



Amazon Elastic MapReduce (API Version 2009-03-31)

Revised: 4/25/2011

Quick Reference Card (page 1)

Create Job Flow

```
JAR $ ./elastic-mapreduce --create --alive --input AmazonS3bucket --output AmazonS3bucket --log-uri AmazonS3bucket
Stream $ ./elastic-mapreduce --create --alive --stream --input AmazonS3bucket --output AmazonS3bucket --log-uri AmazonS3bucket
Pig $ ./elastic-mapreduce --create --alive --name "Pig Test" --input AmazonS3bucket --output AmazonS3bucket --num-instances COUNT --instance-type TYPE --pig-interactive
Hive $ ./elastic-mapreduce --create --alive --name "Hive Test" --input AmazonS3bucket --output AmazonS3bucket --num-instances COUNT --instance-type TYPE --hive-interactive
```

Add a Job Flow Step

```
JAR $ ./elastic-mapreduce -j JobFlowId
Stream $ ./elastic-mapreduce -j JobFlowId --streaming
```

Terminate a Job Flow

```
$ ./elastic-mapreduce --jobflow JobFlowId --terminate
```

Get Information About a Job Flow

```
$ ./elastic-mapreduce --describe --jobflow JobFlowID
```

List Job Flows

```
$ ./elastic-mapreduce --list [--active] [--running] [--terminated]
```

SSH Into a Master Node

```
./elastic-mapreduce --ssh --jobflow JobFlowID
```

Use Additional Files and Libraries With the Mapper or Reducer

```
--cache s3n://bucket/path_to_executable#local_path
```

Adding Files to the Distributed Cache

```
Single file: --cache s3n://my_bucket/sample_dataset.dat#sample_dataset_cached.dat
Archive file: --cache-archive s3n://my_bucket/sample_dataset.tgz#sample_dataset_cached
```

Enable Output Data Compression Using the Console and a Streaming Job Flow

```
--jobconf mapred.output.compress=true
```

Hadoop File Locations

Failure logs: /mnt/var/log/hadoop/ on each node, or *JobFlowID/node/InstanceID/daemons/* on Amazon S3
UI for MapReduce job tracker(s): http://master_dns_name:9100/
UI for HDFS name node(s): http://master_dns_name:9101/
Temporary files: /mnt/var/lib/hadoop/tmp
Cache: /mnt/var/lib/hadoop/mapred/taskTracker/archive/

Credential File Fields

```
"access-id": "AccessKeyID",
"private-key": "PrivateKey",
"key-pair": "KeyPair",
"key-pair-file": "Location_of_key_pair_PEM_file",
"region": "us-east-1 | us-west-1 | eu-west-1 | ap-southeast-1 | ap-northeast-1",
"log-uri": "Amazon_S3_bucket_for_log_files"
```

Useful Links

Forum: <https://forums.aws.amazon.com/forum.jspa?forumID=52>
Resource Center: <http://aws.amazon.com/elasticmapreduce/>
Articles & Tutorials: <http://aws.amazon.com/articles/Elastic-MapReduce>
Release Notes: <http://aws.amazon.com/releasenotes/Elastic%20MapReduce>
Samples & Libraries: <http://aws.amazon.com/code/Elastic-MapReduce>
Developer Tools: <http://aws.amazon.com/developertools/Elastic-MapReduce>
Technical Documentation: <http://aws.amazon.com/documentation/elasticmapreduce/>
WSDL Location: <http://elasticmapreduce.amazonaws.com/doc/2009-03-31/ElasticMapReduce.wsdl>
CLI Download: <http://aws.amazon.com/developertools/2264>

Log File Locations

```
[log-uri]/JobFlowId/jobs/
[log-uri]/JobFlowId/node/
[log-uri]/JobFlowId/steps/
[log-uri]/JobFlowId/steps/stepNumber/syslog
[log-uri]/JobFlowId/steps/stepNumber/stdout
[log-uri]/JobFlowId/steps/stepNumber/controller
[log-uri]/JobFlowId/steps/stepNumber/stderr
[log-uri]/JobFlowId/task-attempts/
```

Command Line Options

--active List running, starting, or shutting down job flows
 --alive Create a job flow that stays running even though it has executed all of its steps
 --all List all job flows in the last 2 months
 --arg ARG Specify an argument to a JAR or a Streaming step
 --cache CACHE_FILE
 A file to load into the cache, e.g. s3://mybucket/sample.py#sample.py
 --create Create a new job flow
 --c CREDENTIALS_FILE
 File containing ACCESS_ID and SECRET_KEY
 --credentials
 -a, --access_id ACCESS_ID AWS Access ID
 -k, --secret_key SECRET_KEY AWS Secret Key
 --debug Print stack traces when exceptions occur
 --endpoint ENDPOINT
 Specify the web service endpoint
 -h, --help Show help message
 --hadoop-version VERSION
 Choose version of Hadoop, default 0.20
 --hive-versions VERSION, [VERSION]
 Choose version(s) of Hive, default 0.5
 --input INPUT Input to the steps, e.g. s3://mybucket/input
 --instance-type INSTANCE_TYPE
 The type of the instances to launch
 --JAR JAR Add a step that executes a JAR
 --jobconf JOB_CONF
 Specify jobconf arguments to pass to streaming
 -j, --jobflow JOB_FLOW_ID
 Job flow ID
 --key-pair KEYPAIR
 Location of key pair PEM file
 --list, --describe List all job flows created in the last 2 days
 --log-uri LOG_URI
 Location in Amazon S3 to store logs from the job
 flow, for example, s3://mybucket/logs
 --main-class MAIN_CLASS
 Specify main class for the JAR
 --mapper MAPPER
 The mapper program or class
 -n, --max-results MAX_RESULTS
 Maximum number of results to list
 --name NAME Name of the job flow
 --nosteps Do not list steps when listing jobs
 --num-instances NUM_INSTANCES
 Number of instances in the job flow
 --output OUTPUT
 The output to the steps, e.g. s3://mybucket/output
 --reduce REDUCER
 The reducer program or class
 --state STATE List job flows in STATE
 --step-name STEP_NAME
 Add a step to the work flow
 --step-action STEP_ACTION
 Action to take when step finishes
 --stream Add a step that performs Hadoop streaming
 --terminate Terminate the job flow
 -v, --verbose Turn on verbose logging of program interaction
 --version Print a version string

Predefined Bootstrap Actions

--bootstrap-action "s3://[mybucket]/[myfile1]" --args "[arg1]", "[arg2]"
 s3://elasticmapreduce/bootstrap-actions/configure-daemons
 s3://elasticmapreduce/bootstrap-actions/configure-hadoop
 s3://elasticmapreduce/bootstrap-actions/configurations/latest/memory-intensive
 s3://elasticmapreduce/bootstrap-actions/run-if

Hive Commands

hive [<-f filename>|<-e query-string>] [-S] [-hiveconf x=y]*
 [-d Var=Value]*
 -e 'query string' SQL from command line (interactive)
 -f filename SQL from file
 -d Var=Value Passes value into Hive script as \${Var}
 -S Silent mode in interactive shell where only data is emitted
 -hiveconf x=y Use this to set Hive or Hadoop configuration variables
 add FILE value value
 Adds a file to the list of resources
 ! command Execute a shell command from Hive shell
 dfs dfs command
 Execute dfs command from Hive shell
 list FILE List all the resources already added
 list FILE value Check given resources are already added or not
 query string Execute Hive query and send results to stdout
 Quit Exit interactive shell
 set key=value Set configuration variable
 Set List configuration variables overridden by user or Hive
 set -v List all Hadoop and Hive configuration variables

Pig Relational Operators

COGROUP alias BY field_alias [INNER | OUTER], alias BY field_alias [INNER | OUTER] [PARALLEL n];
 CROSS alias, alias [, alias ...] [PARALLEL n];
 DISTINCT alias [PARALLEL n];
 DUMP alias;
 FILTER alias BY expression;
 FOREACH { gen_blk | nested_gen_blk } [AS schema];
 GROUP alias { [ALL] | [BY {field_alias [, field_alias]] | * | [expression]] } [PARALLEL n];
 JOIN alias BY field_alias, alias BY field_alias [, alias BY field_alias ...] [USING "replicated"] [PARALLEL n];
 LIMIT alias n;
 LOAD 'data' [USING function] [AS schema];
 ORDER alias BY { * [ASC|DESC] | field_alias [ASC|DESC] [, field_alias [ASC|DESC] ...] } [PARALLEL n];
 SAMPLE alias size;
 SPLIT alias INTO alias IF expression, alias IF expression [, alias IF expression ...];
 STORE alias INTO 'directory' [USING function];
 STREAM alias [, alias ...] THROUGH { 'command' | cmd_alias } [AS schema];
 UNION alias, alias [, alias ...];

CLI Configuration

:endpoint => "https://elasticmapreduce.amazonaws.com",
 :ca_file => File.join(File.dirname(__FILE__), "cacert.pem"),
 :aws_access_key => my_access_id,
 :aws_secret_key => my_secret_key,
 :signature_algorithm => :V2

Resizing Running Job Flows

--modify-instance-group INSTANCE_GROUP_ID
 Modify an existing instance group
 --add-instance-group ROLE
 Add an instance group to an existing job flow
 --instance-count INSTANCE_COUNT
 Set the instance count of an instance group
 --instance-type INSTANCE_TYPE
 Set the instance type of an instance group
 --set-num-instances COUNT
 Change the number of nodes of an instance group