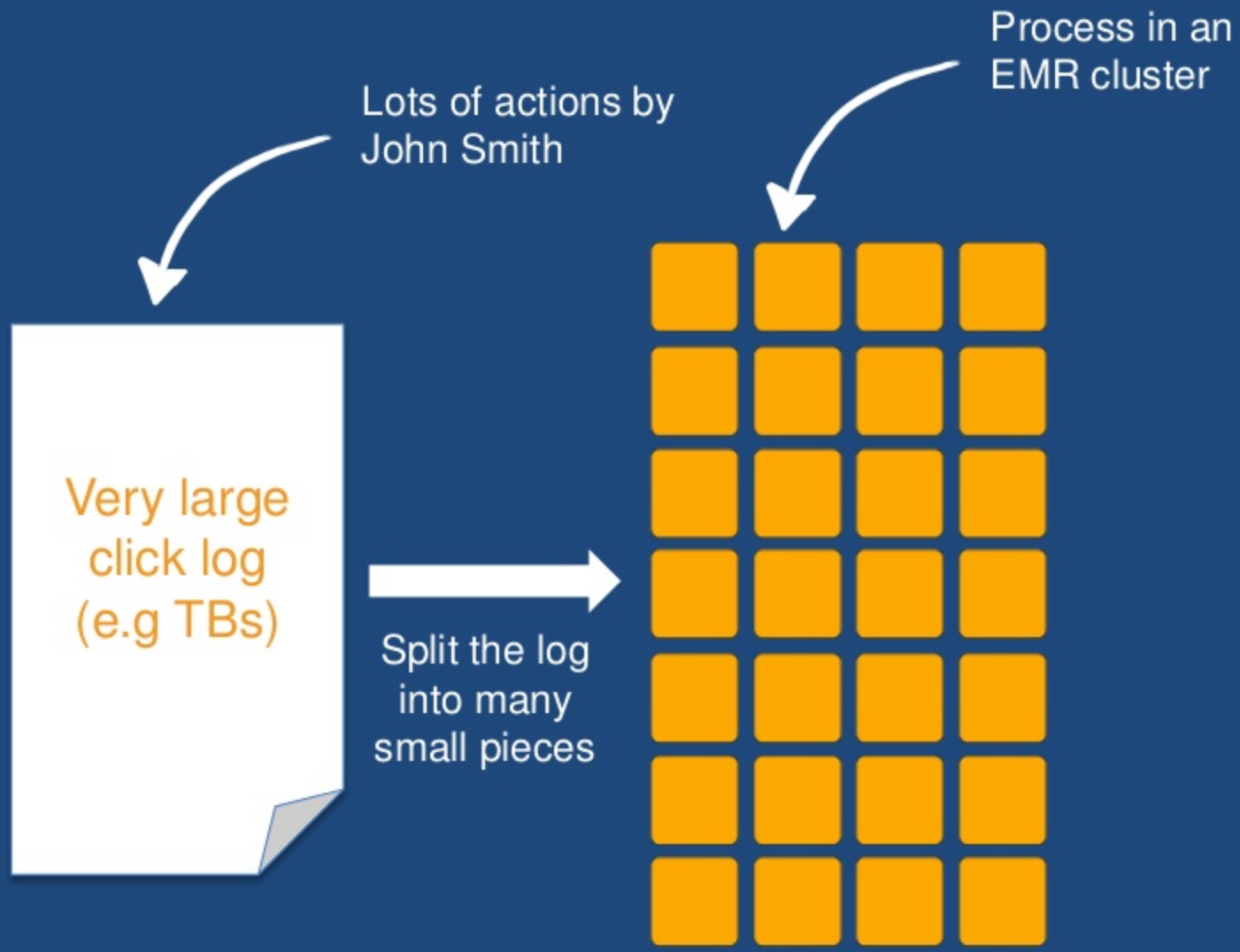
Lets processing occur ◀

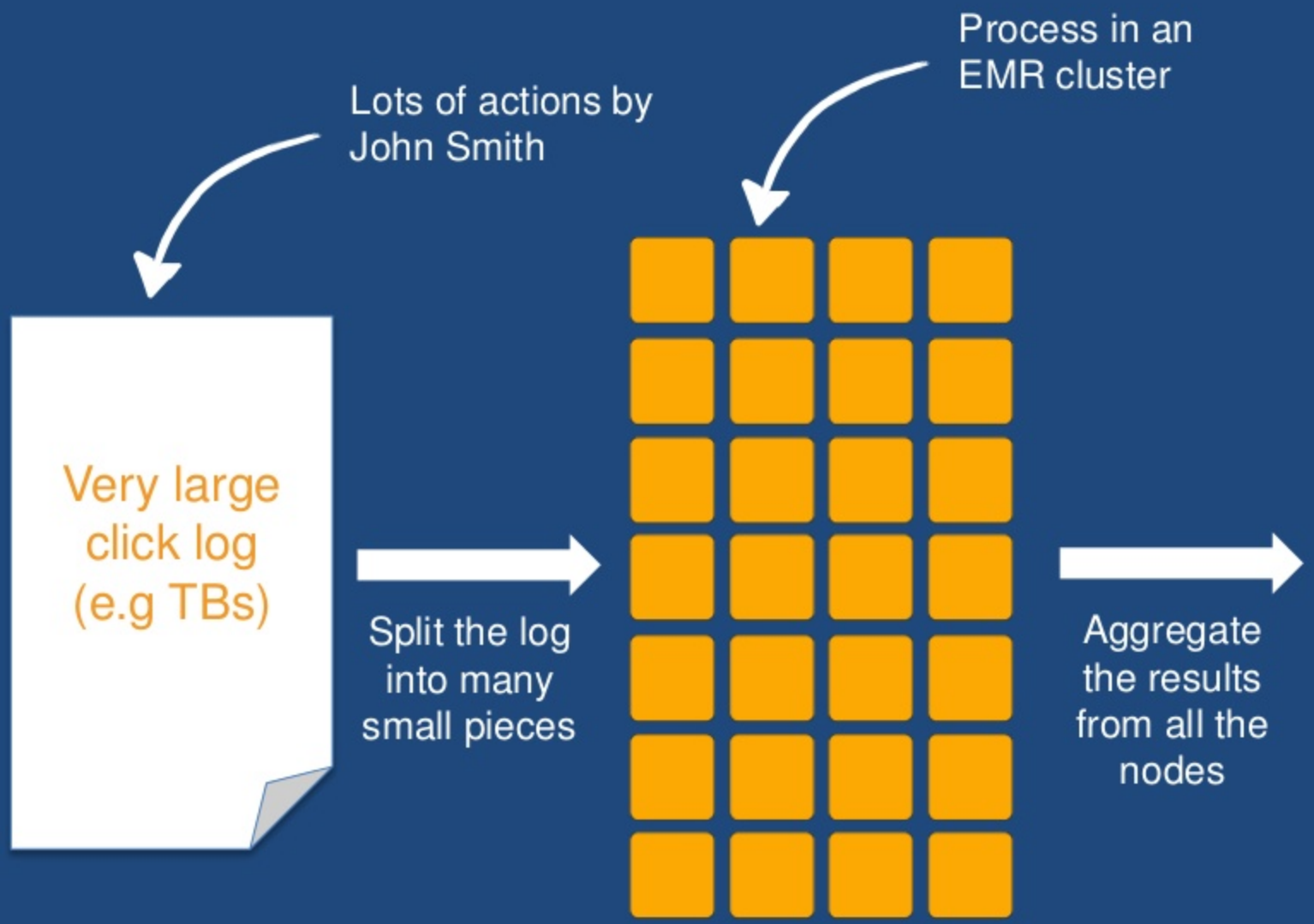
Gathers the results ◀

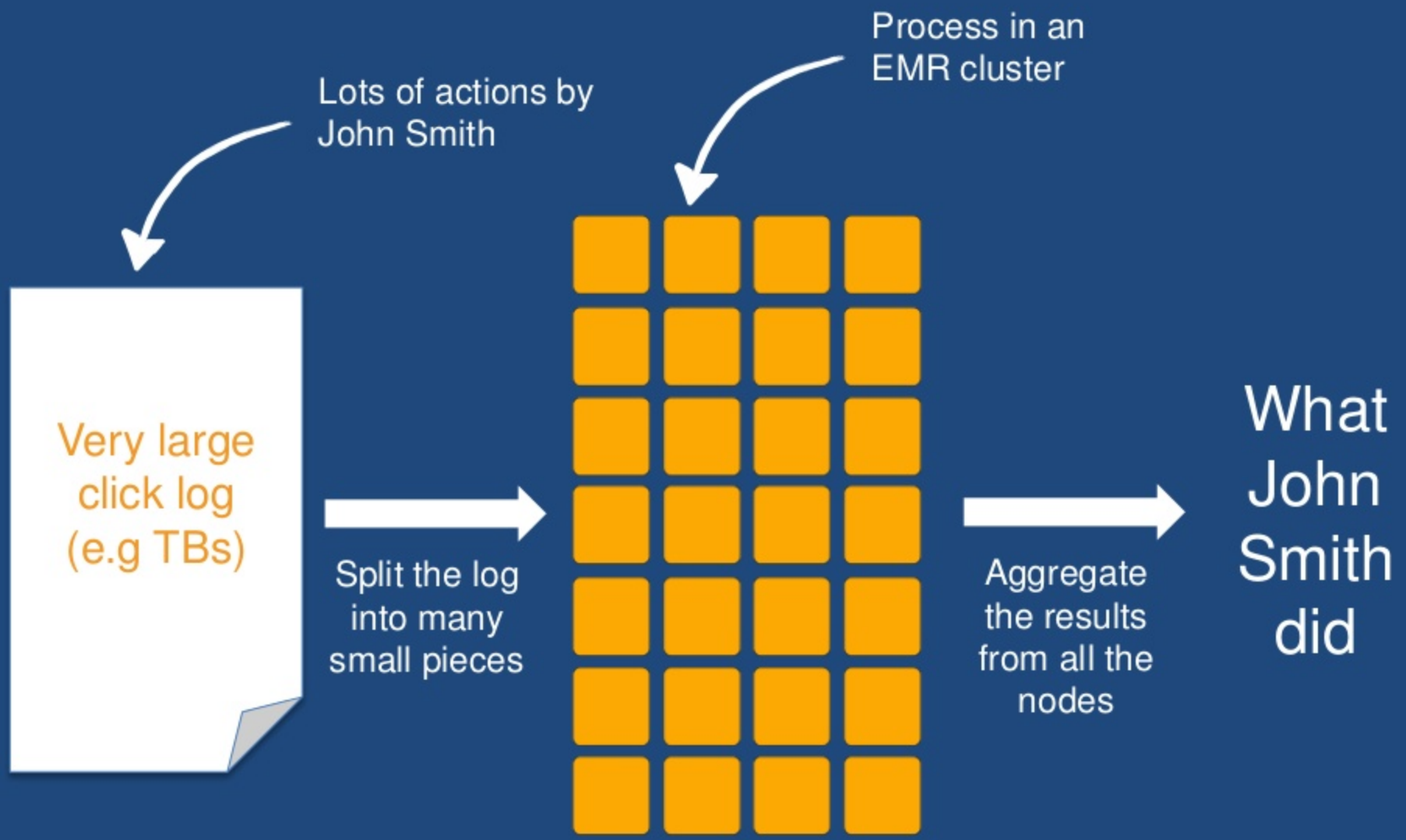

Very large
click log
(e.g TBs)

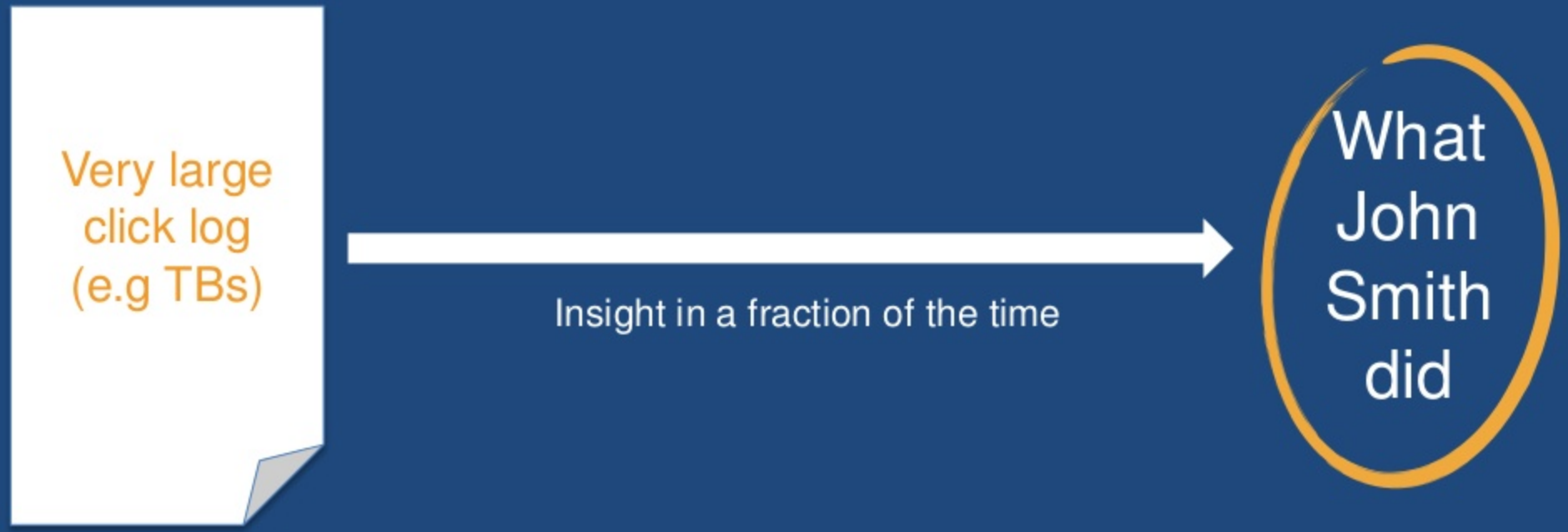

Lots of actions by
John Smith
Very large
click log
(e.g TBs)

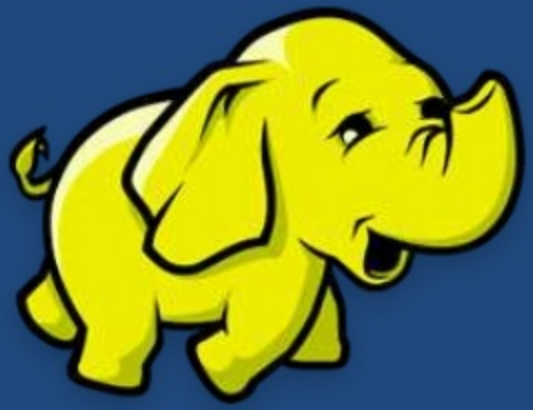

Lots of actions by
John Smith
Very large
click log
(e.g TBs)
Split the log
into many
small pieces

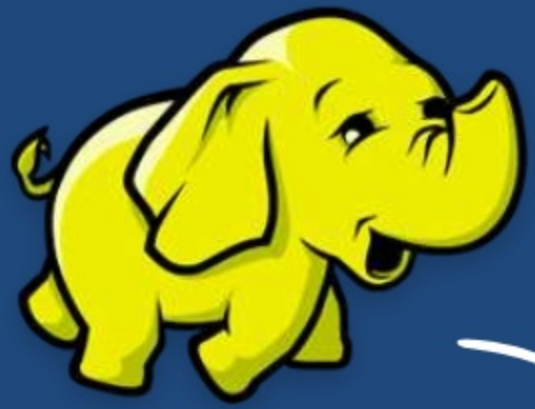

Process in an
EMR cluster
Lots of actions by
John Smith
Very large
click log
(e.g TBs)
Split the log
into many
small pieces


Process in an
EMR cluster
Lots of actions by
John Smith
Very large
click log
(e.g TBs)
Split the log
into many
small pieces
Aggregate
the results
from all the
nodes


Process in an EMR cluster
Lots of actions by John Smith
Very large click log (e.g TBs)
Split the log into many small pieces
Aggregate the results from all the nodes
What John Smith did


Very large
click log
(e.g TBs)
Insight in a fraction of the time
What
John
Smith
did





HDFS

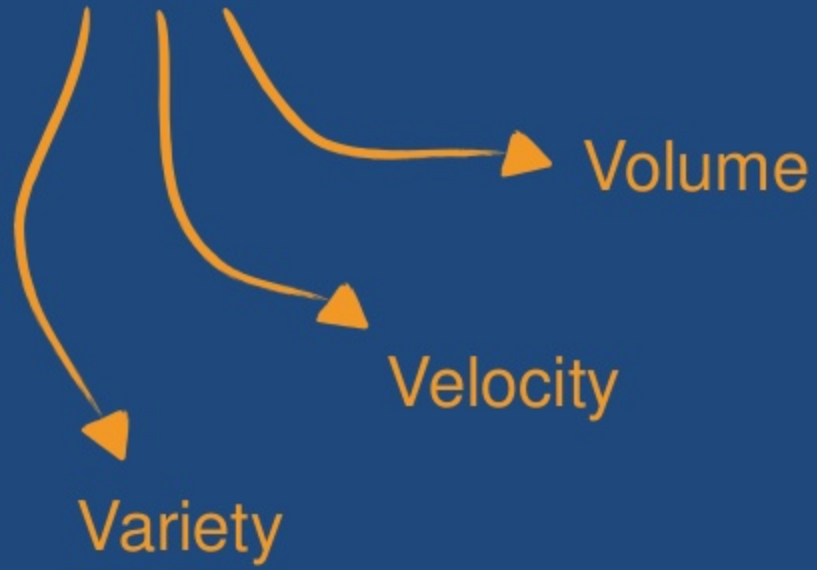
Reliable storage

MapReduce

Data analysis

Big Data != Hadoop

Big Data != Hadoop

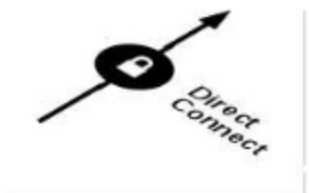


Big Data != Hadoop

When you need to innovate to collect, store, analyze, and manage your data

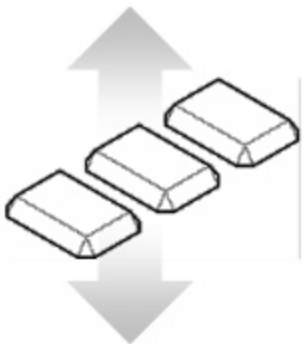
Storage — Big Data — Compute

How do you get your data into AWS?



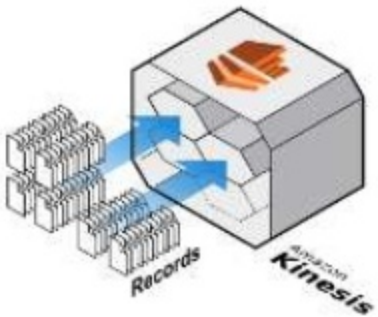
AWS Direct Connect

Dedicated low latency bandwidth



AWS Import/Export

Physical media shipping



Amazon Kinesis

Highly scalable stream processing for event/record buffering

Amazon Storage Gateway

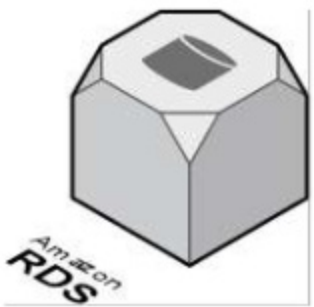
Sync local storage to the cloud

Storage — Big Data — Compute

Where do you put your data once it's in AWS?

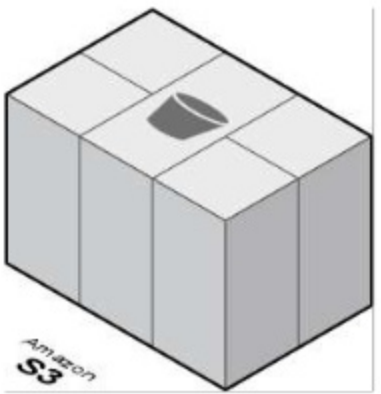
DynamoDB

NoSQL, Schemaless, Provisioned throughput database



Relational Database Service

Fully managed database *(PostgreSQL, MySQL, Oracle, MSSQL)*



S3

Object datastore up to 5TB per object

99.999999999% durability

Elastic MapReduce

Getting started, tools & operating modes

Upload

Upload your data and processing application to S3.

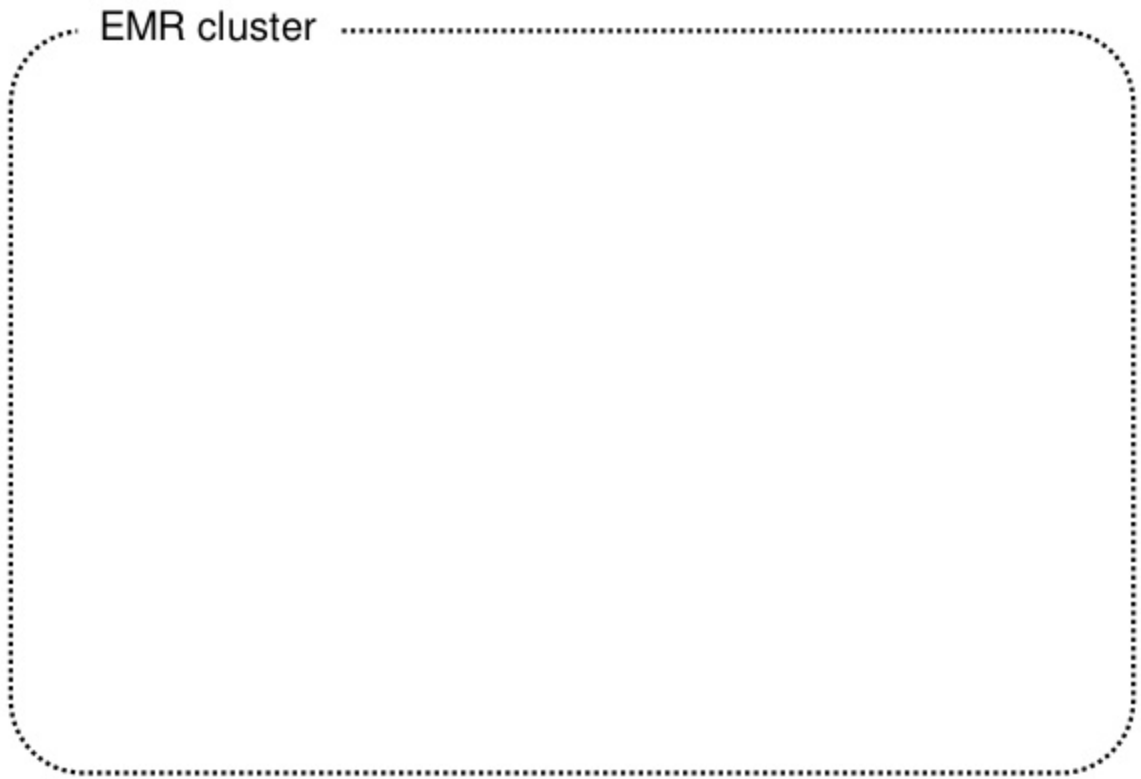
Create

Configure and create your cluster by specifying data inputs, outputs, cluster size, security settings, etc.

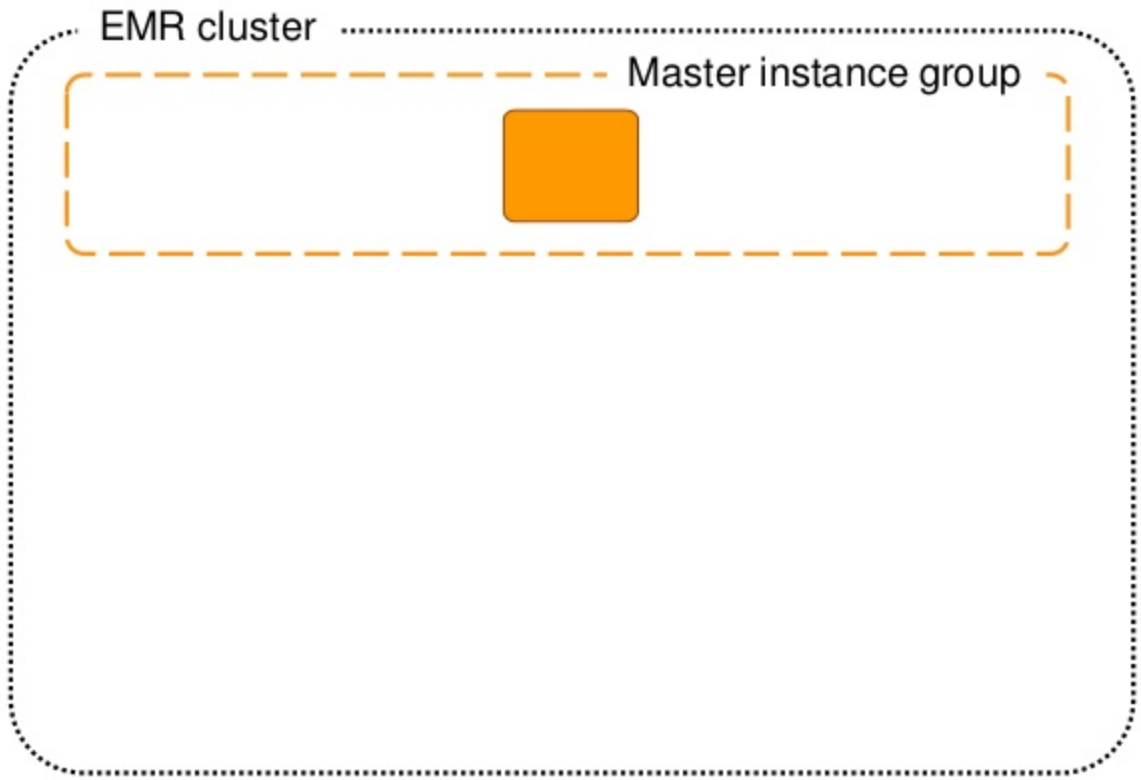
Monitor

Monitor the health and progress of your cluster. Retrieve the output in S3.

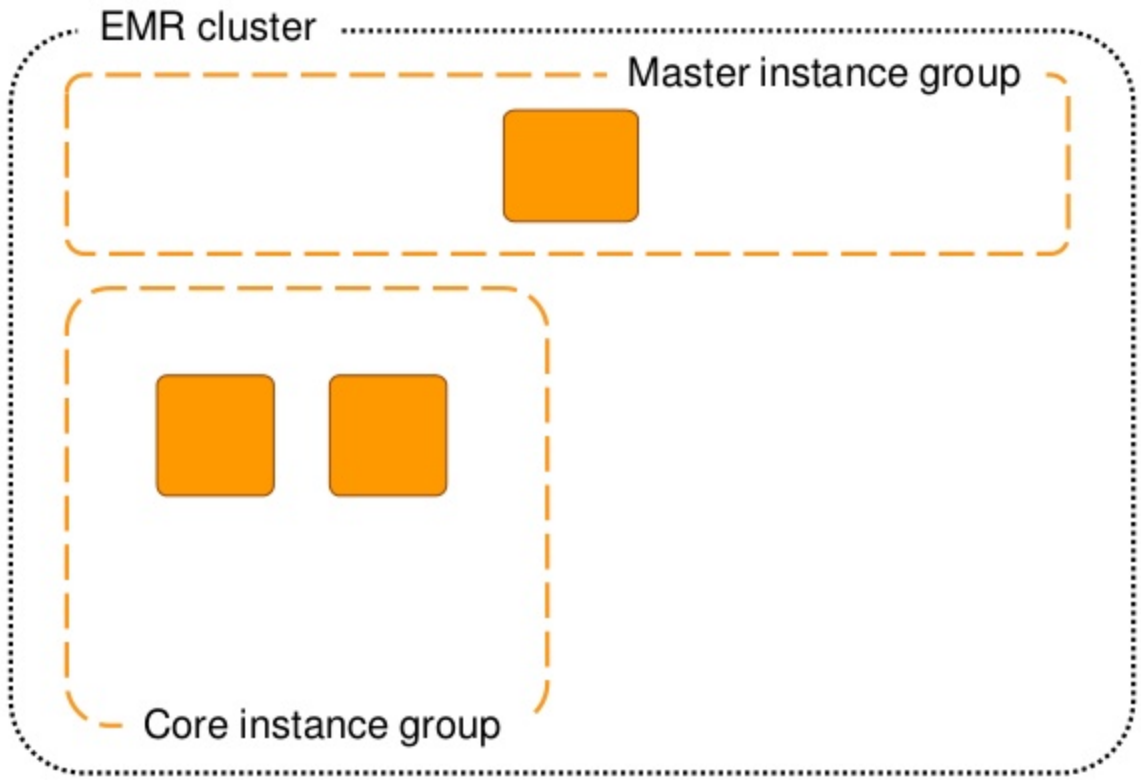
Start an EMR cluster using console or cli tools



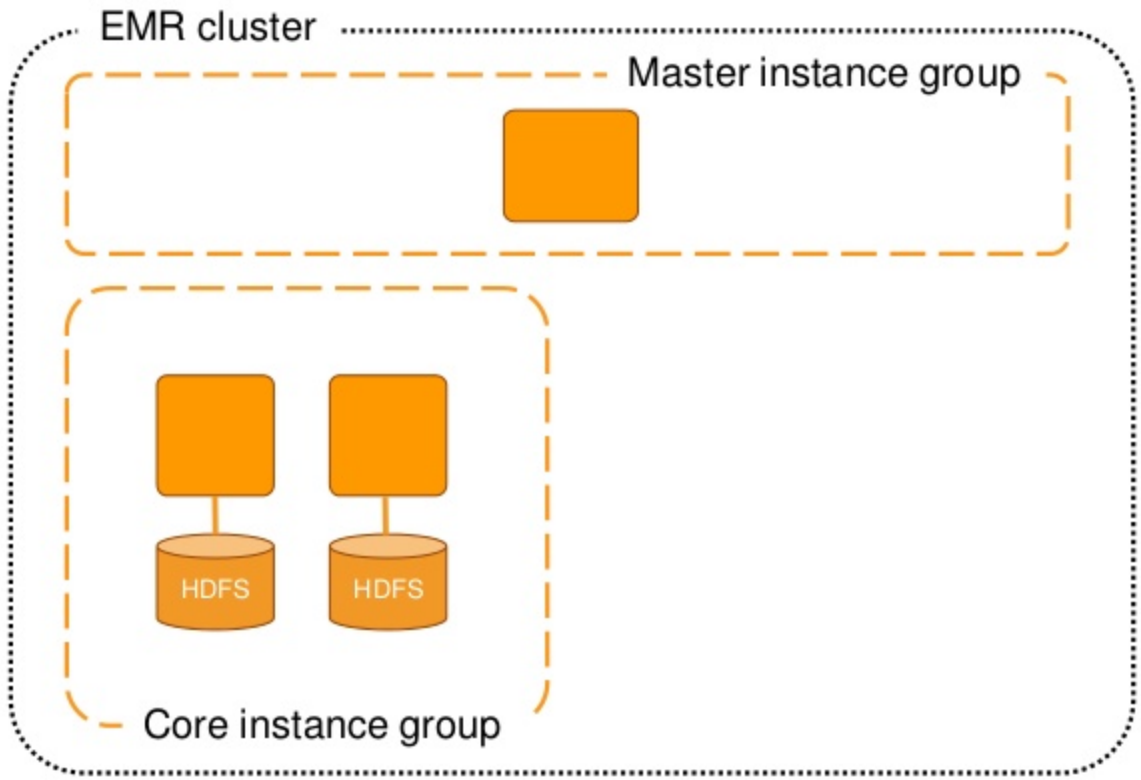
Master instance group created that controls the cluster (runs MySQL)



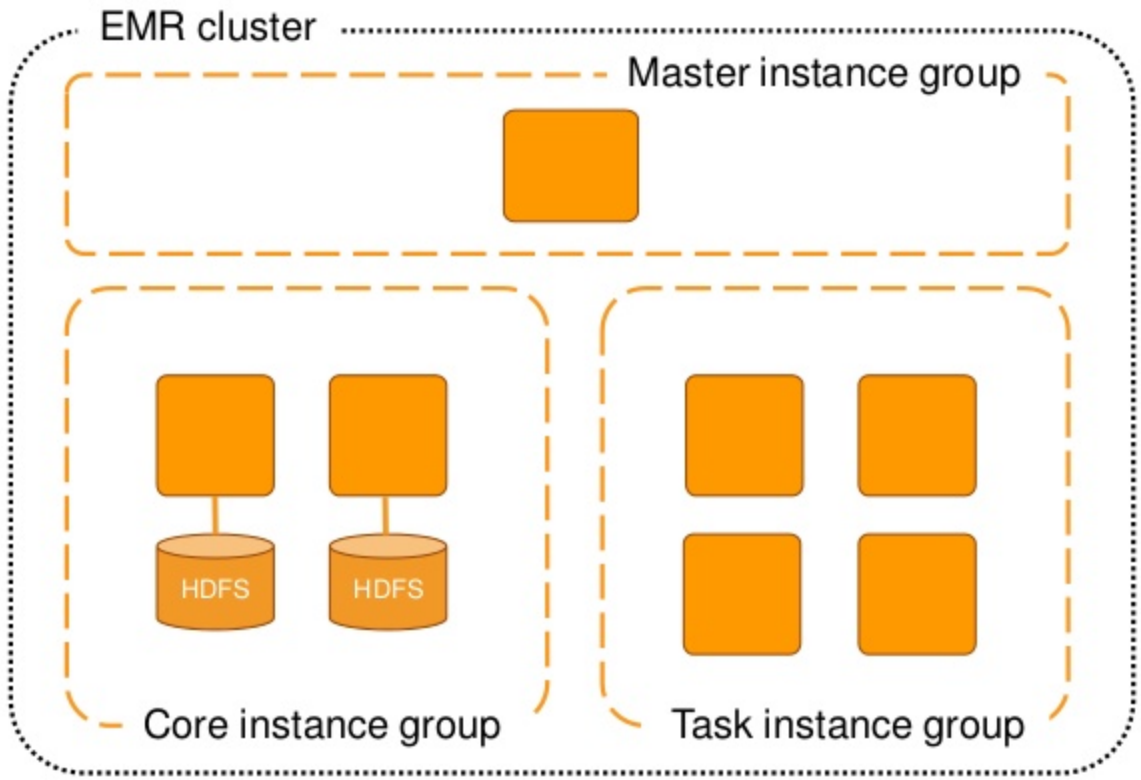
Core instance group created for life of cluster



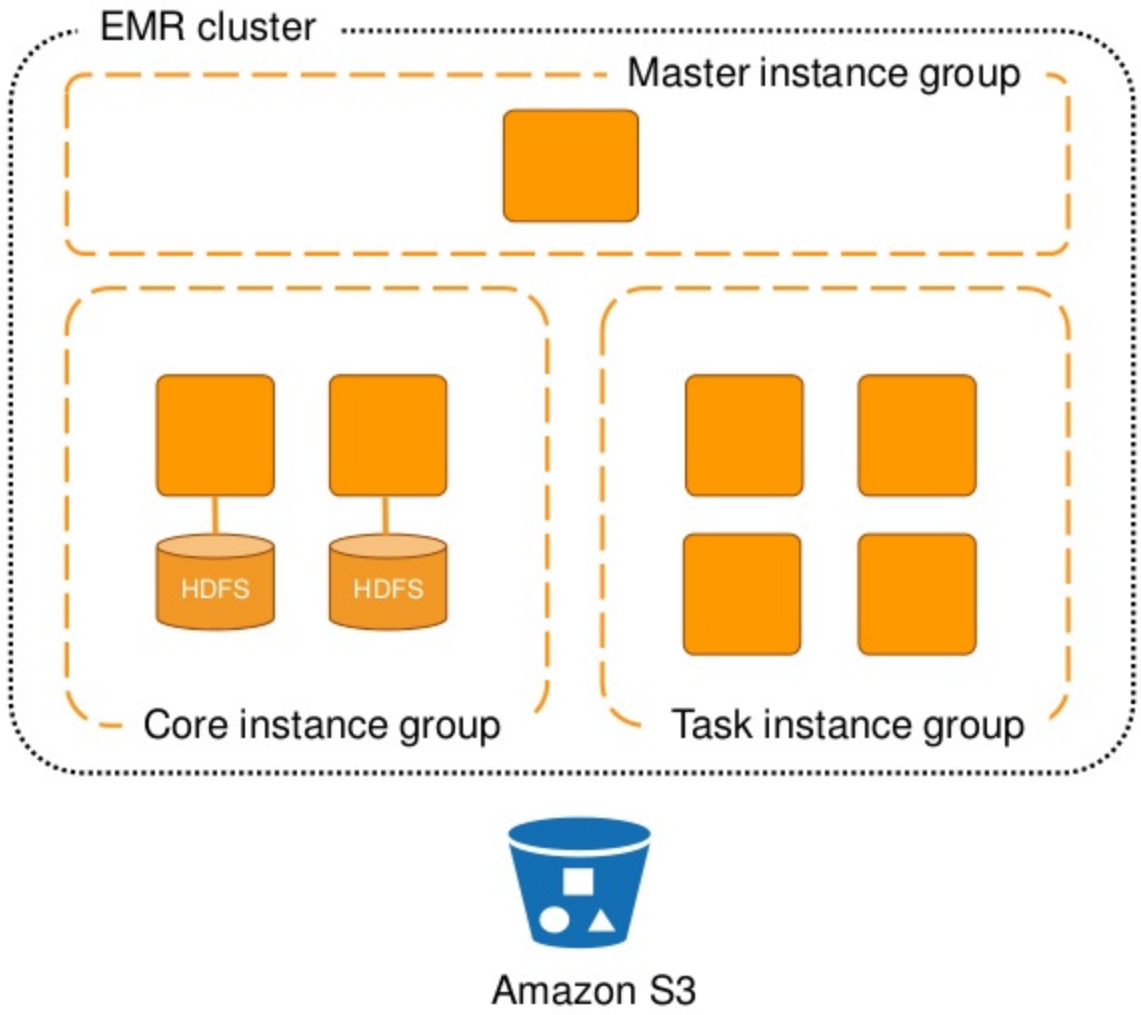
Core instances run DataNode and TaskTracker daemons



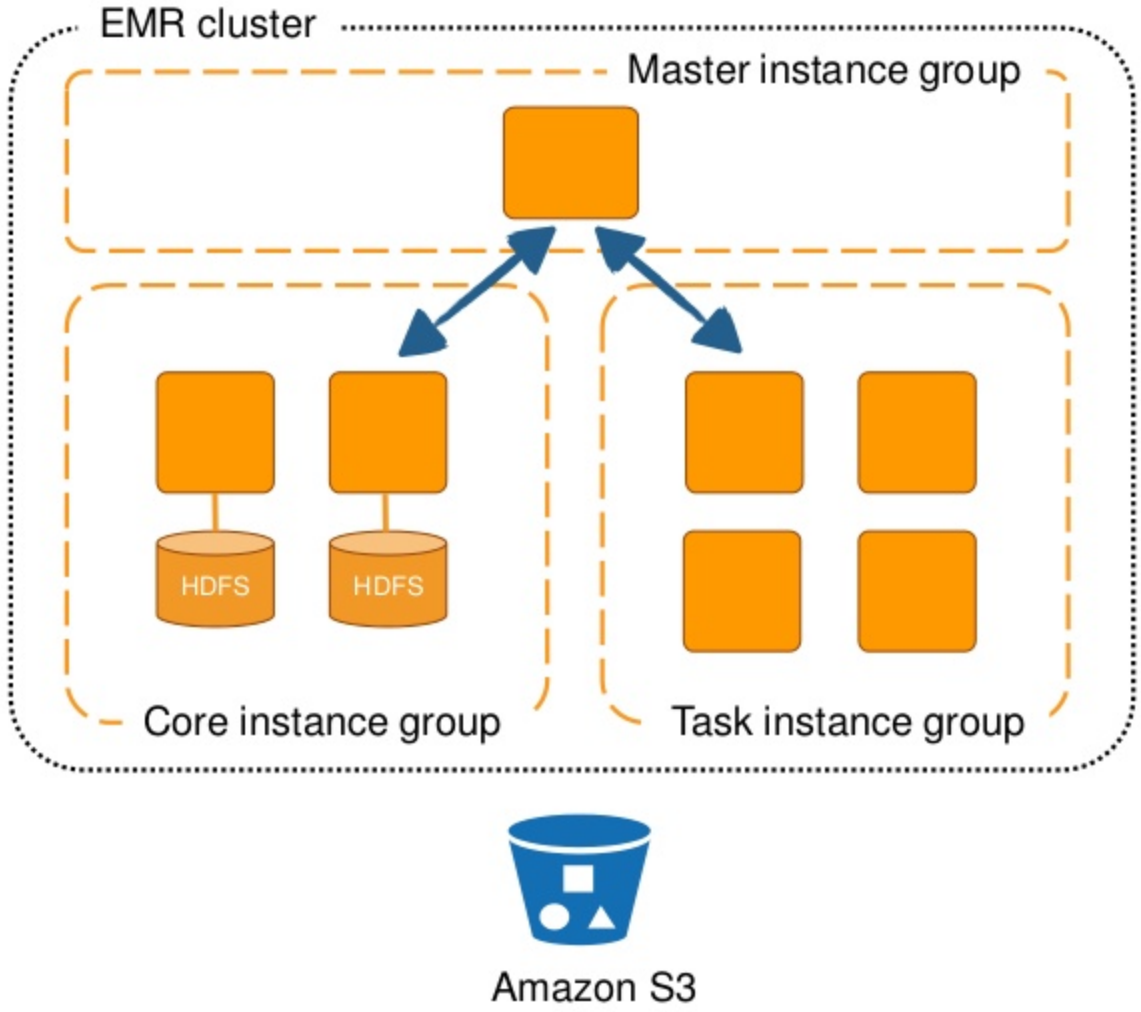
Optional task instances can be added or subtracted to perform work



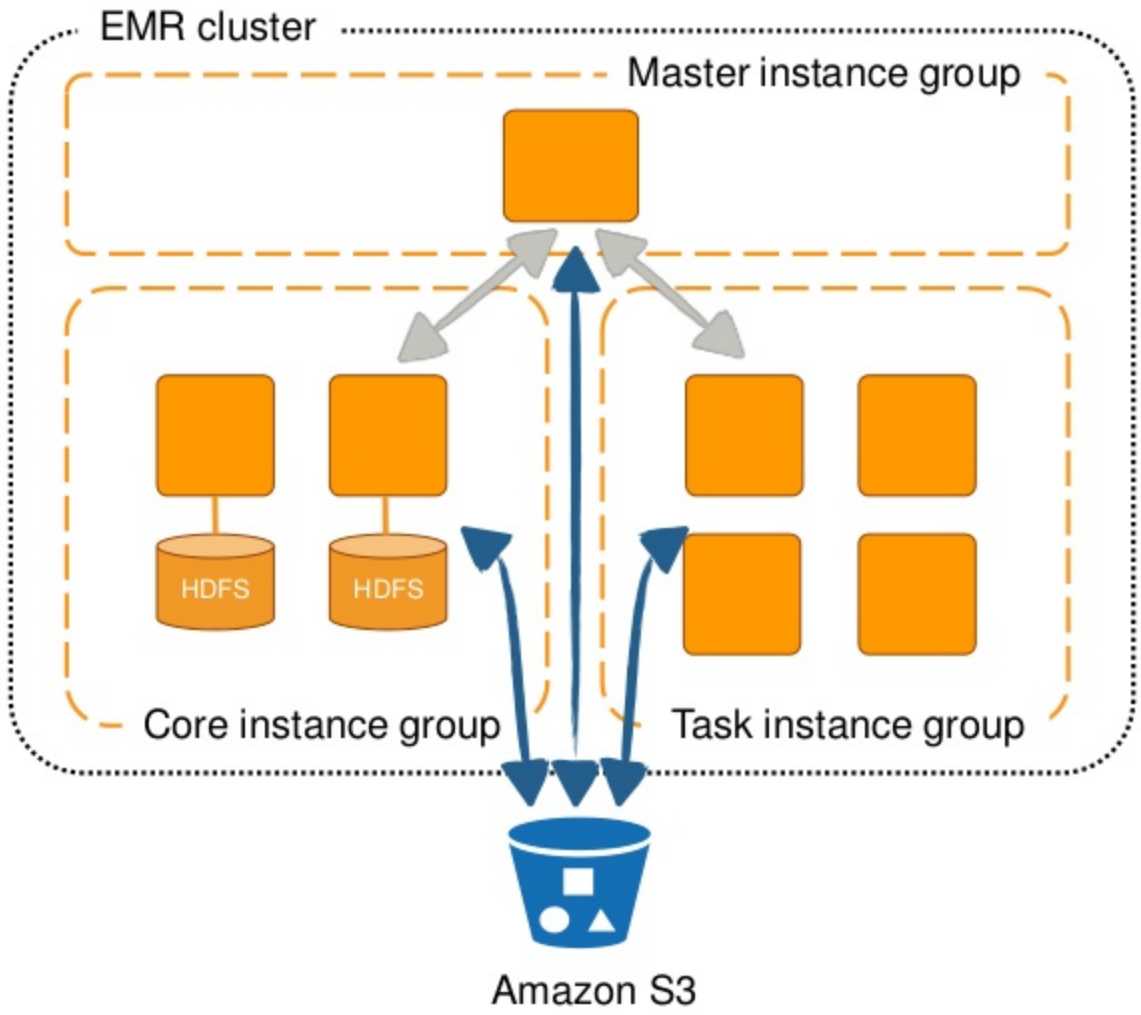
S3 can be used as underlying ‘file system’ for input/output data

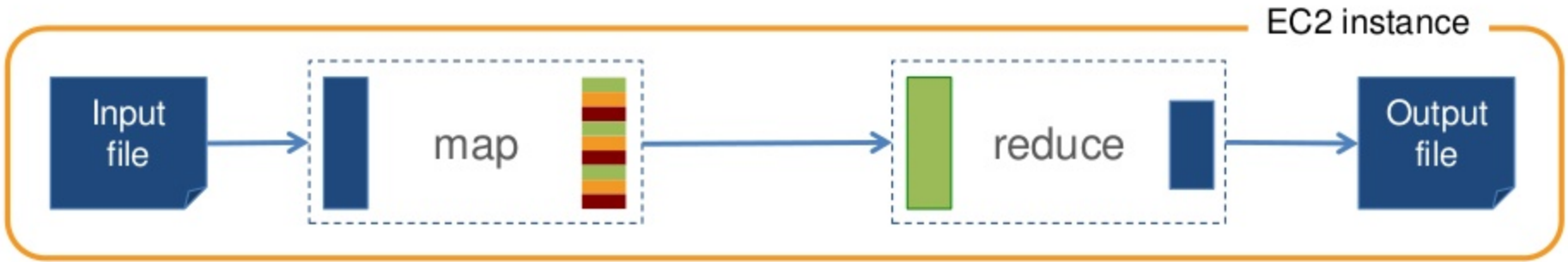


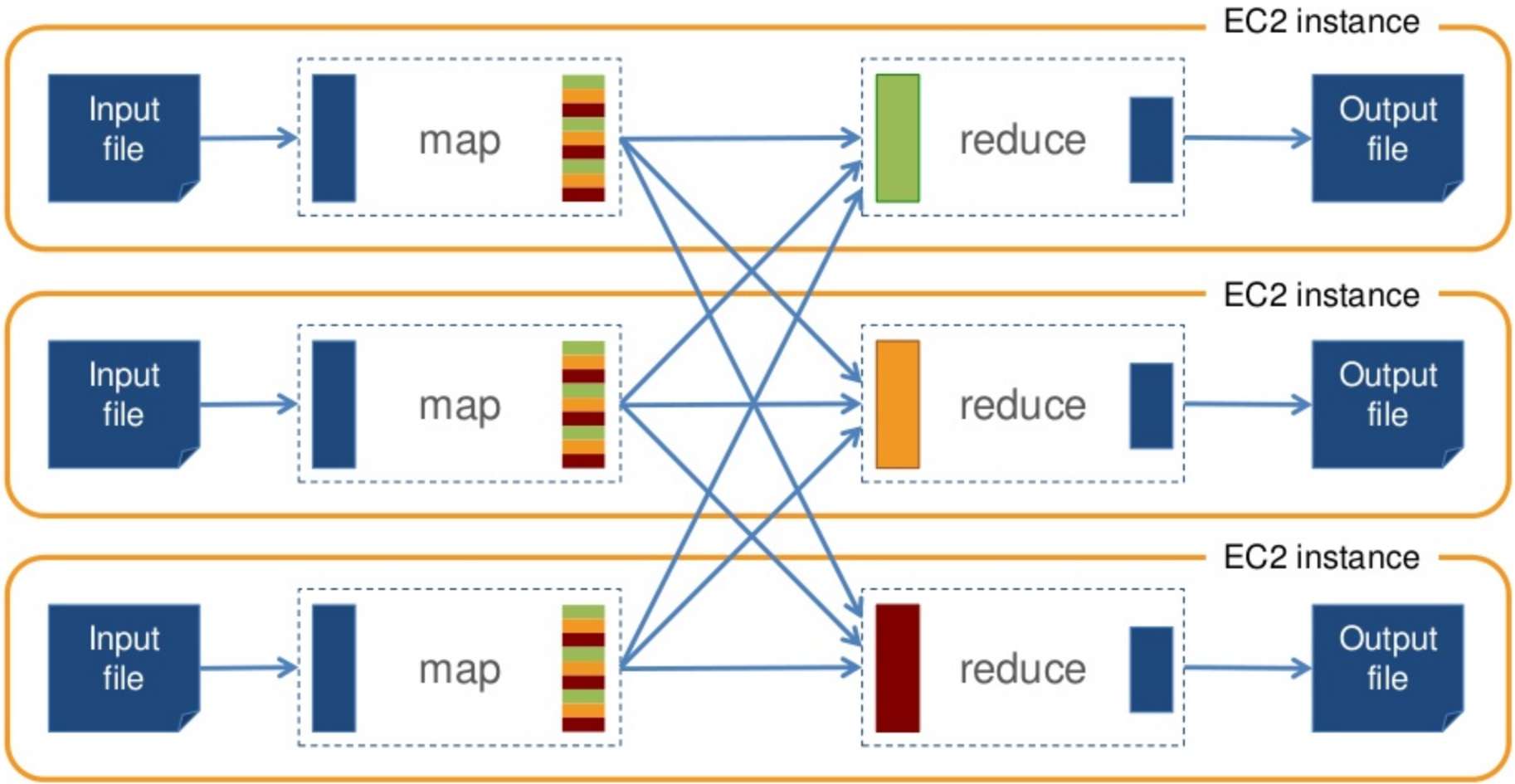
Master node coordinates distribution of work and manages cluster state



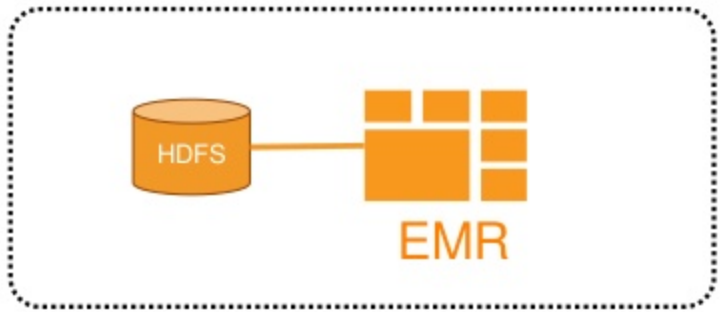
Core and Task instances read-write to S3



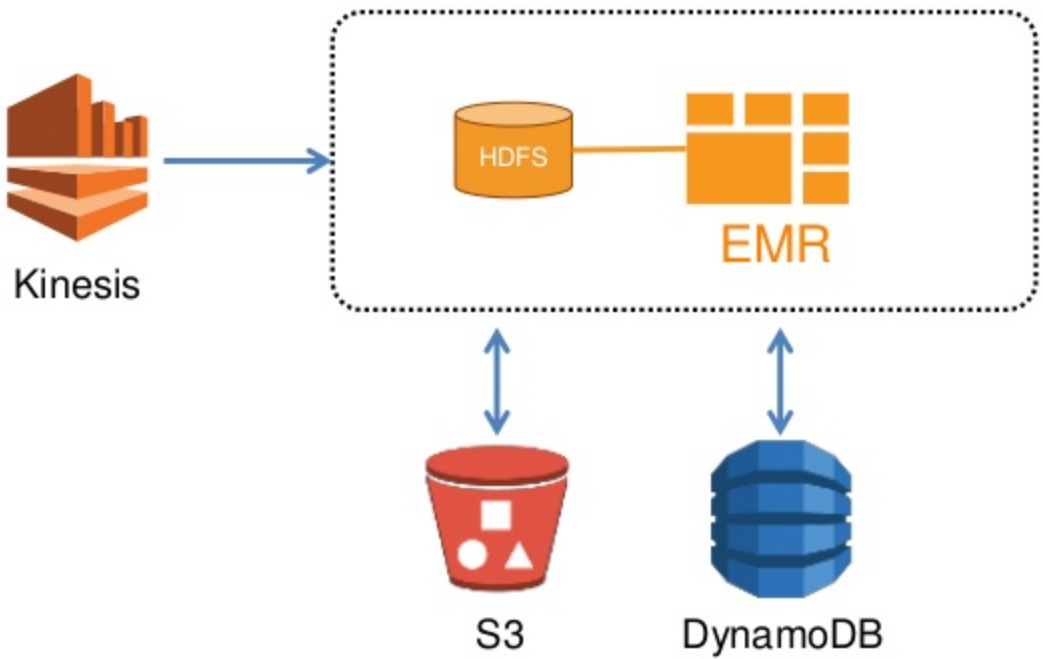

EC2 instance
Input file
map
reduce
Output file

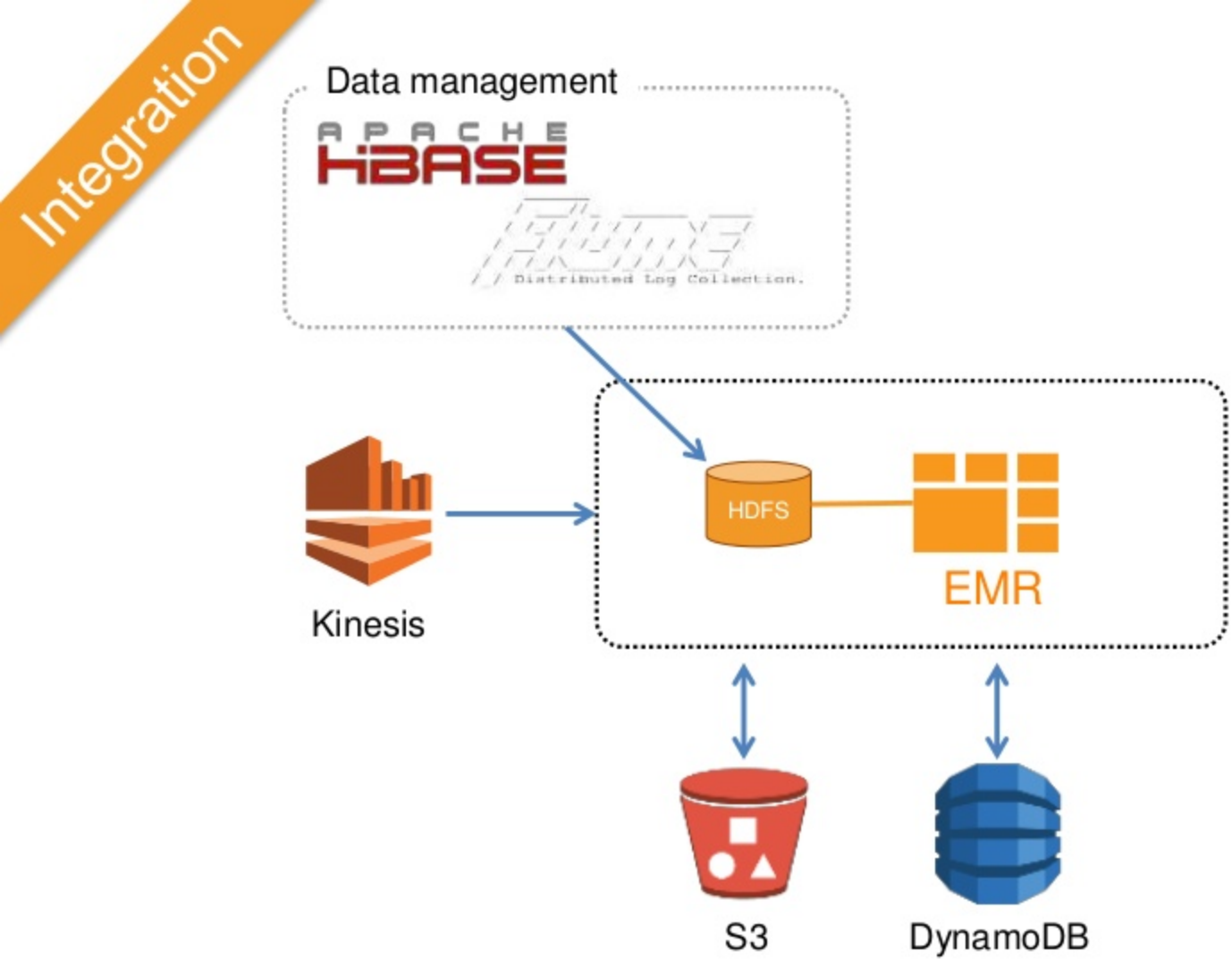

EC2 instance
Input file
map
reduce
Output file
EC2 instance
Input file
map
reduce
Output file
EC2 instance
Input file
map
reduce
Output file

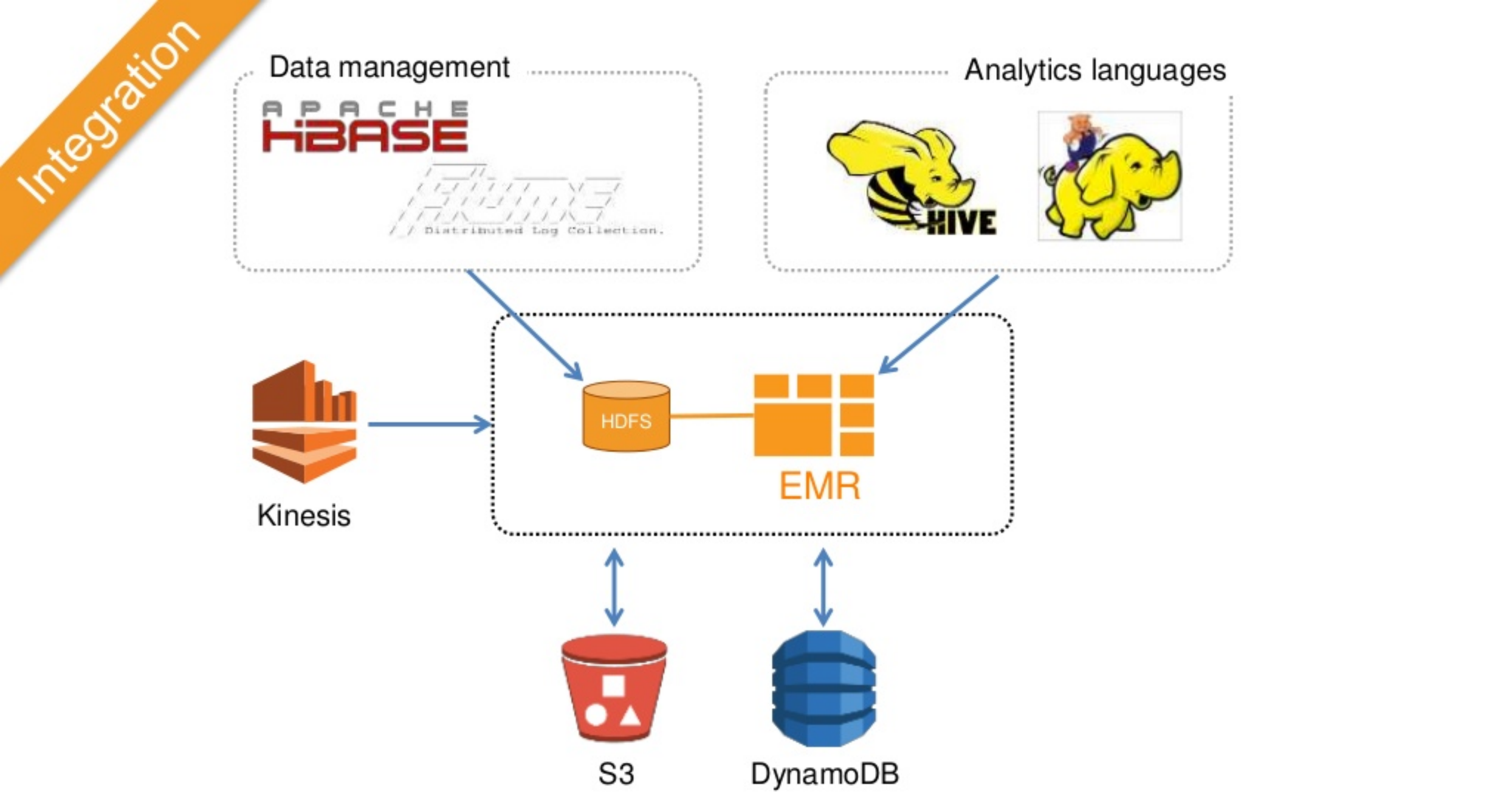
Integration

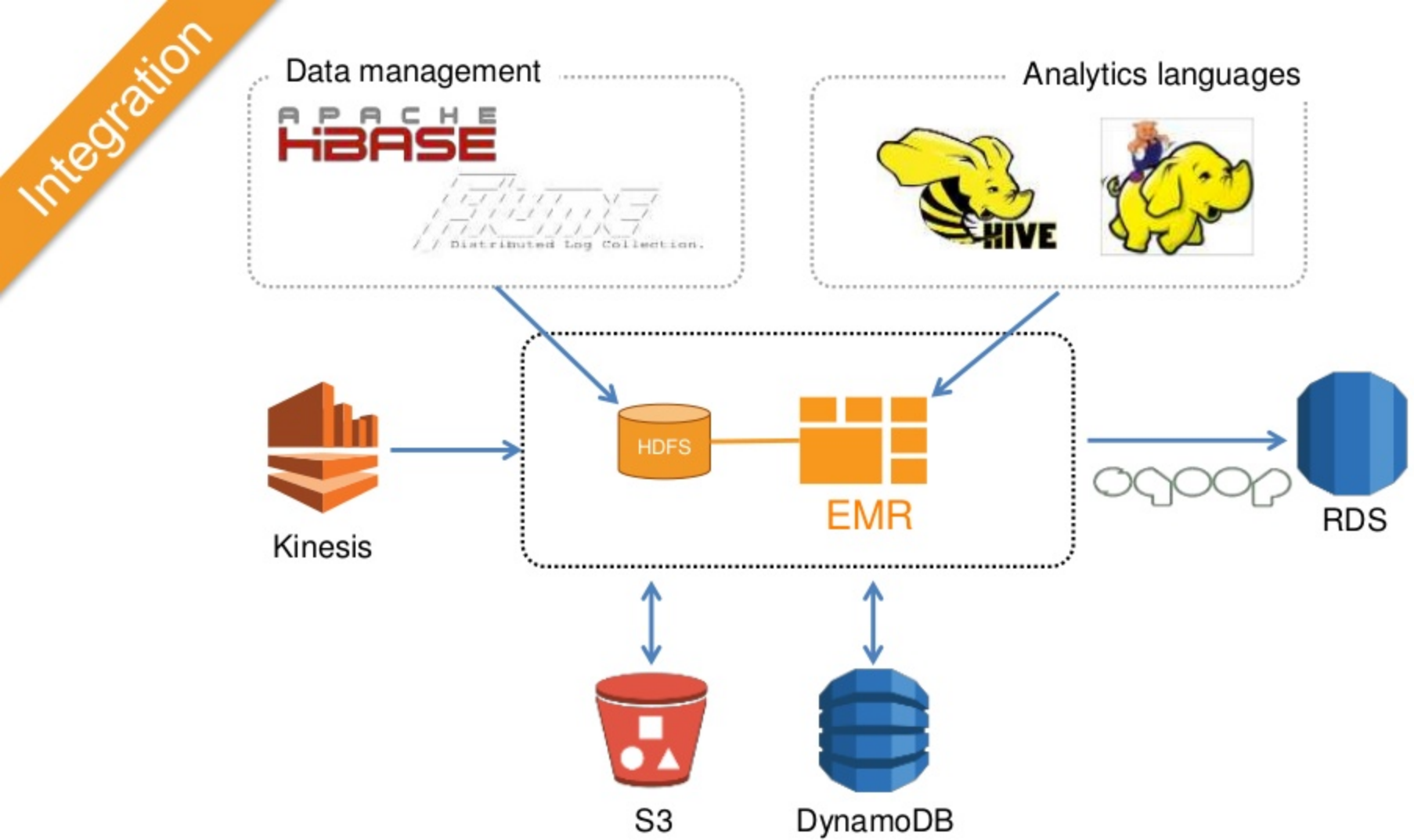


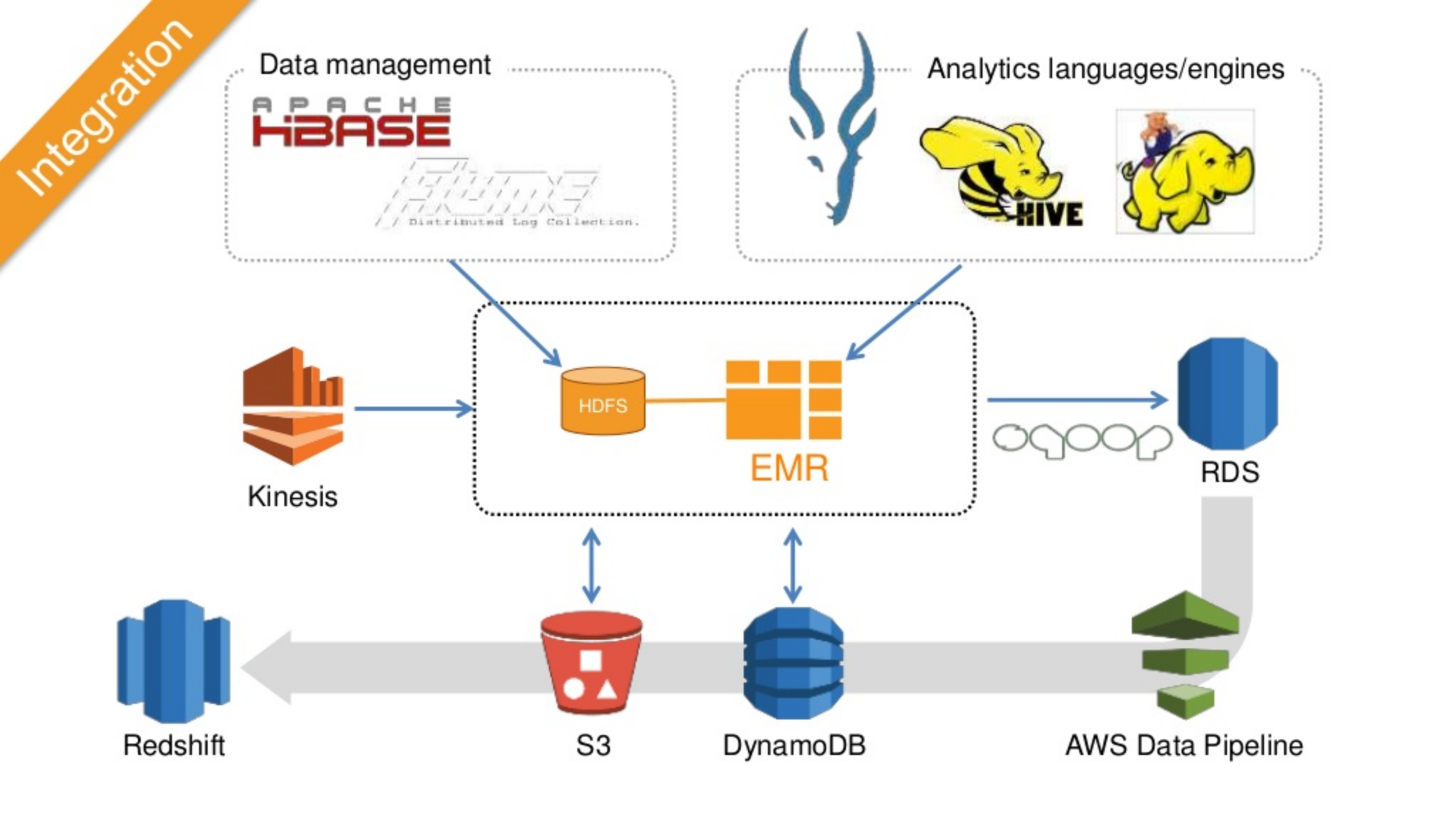
Integration

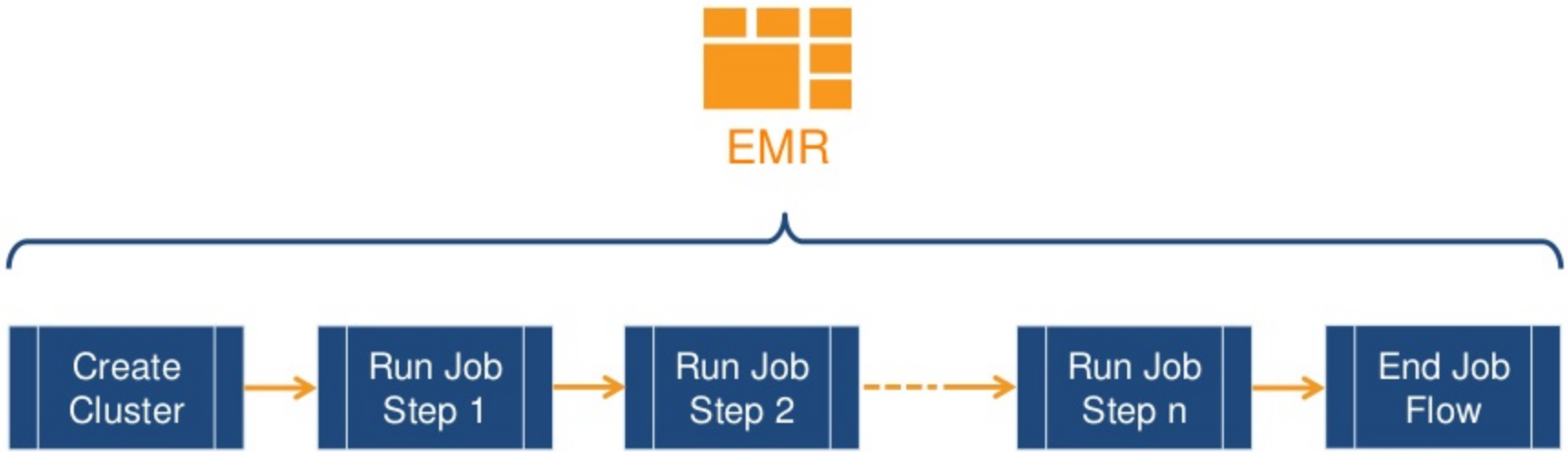



Integration
Data management
APACHE
HBASE
Distributed Log Collection.
HDFS
EMR
Kinesis
S3
DynamoDB


Integration
Data management
APACHE HBASE
Flume
Distributed Log Collection.
Analytics languages
HIVE
HDFS
EMR
Kinesis
S3
DynamoDB


Integration
Data management
APACHE HBASE
Flume
Distributed Log Collection.
Analytics languages
HIVE
HDFS
EMR
Kinesis
sqoop
RDS
S3
DynamoDB


Integration
Data management
APACHE HBASE
Flume
Distributed Log Collection.
Analytics languages/engines
HIVE
HDFS
EMR
Kinesis
RDS
Redshift
S3
DynamoDB
AWS Data Pipeline



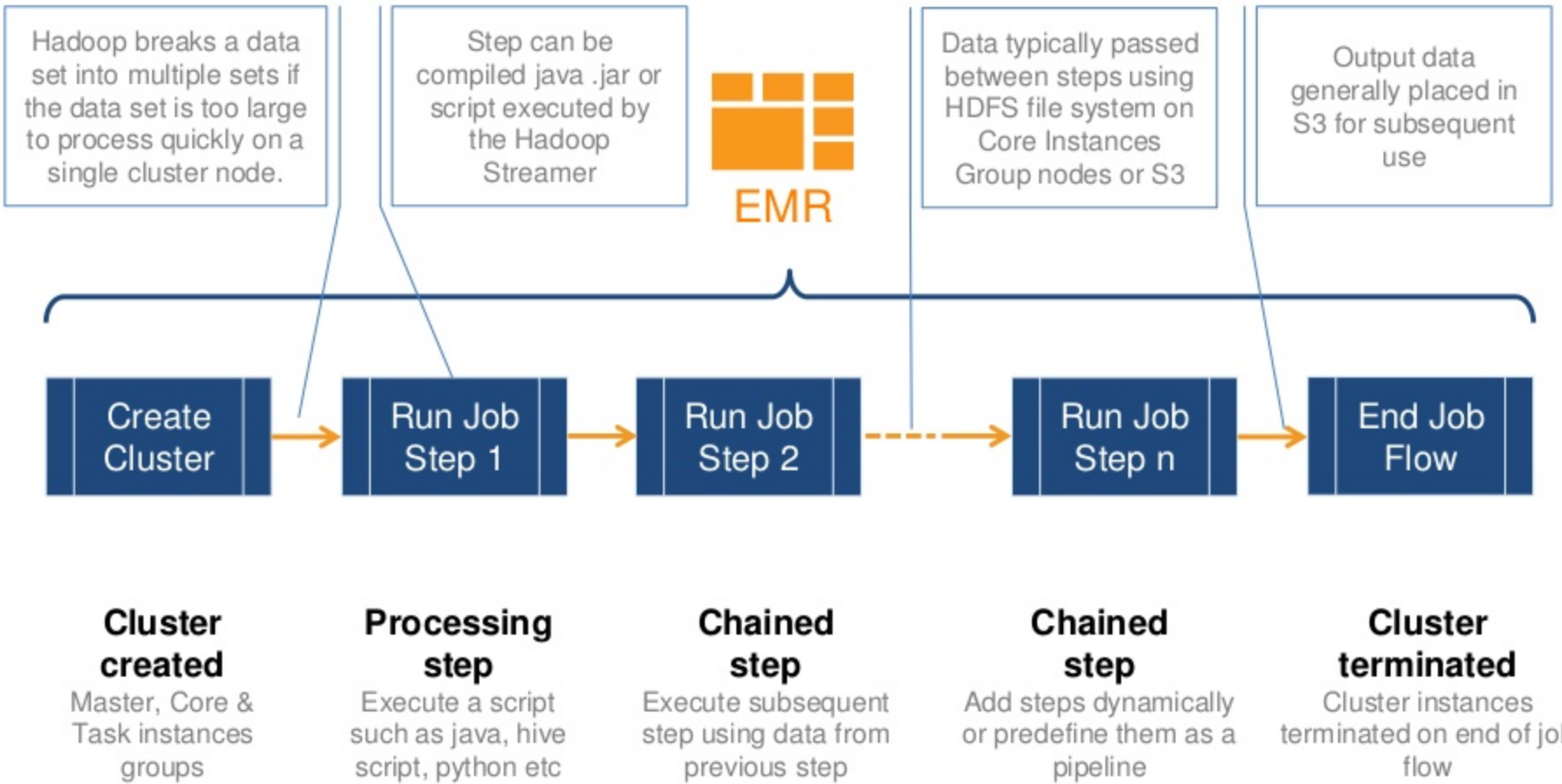
Cluster created
Master, Core & Task instances groups

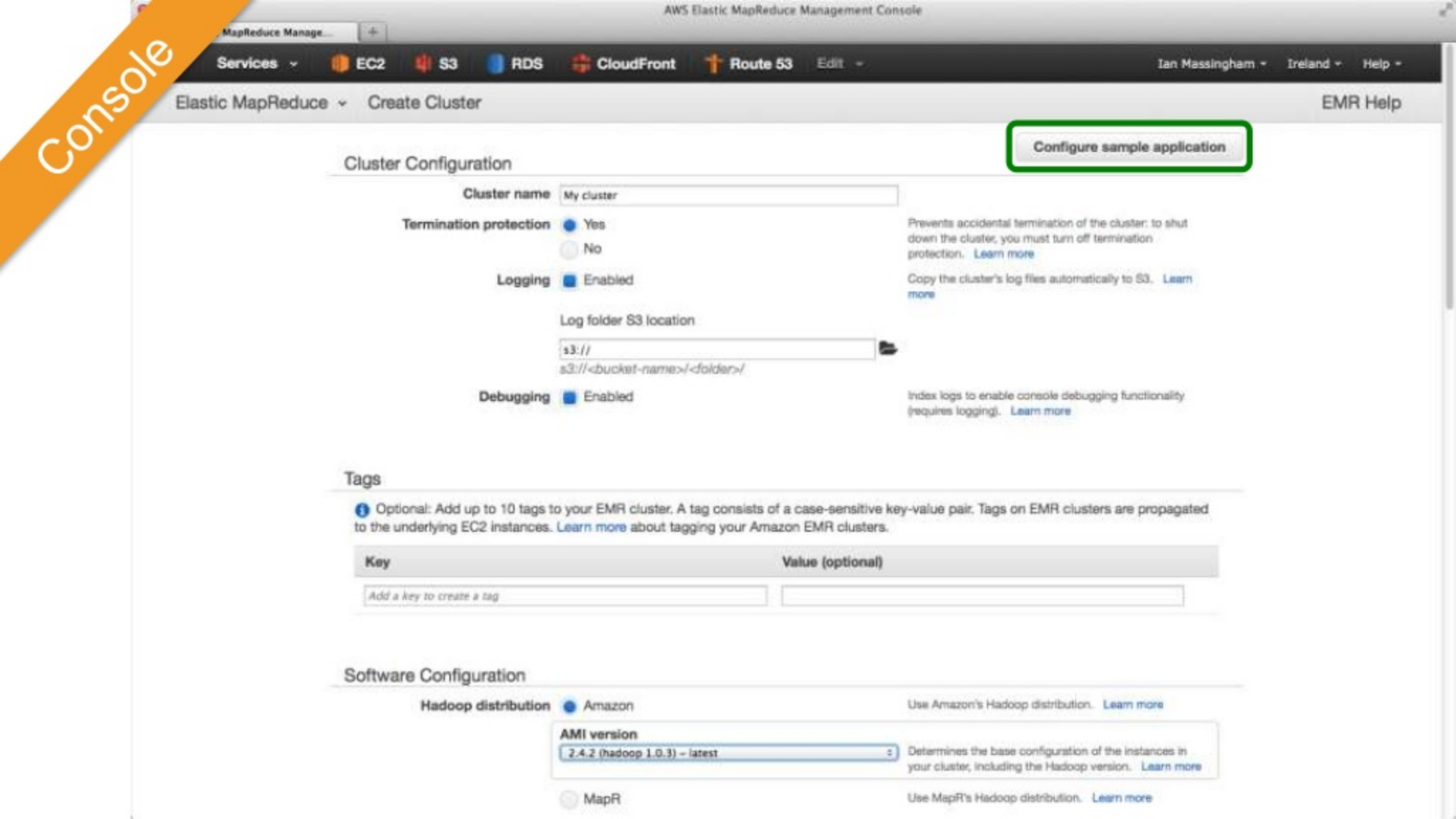
Processing step
Execute a script such as java, hive script, python etc

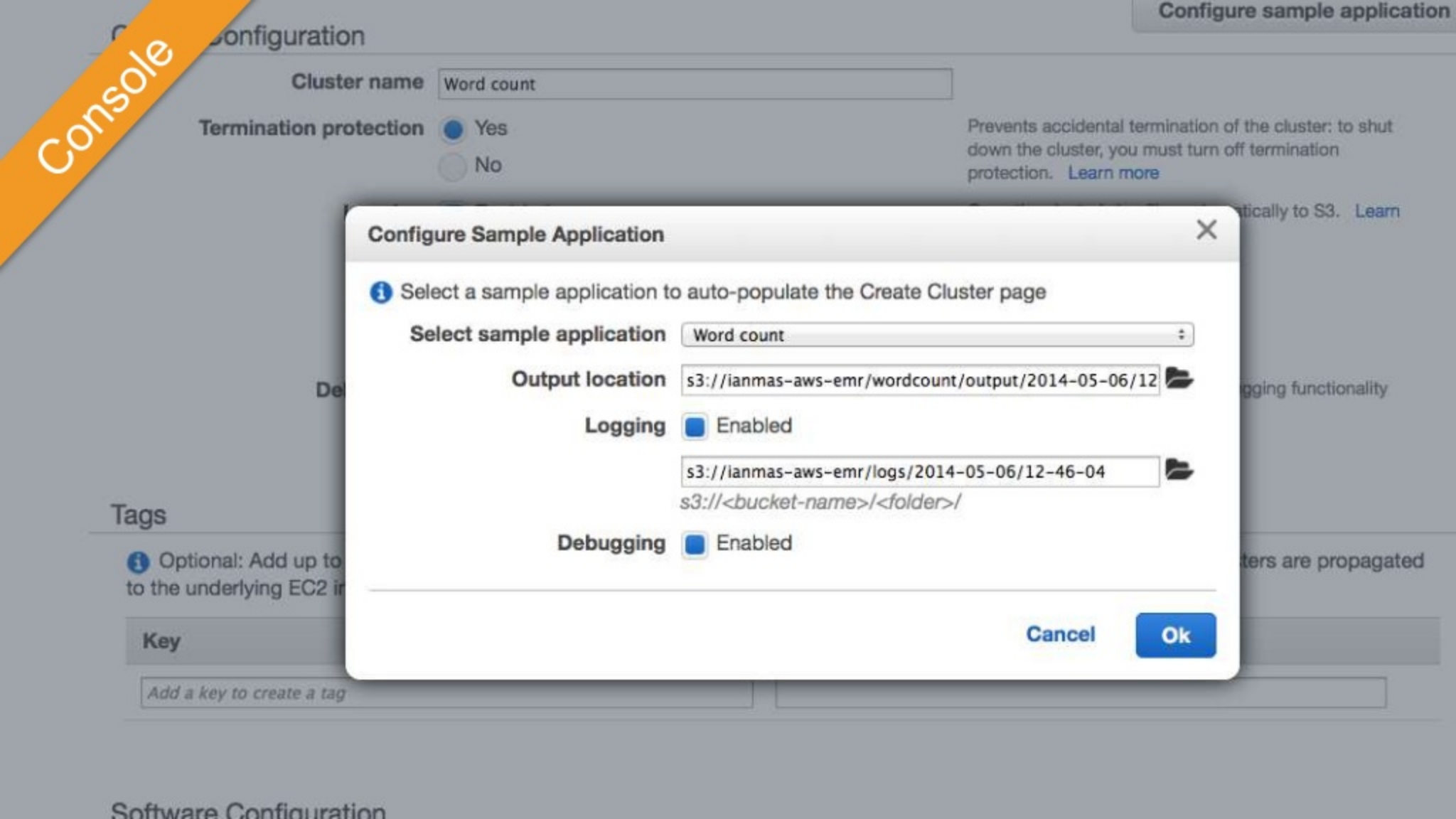
Chained step
Execute subsequent step using data from previous step

Chained step
Add steps dynamically or predefine them as a pipeline

Cluster terminated
Cluster instances terminated on end of job flow


Hadoop breaks a data set into multiple sets if the data set is too large to process quickly on a single cluster node.
Step can be compiled java .jar or script executed by the Hadoop Streamer
EMR
Data typically passed between steps using HDFS file system on Core Instances Group nodes or S3
Output data generally placed in S3 for subsequent use
Create Cluster
Run Job Step 1
Run Job Step 2
Run Job Step n
End Job Flow
Cluster created
Master, Core & Task instances groups
Processing step
Execute a script such as java, hive script, python etc
Chained step
Execute subsequent step using data from previous step
Chained step
Add steps dynamically or predefine them as a pipeline
Cluster terminated
Cluster instances terminated on end of job flow


Console
AWS Elastic MapReduce Management Console
MapReduce Manage...
Services
EC2
S3
RDS
CloudFront
Route 53
Edit
Ian Massingham
Ireland
Help
Elastic MapReduce
Create Cluster
EMR Help
Configure sample application
Cluster Configuration
Cluster name
My cluster
Termination protection
Yes
No
Prevents accidental termination of the cluster: to shut down the cluster, you must turn off termination protection. Learn more
Logging
Enabled
Copy the cluster's log files automatically to S3. Learn more
Log folder S3 location
s3://
s3://<bucket-name>/<folder>/
Debugging
Enabled
Index logs to enable console debugging functionality (requires logging). Learn more
Tags
Optional: Add up to 10 tags to your EMR cluster. A tag consists of a case-sensitive key-value pair. Tags on EMR clusters are propagated to the underlying EC2 instances. Learn more about tagging your Amazon EMR clusters.
Key
Value (optional)
Add a key to create a tag
Software Configuration
Hadoop distribution
Amazon
Use Amazon's Hadoop distribution. Learn more
AMI version
2.4.2 (hadoop 1.0.3) - latest
Determines the base configuration of the instances in your cluster, including the Hadoop version. Learn more
MapR
Use MapR's Hadoop distribution. Learn more


Console
Configure sample application
Configuration
Cluster name
Word count
Termination protection
Yes
No
Prevents accidental termination of the cluster: to shut down the cluster, you must turn off termination protection. Learn more
tically to S3. Learn
gging functionality
Configure Sample Application
Select a sample application to auto-populate the Create Cluster page
Select sample application
Word count
Output location
s3://ianmas-aws-emr/wordcount/output/2014-05-06/12
Logging
Enabled
s3://ianmas-aws-emr/logs/2014-05-06/12-46-04
s3://<bucket-name>/<folder>/
Debugging
Enabled
Cancel
Ok
Tags
Optional: Add up to
to the underlying EC2 ir
ters are propagated
Key
Add a key to create a tag
Software Configuration

Example

Hadoop Streaming

Utility that comes with the Hadoop distribution

Allows you to create and run Map/Reduce jobs with any executable or script as the mapper and/or the reducer

Reads the input from standard input and the reducer outputs data through standard output

By default, each line of input/output represents a record with tab separated key/value

Console

Bootstrap action type	Name	S3 location	Optional arguments

Add bootstrap action: Select a bootstrap action

Configure and add

Steps

A step is a unit of work you submit to the cluster. A step might contain one or more Hadoop jobs, or contain instructions to install or configure an application. You can submit up to 256 steps to a cluster. Learn more

Name	Action on failure	JAR S3 location	Arguments
Word count	Terminate cluster	/home/hadoop/contrib/streaming/hadoop-streaming.jar	-mapper s3://eu-west-1.elasticmapreduce/samples/wordcount/wordSplitter.py -reducer aggregate -input s3://eu-west-1.elasticmapreduce/samples/wordcount/input -output s3://ianmas-aws-emr/wordcount/output/2014-05-06/12-50-13

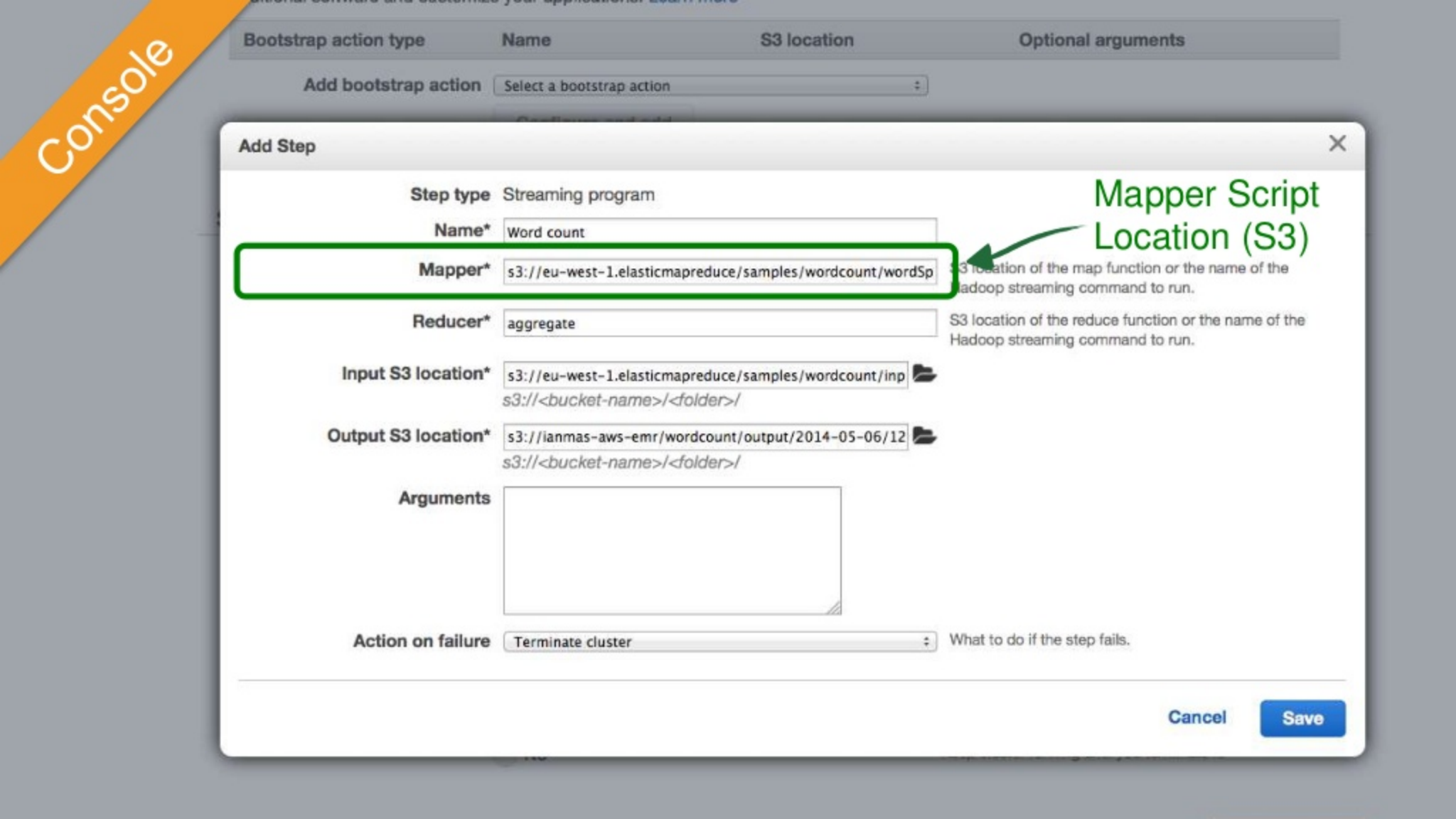
Add step: Select a step

Configure and add

Auto-terminate: Yes — Automatically terminate cluster after the last step is completed.

No — Keep cluster running until you terminate it.

Cancel | Create cluster


Console
Bootstrap action type
Name
S3 location
Optional arguments
Add bootstrap action
Select a bootstrap action
Add Step
Step type
Streaming program
Name*
Word count
Mapper*
s3://eu-west-1.elasticmapreduce/samples/wordcount/wordSp
Mapper Script Location (S3)
S3 location of the map function or the name of the Hadoop streaming command to run.
Reducer*
aggregate
S3 location of the reduce function or the name of the Hadoop streaming command to run.
Input S3 location*
s3://eu-west-1.elasticmapreduce/samples/wordcount/inp
s3://<bucket-name>/<folder>/
Output S3 location*
s3://ianmas-aws-emr/wordcount/output/2014-05-06/12
s3://<bucket-name>/<folder>/
Arguments
Action on failure
Terminate cluster
What to do if the step fails.
Cancel
Save

Streaming: word count python script

Stdin > Stdout

```
#!/usr/bin/python
import sys
import re

def main(argv):
    pattern = re.compile("[a-zA-Z][a-zA-Z0-9]*")
    for line in sys.stdin:
        for word in pattern.findall(line):
            print "LongValueSum:" + word.lower() + "\t" + "1"


if __name__ == "__main__":
    main(sys.argv)
```

Streaming: word count python script

Stdin > Stdout

```
#!/usr/bin/python
import sys
import re

def main(argv):
    pattern = re.compile("[a-zA-Z][a-zA-Z0-9]*")
    for line in sys.stdin:
        for word in pattern.findall(line):
            print "LongValueSum:" + word.lower() + "\t" + "1"


if __name__ == "__main__":
    main(sys.argv)
```

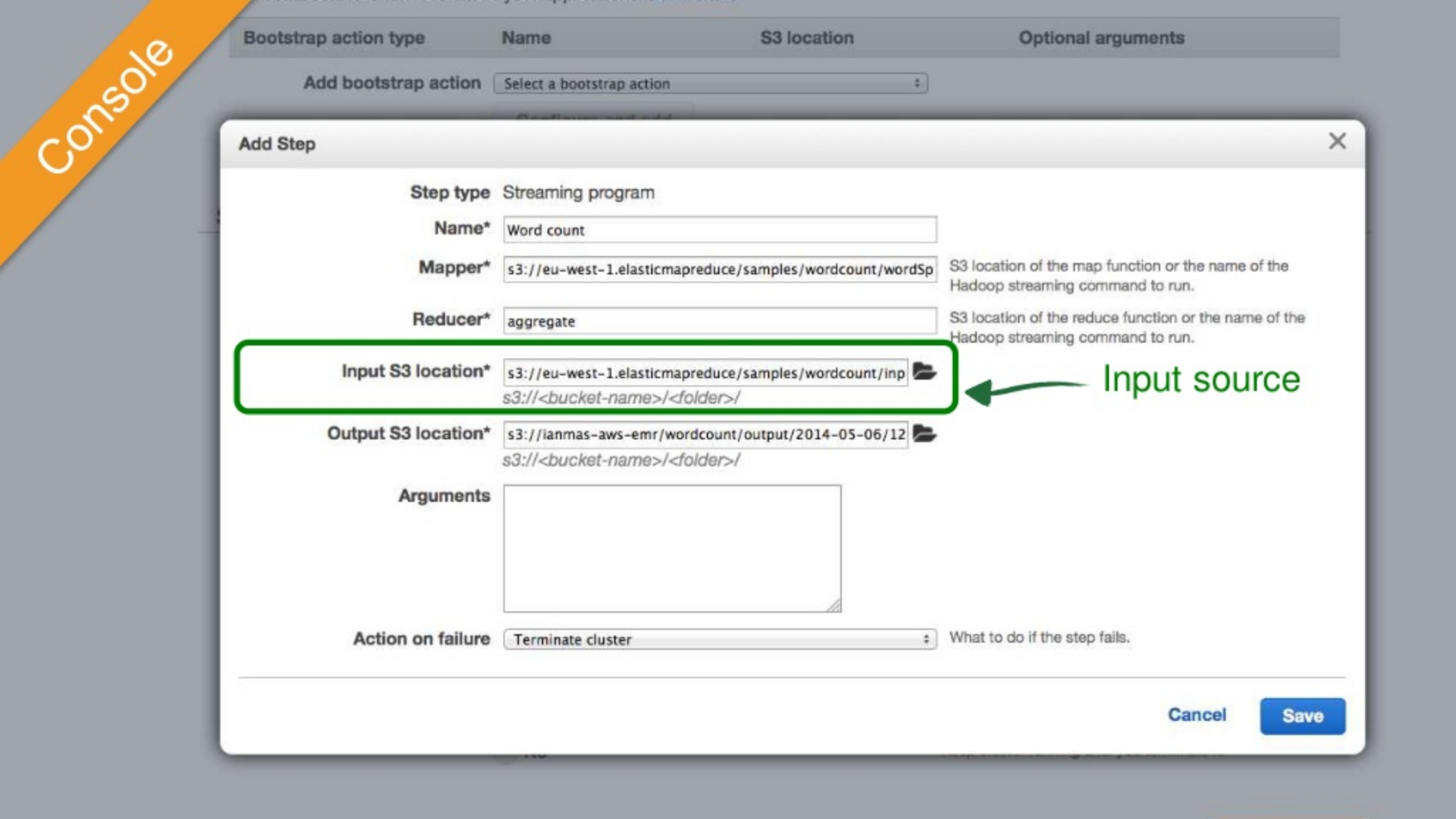
Read words from StdIn line by line

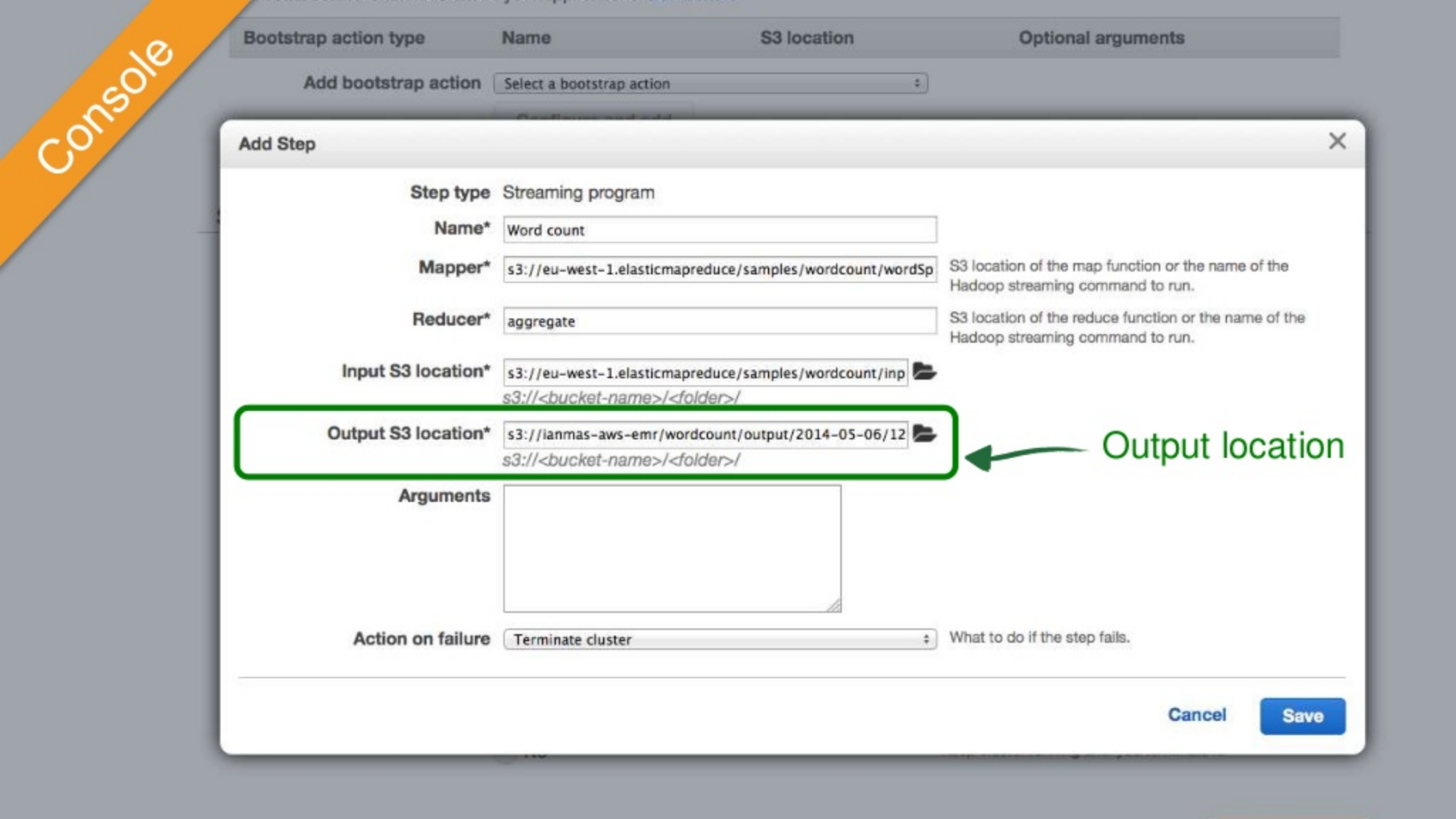
Streaming: word count python script

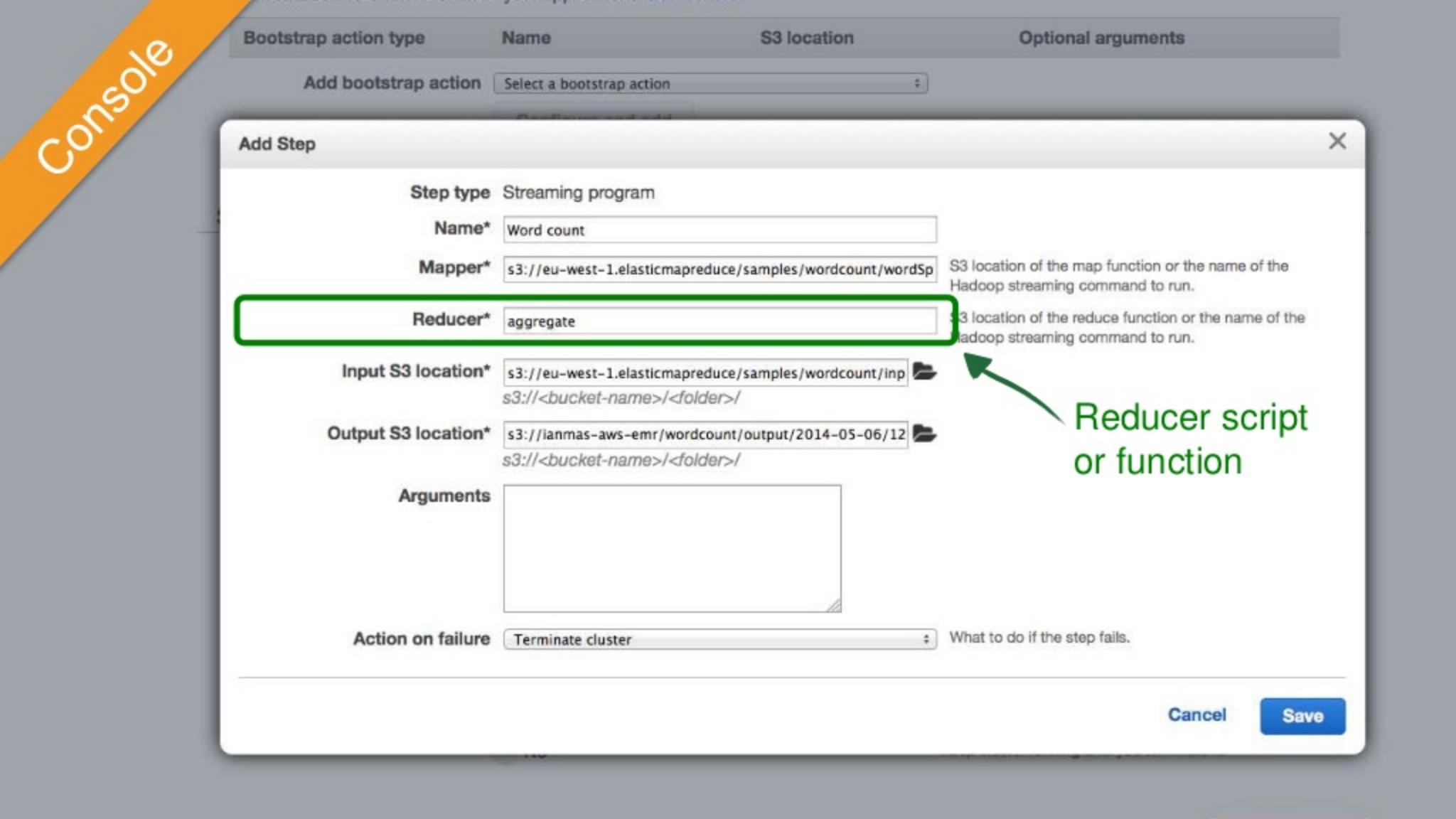
Stdin > Stdout

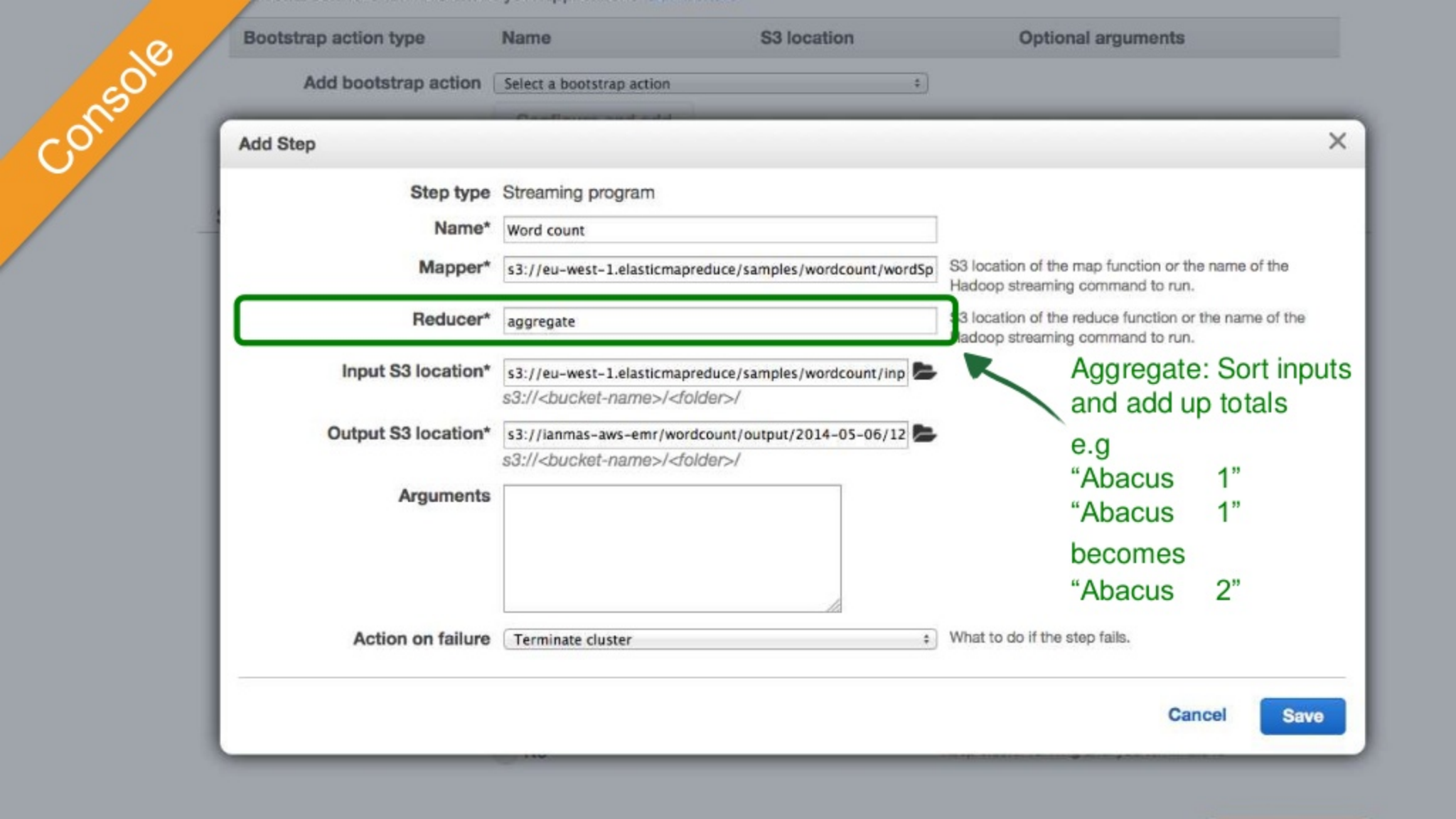
```
#!/usr/bin/python
import sys
import re

def main(argv):
    pattern = re.compile("[a-zA-Z][a-zA-Z0-9]*")
    for line in sys.stdin:
        for word in pattern.findall(line):
            print "LongValueSum:" + word.lower() + "\t" + "1"


if __name__ == "__main__":
    main(sys.argv)
```

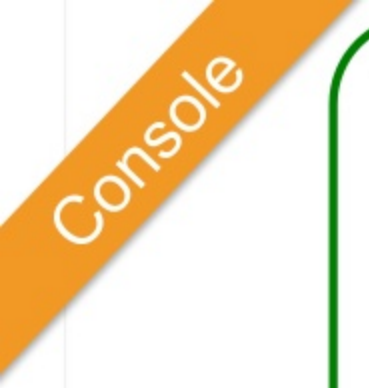
Output to StdOut tab delimited record
e.g. “LongValueSum:abacus 1”


Console
Bootstrap action type
Name
S3 location
Optional arguments
Add bootstrap action
Select a bootstrap action
Add Step
Step type
Streaming program
Name*
Word count
Mapper*
s3://eu-west-1.elasticmapreduce/samples/wordcount/wordSp
S3 location of the map function or the name of the Hadoop streaming command to run.
Reducer*
aggregate
S3 location of the reduce function or the name of the Hadoop streaming command to run.
Input S3 location*
s3://eu-west-1.elasticmapreduce/samples/wordcount/inp
s3://<bucket-name>/<folder>/
Input source
Output S3 location*
s3://ianmas-aws-emr/wordcount/output/2014-05-06/12
s3://<bucket-name>/<folder>/
Arguments
Action on failure
Terminate cluster
What to do if the step fails.
Cancel
Save


Console
Bootstrap action type
Name
S3 location
Optional arguments
Add bootstrap action
Select a bootstrap action
Add Step
Step type
Streaming program
Name*
Word count
Mapper*
s3://eu-west-1.elasticmapreduce/samples/wordcount/wordSp
S3 location of the map function or the name of the Hadoop streaming command to run.
Reducer*
aggregate
S3 location of the reduce function or the name of the Hadoop streaming command to run.
Input S3 location*
s3://eu-west-1.elasticmapreduce/samples/wordcount/inp
s3://<bucket-name>/<folder>/
Output S3 location*
s3://ianmas-aws-emr/wordcount/output/2014-05-06/12
s3://<bucket-name>/<folder>/
Output location
Arguments
Action on failure
Terminate cluster
What to do if the step fails.
Cancel
Save


Console
Bootstrap action type
Name
S3 location
Optional arguments
Add bootstrap action
Select a bootstrap action
Add Step
Step type
Streaming program
Name*
Word count
Mapper*
s3://eu-west-1.elasticmapreduce/samples/wordcount/wordSp
S3 location of the map function or the name of the Hadoop streaming command to run.
Reducer*
aggregate
3 location of the reduce function or the name of the adoop streaming command to run.
Input S3 location*
s3://eu-west-1.elasticmapreduce/samples/wordcount/inp
s3://<bucket-name>/<folder>/
Output S3 location*
s3://ianmas-aws-emr/wordcount/output/2014-05-06/12
s3://<bucket-name>/<folder>/
Arguments
Action on failure
Terminate cluster
What to do if the step fails.
Cancel
Save
Reducer script or function


Console
Bootstrap action type
Name
S3 location
Optional arguments
Add bootstrap action
Select a bootstrap action
Add Step
Step type
Streaming program
Name*
Word count
Mapper*
s3://eu-west-1.elasticmapreduce/samples/wordcount/wordSp
S3 location of the map function or the name of the Hadoop streaming command to run.
Reducer*
aggregate
3 location of the reduce function or the name of the adoop streaming command to run.
Input S3 location*
s3://eu-west-1.elasticmapreduce/samples/wordcount/inp
s3://<bucket-name>/<folder>/
Output S3 location*
s3://ianmas-aws-emr/wordcount/output/2014-05-06/12
s3://<bucket-name>/<folder>/
Arguments
Action on failure
Terminate cluster
What to do if the step fails.
Cancel
Save
Aggregate: Sort inputs and add up totals
e.g
“Abacus 1”
“Abacus 1”
becomes
“Abacus 2”



Hardware Configuration

Specify the networking and hardware configuration for your cluster. If you need more than 20 EC2 instances, complete this form. Request Spot instances (unused EC2 capacity) to save money.

Network vpc-b7b8b3d5 (172.31.0.0/16) (default) — Use a Virtual Private Cloud (VPC) to process sensitive data or connect to a private network. Create a VPC

EC2 Subnet No preference (random subnet) — Create a Subnet

	EC2 instance type	Count	Request spot	
Master	m1.small	1	☐	The Master instance assigns Hadoop tasks to core and task nodes, and monitors their status.
Core	m1.small	2	☐	Core instances run Hadoop tasks and store data using the Hadoop Distributed File System (HDFS).
Task	m1.small	0	☐	Task instances run Hadoop tasks.

Security and Access

EC2 key pair Proceed without an EC2 key pair — Use an existing key pair to SSH into the master node of the Amazon EC2 cluster as the user "hadoop". Learn more

IAM user access ○ All other IAM users ● No other IAM users — Control the visibility of this cluster to other IAM users. Learn more

EC2 role Proceed without role — Control permissions for applications on the cluster. Learn more

Bootstrap Actions

Bootstrap actions are scripts that are executed during setup before Hadoop starts on every cluster node. You can use them to install additional software and customize your applications. Learn more

Bootstrap action type	Name	S3 location	Optional arguments
Add bootstrap action	Select a bootstrap action		

Instance groups

Master instance group

Manages the job flow: coordinating the distribution of the MapReduce executable and subsets of the raw data, to the core and task instance groups

It also tracks the status of each task performed, and monitors the health of the instance groups.

To monitor the progress of the job flow, you can SSH into the master node as the Hadoop user and either look at the Hadoop log files directly or access the user interface that Hadoop

Instance groups

Master instance group

Manages the job flow: coordinating the distribution of the MapReduce executable and subsets of the raw data, to the core and task instance groups

It also tracks the status of each task performed, and monitors the health of the instance groups.

To monitor the progress of the job flow, you can SSH into the master node as the Hadoop user and either look at the Hadoop log files directly or access the user interface that Hadoop

Core instance group

Contains all of the core nodes of a job flow.

A core node is an EC2 instance that runs Hadoop map and reduce tasks and stores data using the Hadoop Distributed File System (HDFS).

The EC2 instances you assign as core nodes are capacity that must be allotted for the entire job flow run.

Core nodes run both the DataNodes and TaskTracker Hadoop daemons.

Instance groups

Master instance group

Manages the job flow: coordinating the distribution of the MapReduce executable and subsets of the raw data, to the core and task instance groups

It also tracks the status of each task performed, and monitors the health of the instance groups.

To monitor the progress of the job flow, you can SSH into the master node as the Hadoop user and either look at the Hadoop log files directly or access the user interface that Hadoop

Core instance group

Contains all of the core nodes of a job flow.

A core node is an EC2 instance that runs Hadoop map and reduce tasks and stores data using the Hadoop Distributed File System (HDFS).

The EC2 instances you assign as core nodes are capacity that must be allotted for the entire job flow run.

Core nodes run both the DataNodes and TaskTracker Hadoop daemons.

Task instance group

Contains all of the task nodes in a job flow.

The task instance group is optional.

You can add it when you start the job flow or add a task instance group to a job flow in progress.

You can increase and decrease the number of task nodes. Because they don't store data and can be added and removed from a job flow, you can use task nodes to manage the EC2 instance capacity your job flow uses

Console

Hardware Configuration

Specify the networking and hardware configuration for your cluster. If you need more than 20 EC2 instances, complete this form. Request Spot instances (unused EC2 capacity) to save money.

Network vpc-b7b8b3d5 (172.31.0.0/16) (default) — Use a Virtual Private Cloud (VPC) to process sensitive data or connect to a private network. Create a VPC

EC2 Subnet No preference (random subnet) — Create a Subnet

	EC2 instance type	Count	Request spot	
Master	m1.small	1	☐	The Master instance assigns Hadoop tasks to core and task nodes, and monitors their status.
Core	m1.small	2	☐	Core instances run Hadoop tasks and store data using the Hadoop Distributed File System (HDFS).
Task	m1.small	0	☐	Task instances run Hadoop tasks.

Security and Access

EC2 key pair Proceed without an EC2 key pair — Use an existing key pair to SSH into the master node of the Amazon EC2 cluster as the user "hadoop". Learn more

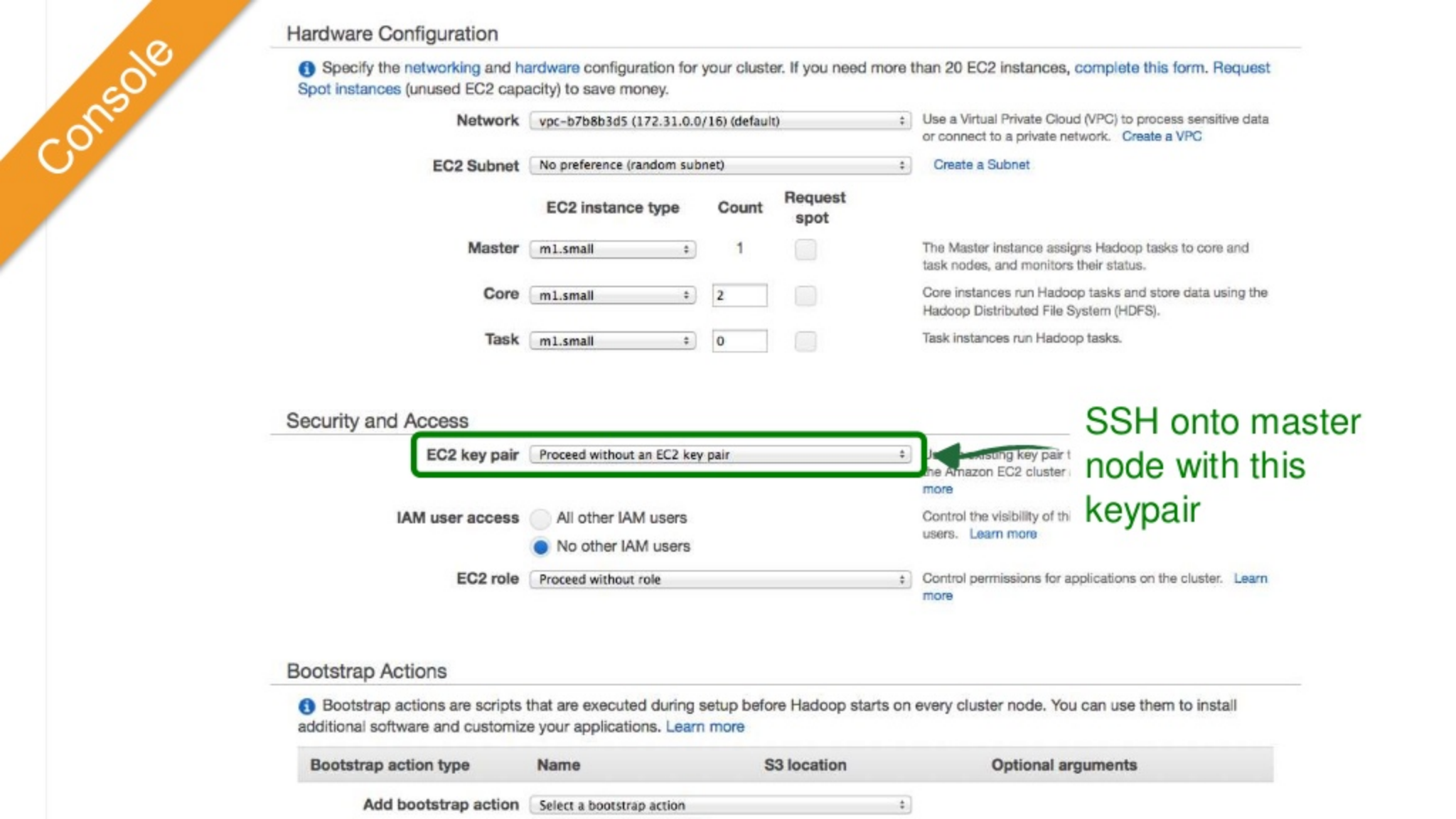
IAM user access ○ All other IAM users ● No other IAM users — Control the visibility of this cluster to other IAM users. Learn more

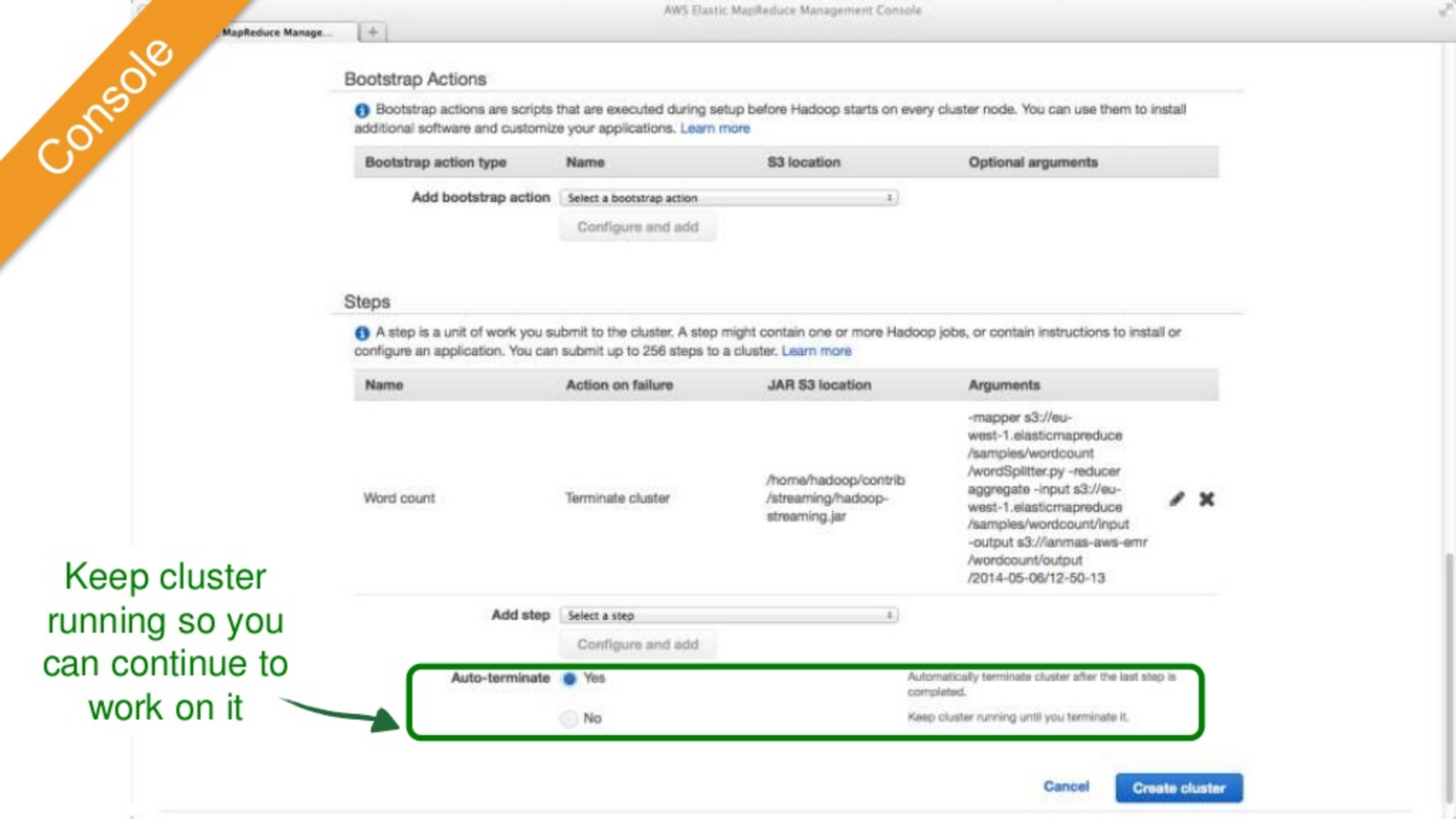
EC2 role Proceed without role — Control permissions for applications on the cluster. Learn more

Bootstrap Actions

Bootstrap actions are scripts that are executed during setup before Hadoop starts on every cluster node. You can use them to install additional software and customize your applications. Learn more

Bootstrap action type	Name	S3 location	Optional arguments
Add bootstrap action	Select a bootstrap action		


Console
Hardware Configuration
Specify the networking and hardware configuration for your cluster. If you need more than 20 EC2 instances, complete this form. Request Spot instances (unused EC2 capacity) to save money.
Network
vpc-b7b8b3d5 (172.31.0.0/16) (default)
Use a Virtual Private Cloud (VPC) to process sensitive data or connect to a private network. Create a VPC
EC2 Subnet
No preference (random subnet)
Create a Subnet
EC2 instance type
Count
Request spot
Master
m1.small
1
The Master instance assigns Hadoop tasks to core and task nodes, and monitors their status.
Core
m1.small
2
Core instances run Hadoop tasks and store data using the Hadoop Distributed File System (HDFS).
Task
m1.small
0
Task instances run Hadoop tasks.
Security and Access
EC2 key pair
Proceed without an EC2 key pair
more
IAM user access
All other IAM users
No other IAM users
Control the visibility of th
users. Learn more
EC2 role
Proceed without role
Control permissions for applications on the cluster. Learn more
SSH onto master node with this keypair
Bootstrap Actions
Bootstrap actions are scripts that are executed during setup before Hadoop starts on every cluster node. You can use them to install additional software and customize your applications. Learn more
Bootstrap action type
Name
S3 location
Optional arguments
Add bootstrap action
Select a bootstrap action


Console
AWS Elastic MapReduce Management Console
MapReduce Manage...
Bootstrap Actions
Bootstrap actions are scripts that are executed during setup before Hadoop starts on every cluster node. You can use them to install additional software and customize your applications. Learn more
Bootstrap action type
Name
S3 location
Optional arguments
Add bootstrap action
Select a bootstrap action
Configure and add
Steps
A step is a unit of work you submit to the cluster. A step might contain one or more Hadoop jobs, or contain instructions to install or configure an application. You can submit up to 256 steps to a cluster. Learn more
Name
Action on failure
JAR S3 location
Arguments
Word count
Terminate cluster
/home/hadoop/contrib/streaming/hadoop-streaming.jar
-mapper s3://eu-west-1.elasticmapreduce/samples/wordcount/wordSplitter.py -reducer aggregate -input s3://eu-west-1.elasticmapreduce/samples/wordcount/input -output s3://anmas-aws-emr/wordcount/output/2014-05-06/12-50-13
Add step
Select a step
Configure and add
Auto-terminate
Yes
Automatically terminate cluster after the last step is completed.
No
Keep cluster running until you terminate it.
Cancel
Create cluster
Keep cluster running so you can continue to work on it

S3 output

```
…
aakar   3
abdal   3
abdelaziz   18
abolish 3
aboriginal  12
abraham 6
abseron 9
abstentions 15
accession   73
accord  90
achabar 3
achievements3
acic4
acknowledged6
acquis  9
acquisitions3

…
```

CLI

```
./elastic-mapreduce --create --stream
--mapper
      s3://elasticmapreduce/samples/wordcount/wordSplitter.py
--input
      s3://elasticmapreduce/samples/wordcount/input
--output
      s3n://myawsbucket/output
--reducer
      aggregate
```

Installation guide for EMR CLI tools: docs.aws.amazon.com/ElasticMapReduce/latest/DeveloperGuide/emr-cli-install.html

CLI

```
./elastic-mapreduce --create --stream
--mapper
      s3://elasticmapreduce/samples/wordcount/wordSplitter.py
--input
      s3://elasticmapreduce/samples/wordcount/input
--output
      s3n://myawsbucket/output
--reducer
      aggregate
```

Installation guide for EMR CLI tools: docs.aws.amazon.com/ElasticMapReduce/latest/DeveloperGuide/emr-cli-install.html

CLI

```
./elastic-mapreduce --create --stream
--mapper
      s3://elasticmapreduce/samples/wordcount/wordSplitter.py
--input
      s3://elasticmapreduce/samples/wordcount/input
--output
      s3n://myawsbucket/output
--reducer
      aggregate
```

Installation guide for EMR CLI tools: docs.aws.amazon.com/ElasticMapReduce/latest/DeveloperGuide/emr-cli-install.html

CLI

```
./elastic-mapreduce --create --stream
--mapper
      s3://elasticmapreduce/samples/wordcount/wordSplitter.py
--input
      s3://elasticmapreduce/samples/wordcount/input
--output
      s3n://myawsbucket/output
--reducer
      aggregate
```

Installation guide for EMR CLI tools: docs.aws.amazon.com/ElasticMapReduce/latest/DeveloperGuide/emr-cli-install.html

CLI

```
./elastic-mapreduce --create --stream
--mapper
      s3://elasticmapreduce/samples/wordcount/wordSplitter.py
--input
      s3://elasticmapreduce/samples/wordcount/input
--output
      s3n://myawsbucket/output
--reducer
      aggregate
```

Installation guide for EMR CLI tools: docs.aws.amazon.com/ElasticMapReduce/latest/DeveloperGuide/emr-cli-install.html

Defaults

launches a job flow to run on a single-node cluster using an Amazon EC2 m1.small instance

when your steps are running correctly on a small set of sample data, you can launch job flows to run on multiple nodes

You can specify the number of nodes and the type of instance to run with the `--num-instances` and `--instance-type` parameters

Starting a flow

```
elastic-mapreduce
--create
--key-pair micro
--region eu-west-1
--name MyJobFlow
--num-instances 5
--instance-type m2.4xlarge
--alive
--log-uri s3n://mybucket/EMR/log
```

Starting a flow

```
elastic-mapreduce
--create
--key-pair micro
--region eu-west-1
--name MyJobFlow
--num-instances 5
--instance-type m2.4xlarge
--alive
--log-uri s3n://mybucket/EMR/log
```

Instance type/count

Starting a flow

```
elastic-mapreduce
--create
--key-pair micro
--region eu-west-1
--name MyJobFlow
--num-instances 5
--instance-type m2.4xlarge
--alive
--log-uri s3n://mybucket/EMR/log
```

Don’t terminate cluster – keep it running so steps can be added

Starting a flow

```
elastic-mapreduce
--create
--key-pair micro
--region eu-west-1
--name MyJobFlow
--num-instances 5
--instance-type m2.4xlarge
--alive
--pig-interactive --pig-versions latest
--hive-interactive --hive-versions latest
--hbase
--log-uri s3n://mybucket/EMR/log
```

Adding Hive, Pig and Hbase to the job flow

Starting a flow

Using a bootstrapping action to configure the cluster

```
elastic-mapreduce
--create
--bootstrap-action s3://elasticmapreduce/bootstrap-actions/configure-hadoop --args
”-s,dfs.block.size=1048576”
--key-pair micro
--region eu-west-1
--name MyJobFlow
--num-instances 5
--instance-type m2.4xlarge
–-alive
--pig-interactive --pig-versions latest
--hive-interactive –-hive-versions latest
--hbase
--log-uri s3n://mybucket/EMR/log
```

You can specify up to 16 bootstrap actions per job flow by providing multiple *bootstrap-action* parameters

Bootstrapping

s3://elasticmapreduce/bootstrap-actions/**configure-daemons**

Configure JVM properties

```
--bootstrap-action s3://elasticmapreduce/bootstrap-actions/configure-daemons --args --namenode-heap-size=2048,--namenode-opts=-XX:GCTimeRatio=19
```

s3://elasticmapreduce/bootstrap-actions/**configure-hadoop**

Configure cluster wide hadoop settings

```
./elastic-mapreduce –create --bootstrap-action s3://elasticmapreduce/bootstrap-actions/configure-hadoop --args "--site-config-file,s3://myawsbucket/config.xml,-s,mapred.tasktracker.map.tasks.maximum=2"
```

s3://elasticmapreduce/bootstrap-actions/configurations/latest/**memory-intensive**

Configure cluster wide memory settings

```
./elastic-mapreduce –create --bootstrap-action s3://elasticmapreduce/bootstrap-actions/configurations/latest/memory-intensive
```

You can specify up to 16 bootstrap actions per job flow by providing multiple *bootstrap-action* parameters

Bootstrapping

s3://elasticmapreduce/bootstrap-actions/**run-if**

Conditionally run a command

```
./elastic-mapreduce --create --alive \
--bootstrap-action s3://elasticmapreduce/bootstrap-actions/run-if \
--args "instance.isMaster=true,echo running on master node"
```

Shutdown actions

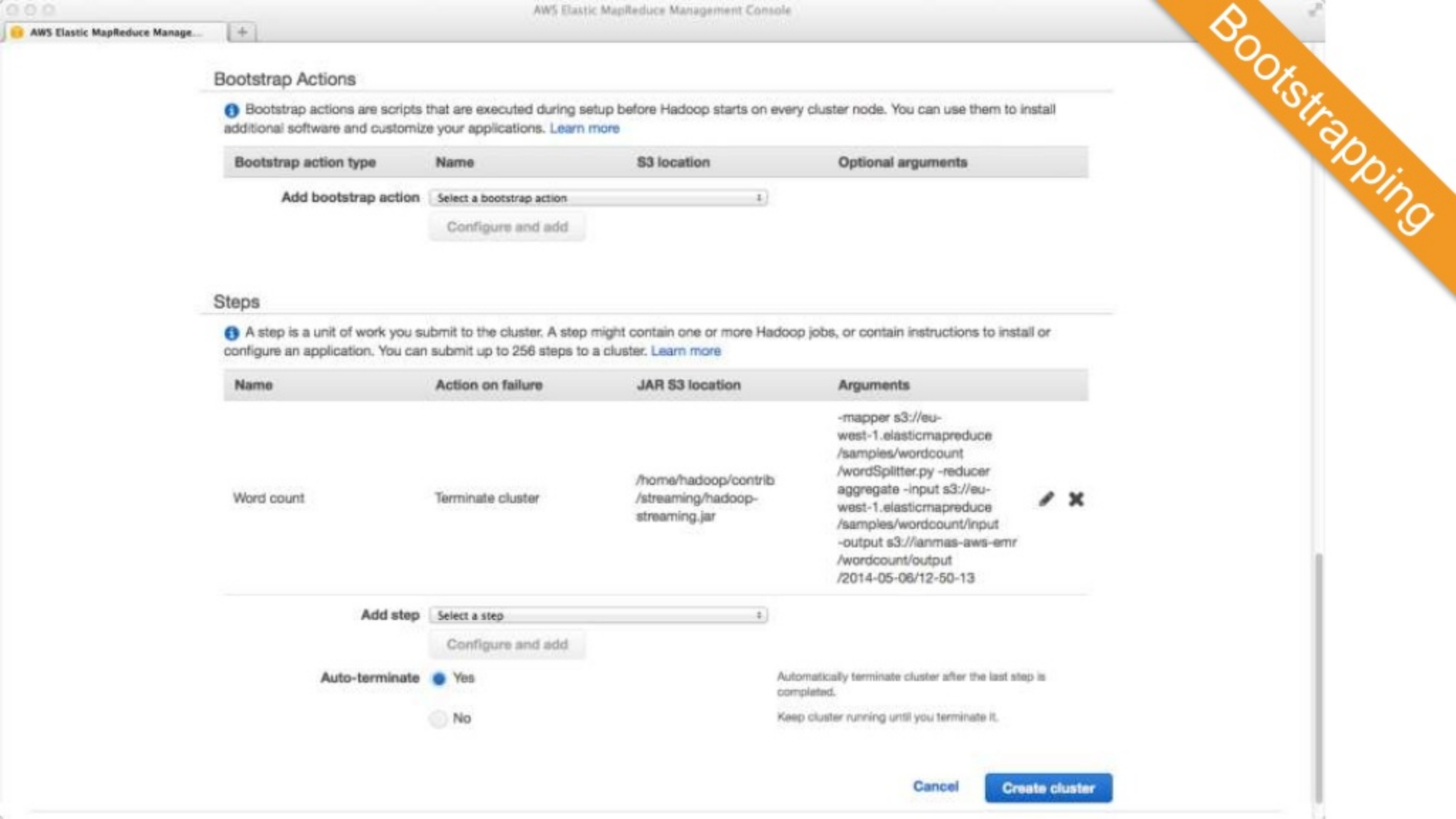
Sun scripts on shutdown

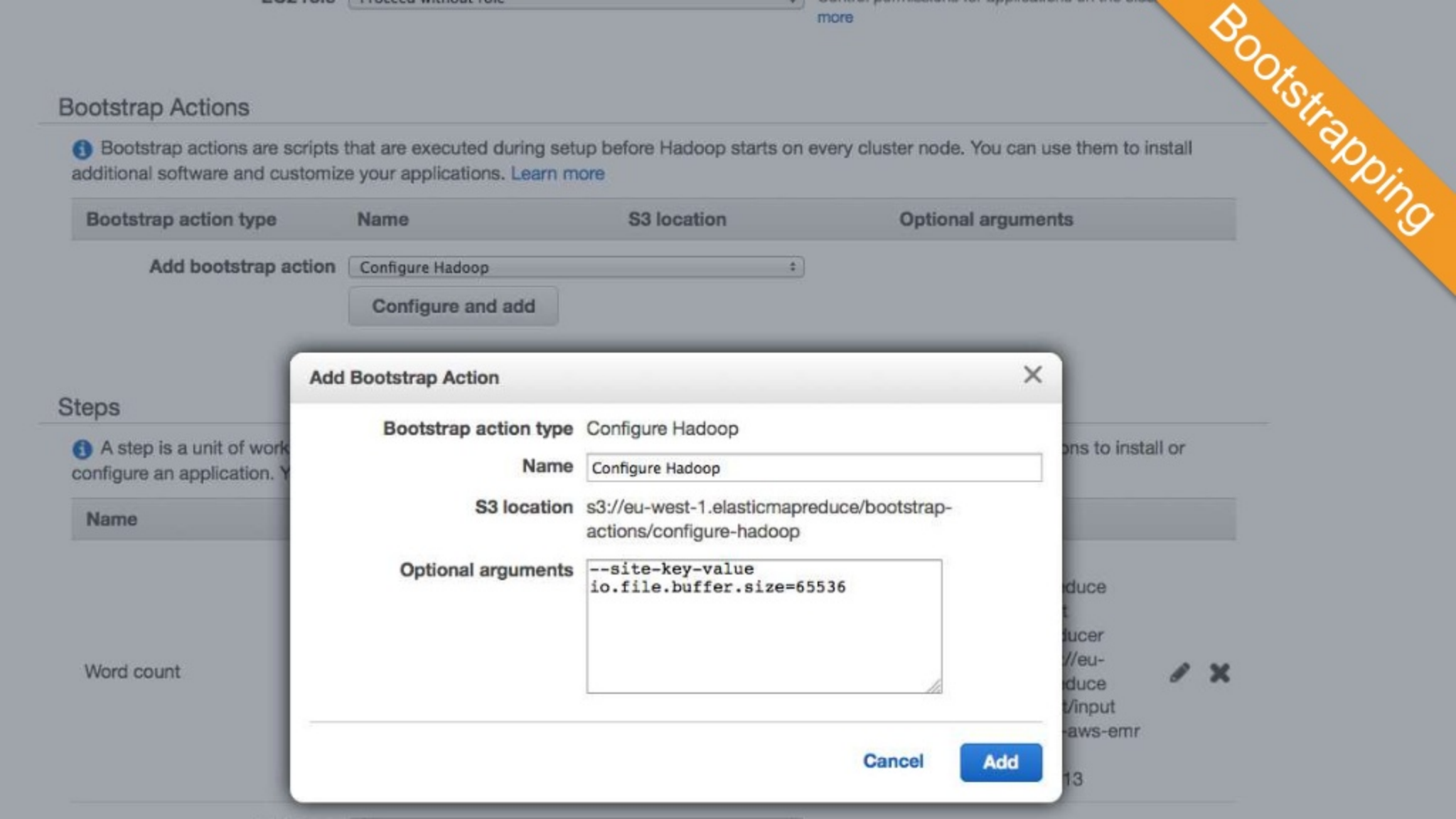
A bootstrap action script can create one or more shutdown actions by writing scripts to the /mnt/var/lib/instance-controller/public/shutdown-actions/ directory. Each script must run and complete within 60 seconds.

Custom actions

Run a custom script

```
--bootstrap-action "s3://elasticmapreduce/bootstrap-actions/download.sh"
```


AWS Elastic MapReduce Management Console
AWS Elastic MapReduce Manage...
Bootstrapping
Bootstrap Actions
Bootstrap actions are scripts that are executed during setup before Hadoop starts on every cluster node. You can use them to install additional software and customize your applications. Learn more
Bootstrap action type
Name
S3 location
Optional arguments
Add bootstrap action
Select a bootstrap action
Configure and add
Steps
A step is a unit of work you submit to the cluster. A step might contain one or more Hadoop jobs, or contain instructions to install or configure an application. You can submit up to 256 steps to a cluster. Learn more
Name
Action on failure
JAR S3 location
Arguments
Word count
Terminate cluster
/home/hadoop/contrib /streaming/hadoop-streaming.jar
-mapper s3://eu-west-1.elasticmapreduce /samples/wordcount /wordSplitter.py -reducer aggregate -input s3://eu-west-1.elasticmapreduce /samples/wordcount/input -output s3://ianmas-aws-emr /wordcount/output /2014-05-06/12-50-13
Add step
Select a step
Configure and add
Auto-terminate
Yes
Automatically terminate cluster after the last step is completed.
No
Keep cluster running until you terminate it.
Cancel
Create cluster


Bootstrapping
more
Bootstrap Actions
Bootstrap actions are scripts that are executed during setup before Hadoop starts on every cluster node. You can use them to install additional software and customize your applications. Learn more
Bootstrap action type
Name
S3 location
Optional arguments
Add bootstrap action
Configure Hadoop
Configure and add
Steps
A step is a unit of work
configure an application. Y
ons to install or
Name
Word count
Add Bootstrap Action
Bootstrap action type
Configure Hadoop
Name
Configure Hadoop
S3 location
s3://eu-west-1.elasticmapreduce/bootstrap-actions/configure-hadoop
Optional arguments
--site-key-value
io.file.buffer.size=65536
Cancel
Add

Integrated tools

Selecting the right one

Hadoop streaming

Unstructured data with wide variation of formats, high flexibility on language

Broad language support (Python, Perl, Bash…)

Mapper, reducer, input and output all supplied by the jobflow

Scale from 1 to unlimited number of processes

Hadoop streaming

Unstructured data with wide variation of formats, high flexibility on language

Broad language support (Python, Perl, Bash…)

Mapper, reducer, input and output all supplied by the jobflow

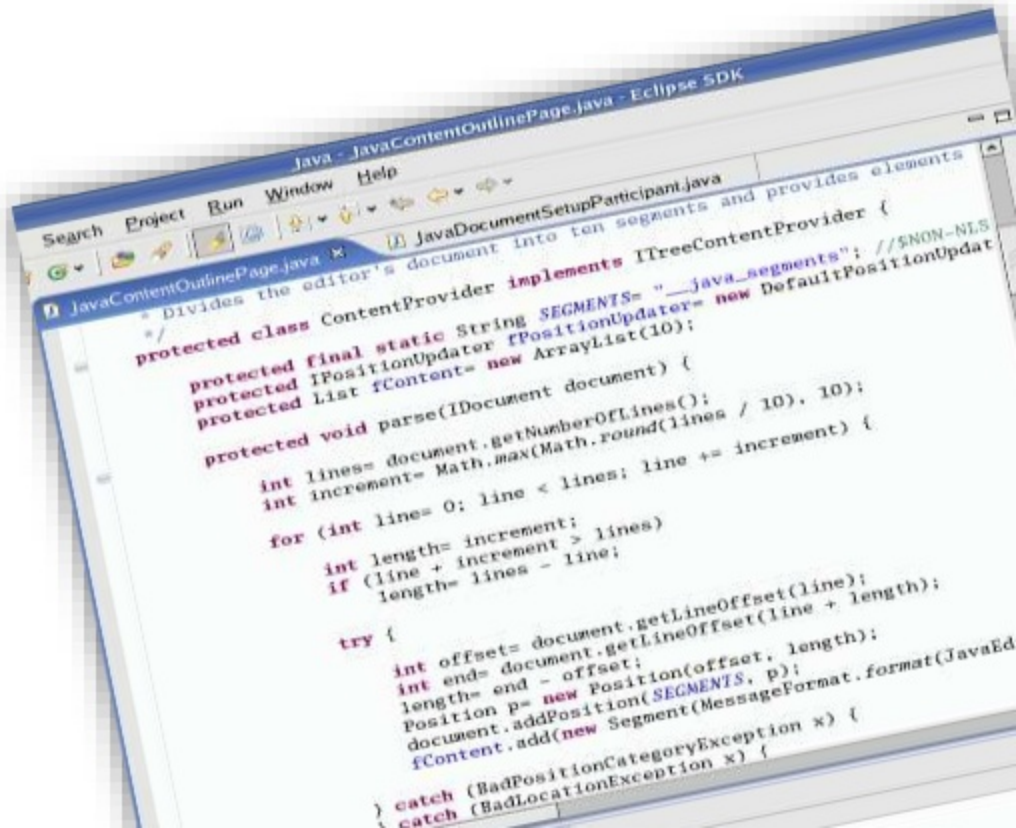
Scale from 1 to unlimited number of processes

Doing analytics in Eclipse is wrong

2-stage dataflow is rigid

Joins are lengthy and error-prone

Prototyping/exploration requires compile & deploy



Hadoop streaming

Unstructured data with wide variation of formats, high flexibility on language

Broad language support (Python, Perl, Bash…)

Mapper, reducer, input and output all supplied by the jobflow

Scale from 1 to unlimited number of processes

Hive

Structured data which is lower value or where higher latency is tolerable

Pig

Semi-structured data which can be mined using set operations

Hbase

Near real time K/V store for structured data

Hive integration

Schema on read

Hive

SQL Interface for working with data

Simple way to use Hadoop

Create Table statement references data location on S3

Language called HiveQL, similar to SQL

An example of a query could be:

```
SELECT COUNT(1) FROM sometable;
```

Requires to setup a mapping to the input data

Uses SerDes to make it possible to query different input formats

Powerful data types (Array & Map..)

Hive

	SQL	HiveQL
Updates	UPDATE, INSERT, DELETE	INSERT, OVERWRITE TABLE
Transactions	Supported	Not supported
Indexes	Supported	Not supported
Latency	Sub-second	Minutes
Functions	Hundreds	Dozens
Multi-table inserts	Not supported	Supported
Create table as select	Not valid SQL-92	Supported

JDBC

JDBC Listener started by default on EMR

Port 10003 for Hive 0.8.1, Ports 9100-9101 for Hadoop

Download hive-jdbc-0.8.1.jar

Access by:

Open port on security group ElasticMapReduce-master

SSH Tunnel

CLI

HiveQL to execute

```
./elastic-mapreduce –create
--name "Hive job flow”
--hive-script
--args s3://myawsbucket/myquery.q
--args -d,INPUT=s3://myawsbucket/input,-
d,OUTPUT=s3://myawsbucket/output
```

CLI

```
./elastic-mapreduce
--create
--alive
--name "Hive job flow”
--num-instances 5 --instance-type m1.large \
--hive-interactive
```

Interactive hive session

HiveQL

```
CREATE EXTERNAL TABLE impressions (
        requestBeginTime string,
        adId string,
        impressionId string,
        referrer string,
        userAgent string,
        userCookie string,
        ip string
    )
    PARTITIONED BY (dt string)
    ROW FORMAT
        serde 'com.amazon.elasticmapreduce.JsonSerde'
        with serdeproperties ( 'paths'='requestBeginTime,
      adId, impressionId, referrer, userAgent,        userCookie,
ip' )
    LOCATION ‘s3://mybucketsource/tables/impressions' ;
```

HiveQL

```
CREATE EXTERNAL TABLE impressions (
        requestBeginTime string,
        adId string,
        impressionId string,
        referrer string,
        userAgent string,
        userCookie string,
        ip string
    )
    PARTITIONED BY (dt string)
    ROW FORMAT
        serde 'com.amazon.elasticmapreduce.JsonSerde'
        with serdeproperties ( 'paths'='requestBeginTime,
      adId, impressionId, referrer, userAgent,        userCookie,
ip' )
    LOCATION ‘s3://mybucketsource/tables/impressions' ;
```

Table structure to create (happens fast as just mapping to source)

HiveQL

```
CREATE EXTERNAL TABLE impressions (
        requestBeginTime string,
        adId string,
        impressionId string,
        referrer string,
        userAgent string,
        userCookie string,
        ip string
    )
    PARTITIONED BY (dt string)
    ROW FORMAT
        serde 'com.amazon.elasticmapreduce.JsonSerde'
        with serdeproperties ( 'paths'='requestBegi
      adId, impressionId, referrer, userAgent,
ip' )
    LOCATION ‘s3://mybucketsource/tables/impressions' ;
```

Source data in S3

HiveQL

```
hive> select * from impressions limit 5;
```

Selecting from source
data directly via Hadoop

	Streaming	Hive	Pig	DynamoDB	Redshift
Unstructured Data	✓			✓	
Structured Data		✓	✓	✓	✓
Language Support	Any*	HQL	Pig Latin	Client	SQL
SQL		✓			✓
Volume	Unlimited	Unlimited	Unlimited	Relatively Low	1.6 PB
Latency	Medium	Medium	Medium	Ultra Low	Low

Cost considerations

Making the most of EC2 resources

1 instance for 100 hours

=

100 instances for 1 hour

Small instance = $8

1 instance for 1,000 hours

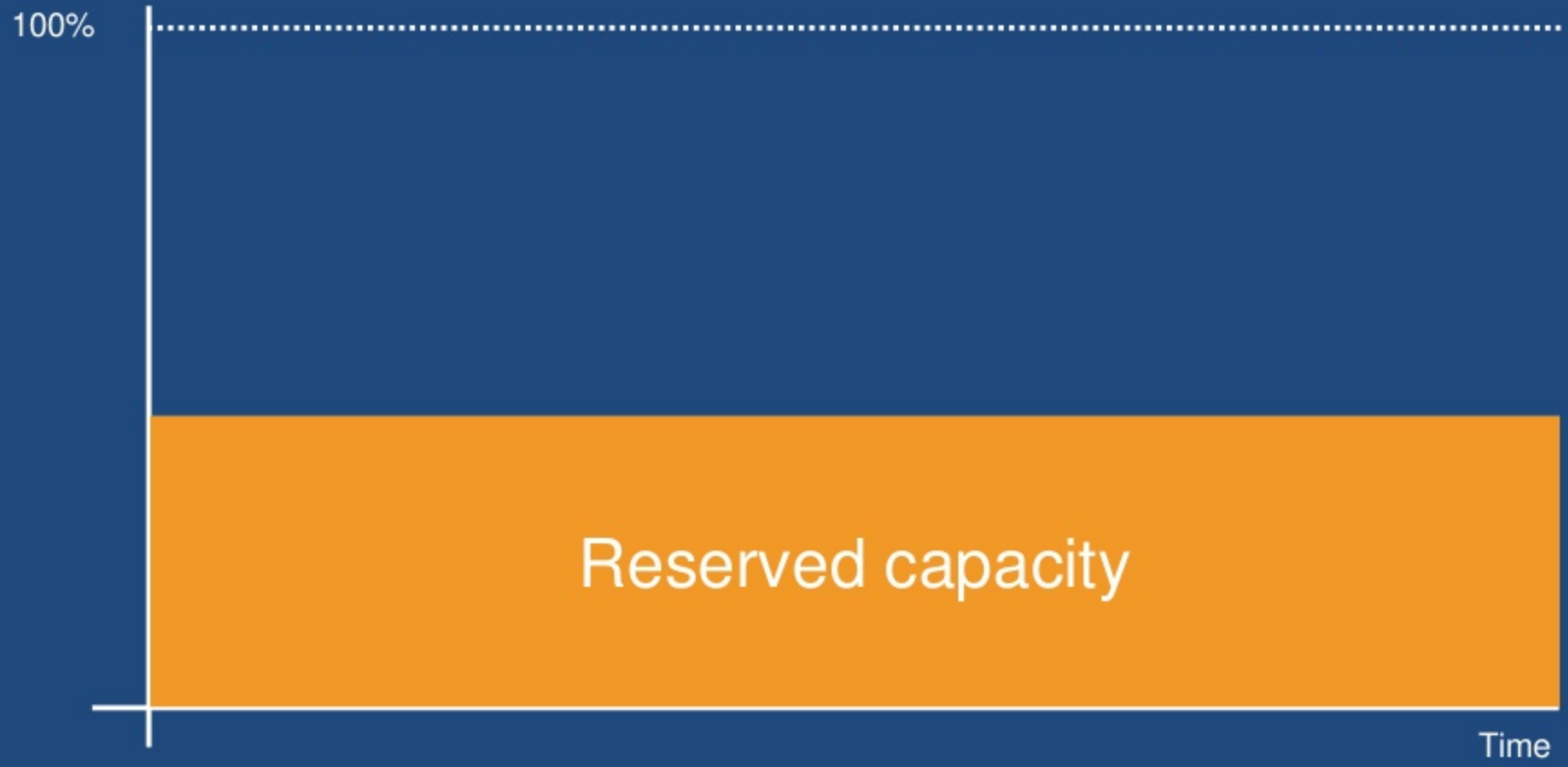
=

1,000 instances for 1 hour

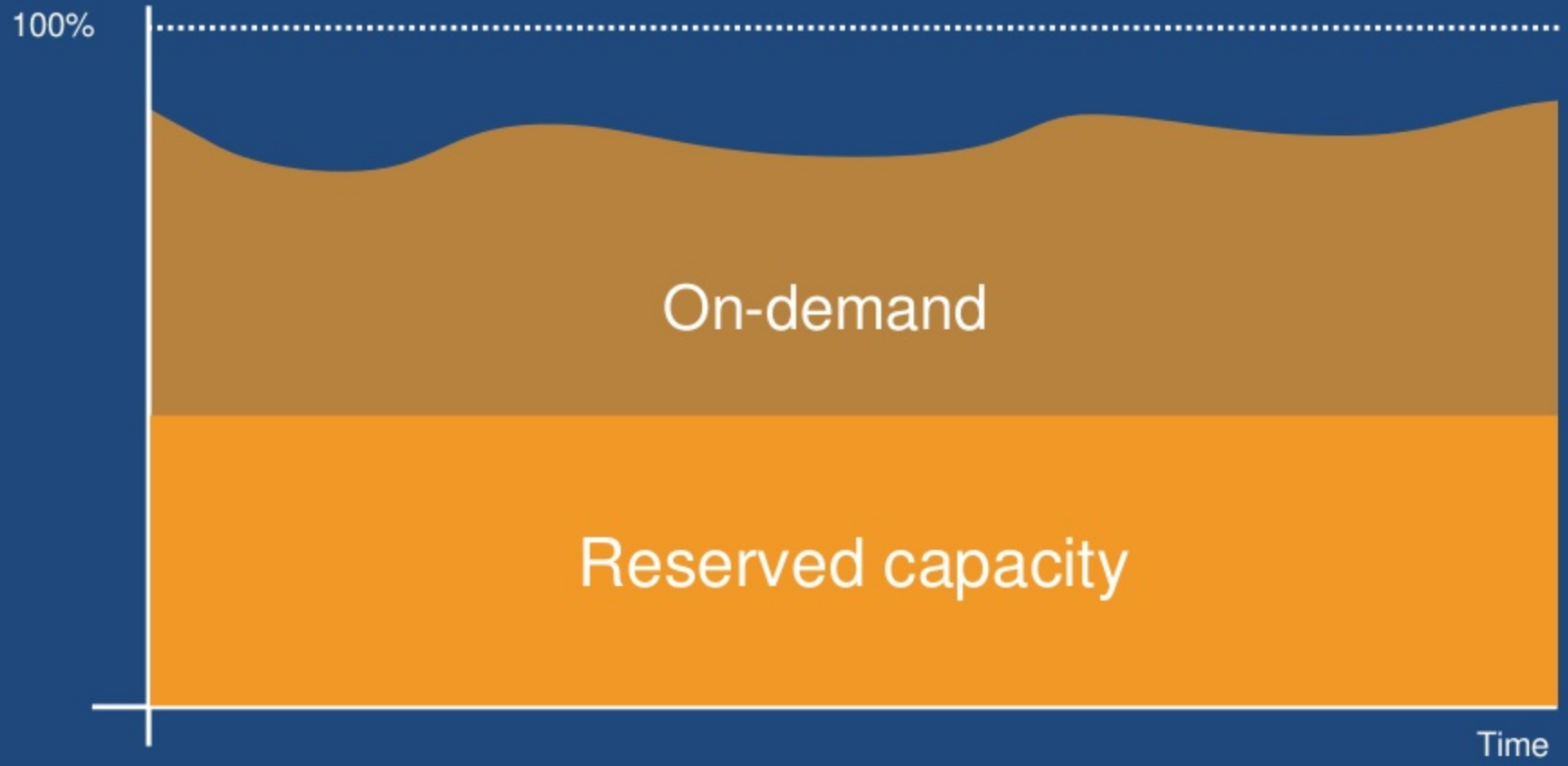
Small instance = $80

Achieving economies of scale

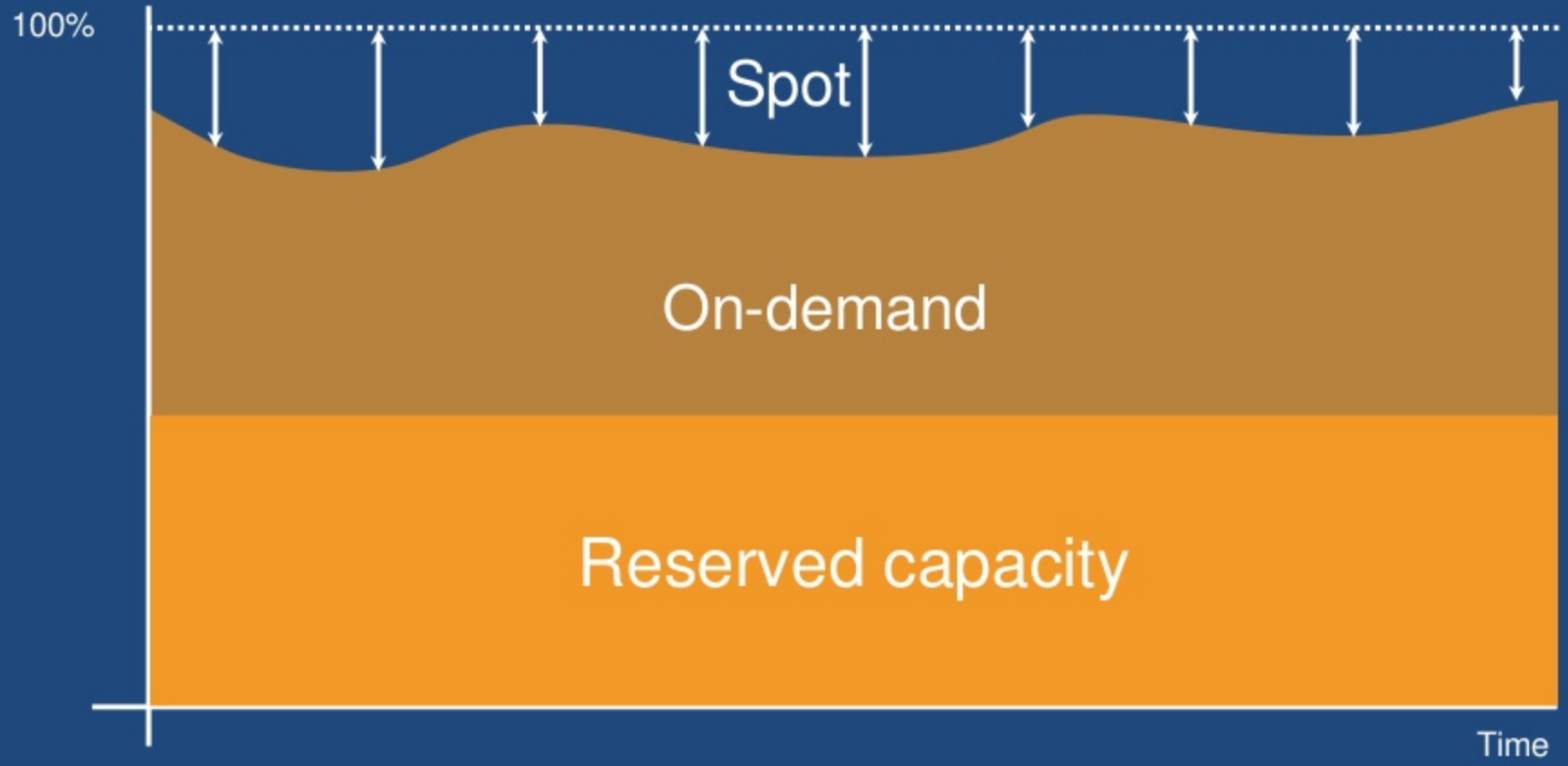
Achieving economies of scale



Achieving economies of scale



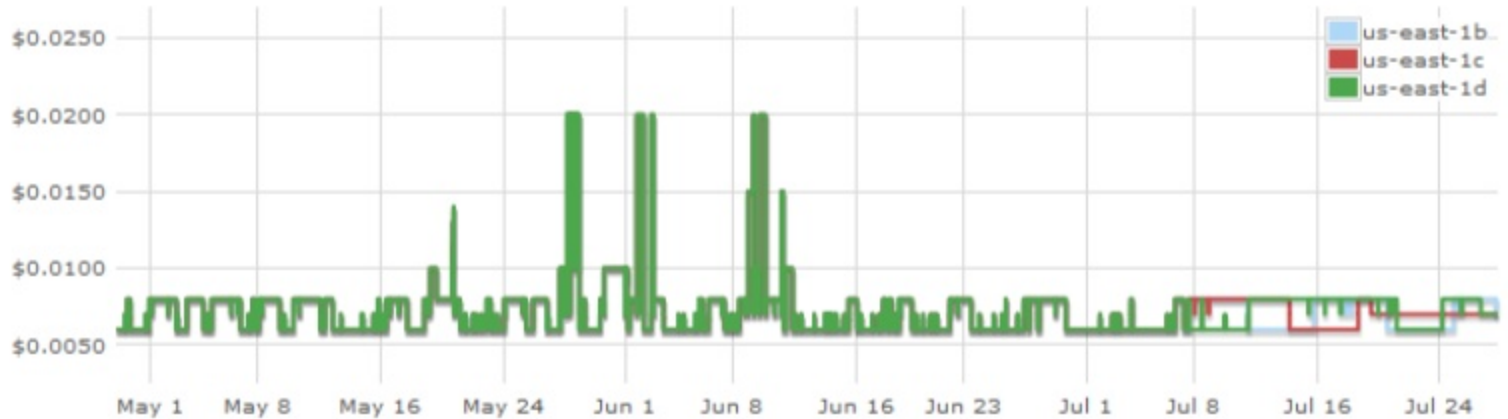
Achieving economies of scale



Spot Instance Pricing History

Cancel

Product: Linux/UNIX **Instance Type:** t1.micro **Date Range:** 3 months **Zone:** All zones



Close

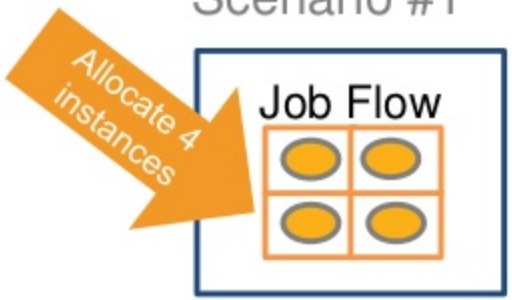
EMR with spot instances

#1: Cost without Spot

4 instances *14 hrs * $0.50 = $28

EMR with spot instances

Scenario #1

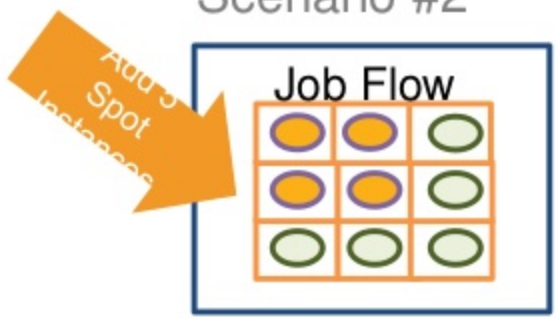


Duration:

#1: Cost without Spot

4 instances *14 hrs * $0.50 = $28

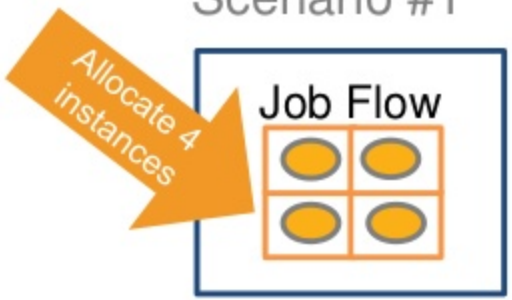
Scenario #2



Duration:

EMR with spot instances

Scenario #1

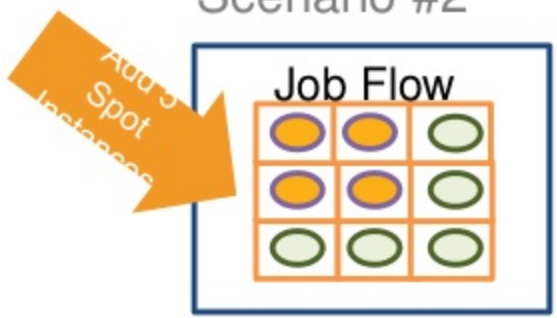


Duration:

#1: Cost without Spot

4 instances *14 hrs * $0.50 = $28

Scenario #2



Duration:

#2: Cost with Spot

4 instances *7 hrs * $0.50 = $14 +

5 instances * 7 hrs * $0.25 = $8.75

Total = $22.75

EMR with spot instances

#1: Cost without Spot

4 instances *14 hrs * $0.50 = $28

#2: Cost with Spot

4 instances *7 hrs * $0.50 = $14 +
5 instances * 7 hrs * $0.25 = $8.75
Total = $22.75

Some other things…

Hints and tips

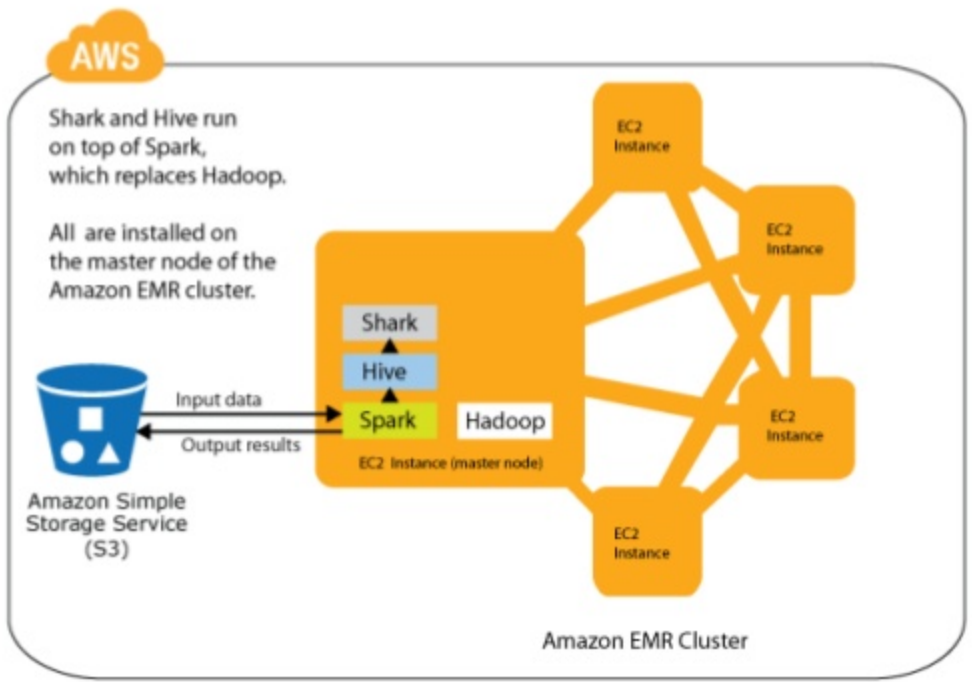
Managing big data

- Dynamic Creation of Clusters for Processing Jobs
- Bulk of Data Stored in S3
- Use Reduced Redundancy Storage for Lower Value / Re-creatable Data, and Lower Cost by 30%
- DynamoDB for High Performance Data Access
- Migrate data to Glacier for Archive and Periodic Access
- VPC & IAM for Network Security

Shark

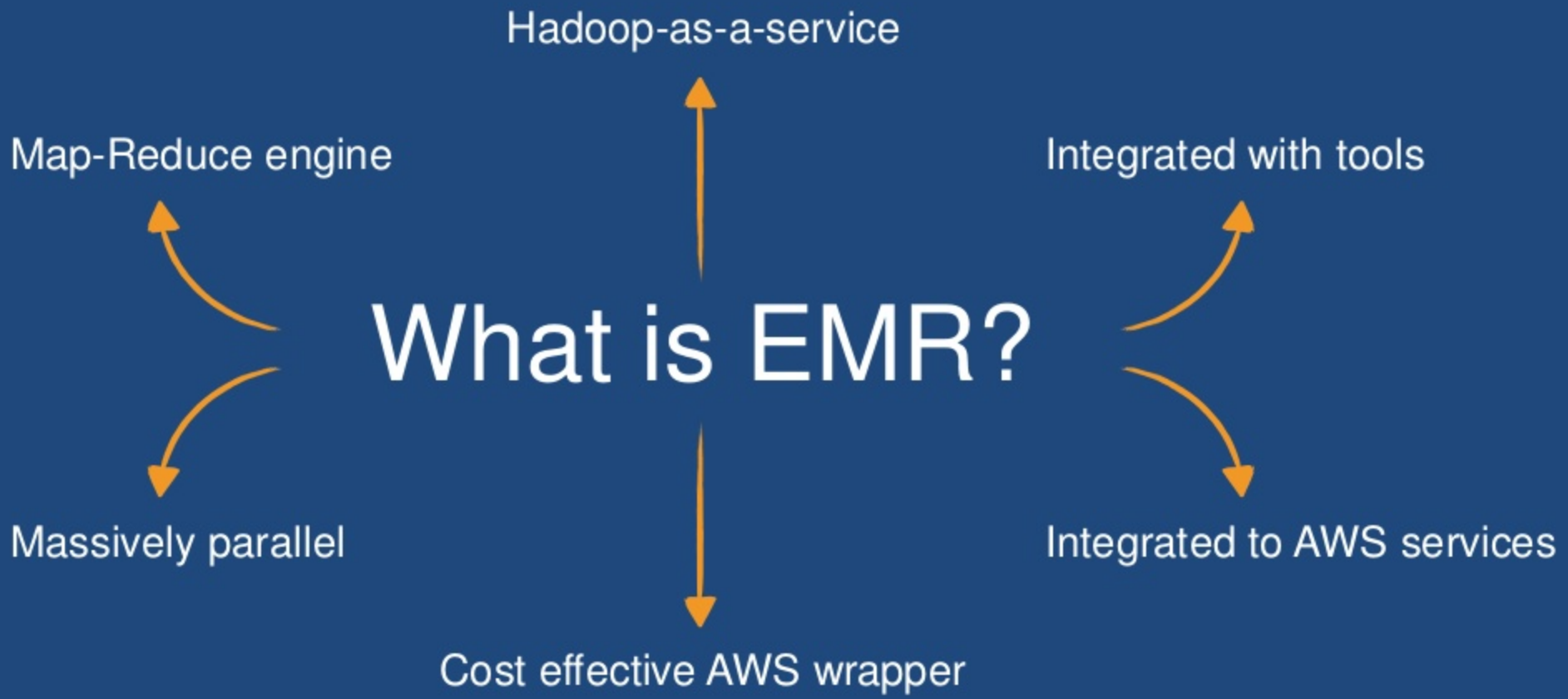
Shark can return results up to 30 times faster than the same queries run on Hive.

```
elastic-mapreduce --create --alive --name "Spark/Shark
Cluster”
--bootstrap-action
s3://elasticmapreduce/samples/spark/install-spark-
shark.sh
--bootstrap-name "install Mesos/Spark/Shark”
--hive-interactive --hive-versions 0.7.1.3
--ami-version 2.0
--instance-type m1.xlarge --instance-count 3
--key-pair ec2-keypair
```



aws.amazon.com/blogs/aws/

Summary


Hadoop-as-a-service
Map-Reduce engine
Integrated with tools
What is EMR?
Massively parallel
Integrated to AWS services
Cost effective AWS wrapper

Try the tutorial:

Contextual Advertising using Apache Hive and Amazon EMR

aws.amazon.com/articles/2855

Find out more:

aws.amazon.com/big-data

http://aws.amazon.com/elasticmapreduce/getting-started/

Follow us for more events & webinars

Ian Massingham – Technical Evangelist

@IanMmmm

@AWS_UKI for local AWS events & news

@AWScloud for Global AWS News and Announcements

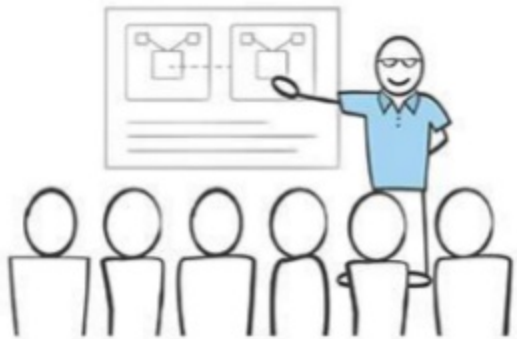
AWS Training & Certification

Self-Paced Labs

Try products, gain new skills, and get hands-on practice working with AWS technologies

aws.amazon.com/training/self-paced-labs

Training



Skill up and gain confidence to design, develop, deploy and manage your applications on AWS

aws.amazon.com/training

Certification

Demonstrate your skills, knowledge, and expertise with the AWS platform

aws.amazon.com/certification

amazon web services

We typically see customers start by trying our services

Get started now at : aws.amazon.com/getting-started

amazon web services

Design your application for the AWS Cloud

AWS Reference Architectures

The flexibility of AWS allows you to design your application architectures the way you like. AWS Reference Architecture Datasheets provide you with the architectural guidance you need in order to build an application that takes full advantage of the AWS cloud. Each datasheet includes a visual representation of the architecture and basic description of how each service is used.

Web Application Hosting
Build highly-scalable and reliable web or mobile-web applications (PDF)

Content and Media Serving
Build highly reliable systems that serve massive amounts of content and media (PDF)

Batch Processing
Build auto-scalable batch processing systems like video processing pipelines (PDF)

Fault tolerance and High Availability
Build systems that quickly failover to new instances in an event of failure (PDF)

Large Scale Processing and Huge Data sets
Build high-performance computing systems that involve Big Data (PDF)

Ad Serving
Build highly-scalable online ad serving solutions (PDF)

Disaster Recovery for Local Applications
Build cost-effective Disaster Recovery solutions for on-premises applications (PDF)

File Synchronization
Build simple file synchronization service (PDF)

More details on the AWS Architecture Center at : aws.amazon.com/architecture

amazon web services