

---

# **Amazon Elastic MapReduce**

## **Getting Started Guide**

**API Version 2009-03-31**



# **Amazon Elastic MapReduce: Getting Started Guide**

Copyright © 2009 Amazon Web Services LLC or its affiliates. All rights reserved.

## Table of Contents

Welcome .....	1
What's New .....	4
Introduction to Elastic MapReduce .....	5
Getting Set Up .....	8
Signing Up for Amazon S3 .....	10
Amazon S3 Bucket Creation .....	11
Using the Elastic MapReduce Console .....	13
Additional Console Tutorials .....	19
Where Do I Go from Here? .....	21
Document Conventions .....	24

# Welcome

---

## Topics

- [Who Should Read This Guide \(p. 1\)](#)
- [How to Give Us Feedback \(p. 2\)](#)
- [How to Use this Guide \(p. 2\)](#)
- [Amazon Elastic MapReduce Resources \(p. 2\)](#)

This is the *Amazon Elastic MapReduce Getting Started Guide*. This section describes who should read this guide, how the guide is organized, and other resources related to Amazon Elastic MapReduce.

This guide describes how to use the Amazon Elastic MapReduce console to perform Hadoop processing on sample data.

Amazon Elastic MapReduce, Amazon Elastic Compute Cloud, and Amazon Simple Storage Service are sometimes referred to within this guide as "Elastic MapReduce," "EC2," and "Amazon S3," respectively. All copyrights and legal protections still apply.

For a description of what's new in this release of the Amazon Elastic MapReduce service, see [What's New \(p. 4\)](#).

## Who Should Read This Guide

This guide is for developers and for the community of researchers and data analysts that need to process vast amounts of data efficiently and cost-effectively. Because this guide uses the Amazon Elastic MapReduce console, you do not need to be a programmer to use this guide.

## Required Knowledge and Skills

Use of this guide assumes you have a basic understanding of Hadoop. For an overview, go to <http://hadoop.apache.org/core/>. We also assume that you have a very broad understanding of what Amazon S3 and EC2 do.

You should be familiar with the basics of:

- Hadoop (go to <http://hadoop.apache.org/core/>)
- Amazon S3 and EC2 (go to [Amazon Simple Storage Service Developer Guide](#) and [Amazon Elastic Compute Cloud Developer Guide](#), respectively)

- Web services (go to [W3 Schools Web Services Tutorial](#))

## How to Give Us Feedback

The online version of this guide provides a link at the top of each page that enables you to enter feedback about this guide. We strive to make our guides as complete, error free, and easy to read as possible. You can help by giving us feedback. Thank you in advance!



## How to Use this Guide

This guide is organized as a high-level introduction and tutorial. It is divided into several major sections that enable you to practice using Elastic MapReduce. Each section builds on the previous sections, so if you read and work through the examples in sequence, you'll get a basic understanding of Elastic MapReduce.

Depending on your AWS programming experience, you can use this guide in the following ways.

### **If you are new to AWS**

Read the entire guide. It takes you step-by-step through the process of using Elastic MapReduce to run a Hadoop job flow.

### **If you have developed applications for other AWS services and know Amazon S3 basics**

You can skip most of the *Getting Set Up* section. Just sign up to use Elastic MapReduce and move on to the tutorial or implementation sections. For more information about signing up for Elastic MapReduce, see [How to Get an Elastic MapReduce Developer Account](#) (p. 8).

### **If you want to use the Elastic MapReduce API and not the Console**

You can skip to the implementation section that lists many different ways to get in-depth knowledge of Elastic MapReduce that go beyond the scope of this guide. To skip to more advanced topics, see [Where Do I Go from Here?](#) (p. 21).

Each section of this guide builds on those that precede it. For that reason, you should read the sections in the order they are presented.

## Amazon Elastic MapReduce Resources

The following table lists related resources that you'll find useful as you work with this service.

Resource	Description
<a href="#">Amazon Elastic MapReduce Developer Guide</a>	Shows how to implement Query requests and handle responses in various computing languages
<a href="#">Amazon Elastic MapReduce Technical FAQ</a>	The FAQ covers the top 20 questions developers have asked about this product.

**Amazon Elastic MapReduce Getting Started Guide**  
**Amazon Elastic MapReduce Resources**

---

Resource	Description
<a href="#">Release notes</a>	The release notes give a high-level overview of the current release. They specifically note any new features, corrections, and known issues.
<a href="#">AWS Developer Resource Center</a>	A central starting point to find documentation, code samples, release notes, and other information to help you build innovative applications with AWS.
<a href="#">AWS Management Console</a>	Location of the Amazon Elastic MapReduce console.
<a href="#">Discussion Forums</a>	A community-based forum for developers to discuss technical questions related to Amazon Web Services.
<a href="#">AWS Support Center</a>	The home page for AWS Technical Support, including access to our Developer Forums, Technical FAQs, Service Status page, and Premium Support .
<a href="#">Amazon Elastic MapReduce product information</a>	The primary web page for information about Amazon Elastic MapReduce.
<a href="#">Contact Us</a>	A central contact point for inquiries concerning AWS billing, account, events, abuse etc.
<a href="#">Conditions of Use</a>	Detailed information about the copyright and trademark usage at Amazon.com and other topics.

## What's New

---

This What's New is associated with the 2009-03-31 version of the Amazon Elastic MapReduce web service. This guide was last updated on 2009-12-02.

Change	Description	Release Date
Support for Hive	Hive is an open source library that runs on top of Hadoop. The library takes SQL-like commands in a language called Hive QL and converts those commands into MapReduce jobs. Using Hive saves you the trouble of programming MapReduce applications in lower-level computing languages, such as Java.	October 1, 2009
Support for Pig	Pig is an open source, Apache library that runs on top of Hadoop. The library takes SQL-like commands in a language called Pig Latin and converts those commands into a graph of MapReduce jobs, which is executed on data in an Amazon EC2 cluster. So now, in addition to running MapReduce jobs using a JAR or stream you can use Pig in programmatic or interactive modes.	August 10, 2009
New guide	This is the first publication of this guide.	April 2, 2009

# Introduction to Elastic MapReduce

---

## Topics

- [Overview of Elastic MapReduce \(p. 5\)](#)
- [Elastic MapReduce Concepts \(p. 6\)](#)
- [Overview of Examples \(p. 7\)](#)

This introduction to Elastic MapReduce provides a high-level overview of this web service. After reading this section, you should understand the basics you need in order to work through the examples in this guide.

## Overview of Elastic MapReduce

Elastic MapReduce is a web service that makes it easy to process vast amounts of data using Amazon Simple Storage Service (Amazon S3), which is where you store the data, and a cluster of Amazon Elastic Compute Cloud (EC2) instances, which process the data. Elastic MapReduce uses Hadoop processing to do such things as web indexing, data mining, log file analysis, machine learning, scientific simulation, and bioinformatics research.

## Features

The following list describes the features of Elastic MapReduce highlighted by the tutorial in this guide.

- **Hadoop processing**—Elastic MapReduce automatically configures, starts up, and shuts down a cluster of EC2 instances to process a Hadoop work flow.
- **Job flow with multiple steps**—A single job flow can have one or more steps. Each step applies an algorithm to the results of the previous step.
- **View job flow details**—You can monitor the status of a job flow and get extended information about its execution.



Elastic MapReduce provides an API, command line tool, and Console to make it easier to build solutions leveraging Elastic MapReduce.

## Elastic MapReduce Concepts

### Topics

- [Amazon Elastic MapReduce and Hadoop](#) (p. 6)
- [Master and Slave Nodes](#) (p. 6)

This section describes key Elastic MapReduce concepts.

## Amazon Elastic MapReduce and Hadoop

Elastic MapReduce uses Apache Hadoop, which is an open source Java software framework that supports massive data processing across a cluster of servers. Hadoop uses a programming model called *MapReduce* that divides a large dataset into many small fragments. Hadoop distributes a data fragment and a copy of the MapReduce executable to each of the slave nodes in an EC2 cluster. All slave nodes run the MapReduce executable on their subset of the data. Hadoop then combines the results from all of the slave nodes into a finished output. Elastic MapReduce uploads that output into the Amazon S3 bucket you designate.

Typically, the processing involves performing relatively simple operations on very large amounts of data, for example, adding a watermark to 1,000,000 digital images. It is also typical to perform more than one process on the data. Each process is called a *step* and the sequence of steps is called a job flow.

For more information about Hadoop, go to <http://hadoop.apache.org/core/>.

## Job Flow and Steps

A job flow is a user-defined task that Elastic MapReduce performs. A job flow consists of one or more steps each of which must complete in sequence successfully for the job flow to finish successfully. A step maps roughly to one algorithm that manipulates the data. A job flow typically consists of multiple steps. The output of one step becomes the input of the next.

## Hadoop and Amazon Elastic MapReduce

Hadoop performs the MapReduce functions of dividing up the work among the slave instances in the cluster, tracking status, and combining the individual results into one output.

Elastic MapReduce takes care of provisioning an EC2 cluster, terminating it, moving the data between it and Amazon S3, and optimizing Hadoop. Elastic MapReduce removes most of the cumbersome details of setting up the hardware and networking required by the server cluster, such as monitoring that setup, configuring Hadoop, and executing the job flow. Together, Elastic MapReduce and Hadoop provide all of the power of Hadoop processing with the ease, low cost, scalability, and power afforded by Amazon S3 and EC2.

## Master and Slave Nodes

An EC2 cluster contains one master node and one or more slave nodes, as depicted in the following figure.



The master node helps coordinate the distribution of the MapReduce executable and subsets of the entire dataset to the slaves and tracks the status of each task being performed by the slaves. Hadoop monitors the health of all of the instances. If Hadoop determines that an instance is not performing, Elastic MapReduce terminates the instance and restarts another.

Each slave node has a copy of the MapReduce executable and receives from Amazon S3 a portion of the entire data set. Each slave node processes the data, uploads it back to Amazon S3, and provides status metadata to the master node.

## Overview of Examples

This guide is a tutorial that shows how to use the Elastic MapReduce console to create a job flow and view extensive details about it

The next section explains how to sign up for Elastic MapReduce, and use third party tools to create and load Amazon S3 buckets.

If you are familiar with Amazon S3 and tools, such as S3 Firefox Organizer, that enable you to create buckets and load data into them, you can skip to just signing up to use Elastic MapReduce and move on to the tutorial or implementation section. For more information about signing up for Elastic MapReduce, see [Signing Up for Elastic MapReduce \(p. 10\)](#).

If you are interested in using the Elastic MapReduce API and not the console, you can skip to the implementation section that lists many different ways to get an in-depth knowledge of Elastic MapReduce. To skip to more advanced topics, see [Implementing Elastic MapReduce in a Live Environment \(p. 21\)](#)

# Getting Set Up

---

## Topics

- [How to Get an Elastic MapReduce Developer Account \(p. 8\)](#)
- [Viewing Your AWS Security Credentials \(p. 9\)](#)
- [Signing Up for Amazon S3 \(p. 10\)](#)
- [Amazon S3 Bucket Creation \(p. 11\)](#)

This section discusses the tasks you need to perform before using Elastic MapReduce.

## How to Get an Elastic MapReduce Developer Account

This section explains how to sign up for an Elastic MapReduce developer account. This procedure automatically registers you to use Amazon S3 and Amazon EC2 unless you have already registered yourself for those products.

### To sign up for an Elastic MapReduce developer account

1. Go to <http://aws.amazon.com/elasticmapreduce/> and click **Sign Up for Elastic MapReduce**.
2. If asked, log in with your AWS account login and password.

If you do not already have an Amazon account, the instructions prompt you to enter account login names and passwords.

3. Review the information on Sign Up For Amazon Elastic MapReduce and, if you accept the terms and conditions, click **Complete Sign Up** and follow the instructions on the subsequent pages.

This process also assigns you an AWS account, which is similar to an Amazon.com customer account except that the AWS account gives you access to Amazon Web Services.

# Viewing Your AWS Security Credentials

AWS uses special identifiers to help protect your data. In this section, we show you how to view your identifiers so you can use them.



## Tip

If you already know how to view your AWS security credentials, skip to the next section. For more information, see [Signing Up for Amazon S3 \(p. 10\)](#).

AWS assigns you the following credentials when you create your AWS account:

- Access Key ID (a 20-character, alphanumeric sequence, for example: 022QF06E7MXBSH9DHM02)  
You include your Access Key ID in all AWS service requests to identify yourself as the sender of the request.
- Secret Access Key (a 40-character sequence, for example: kWcrIUX5JEDGM/LtmEENI/aVmYvHNif5zB+d9+ct)



## Caution

Your Secret Access Key is a shared secret between you and AWS. Keep this ID secret; we use it to bill you for the AWS services you use. Never include the ID in your requests to AWS and never e-mail the ID to anyone even if an inquiry appears to originate from AWS or Amazon.com. No one who legitimately represents Amazon will ever ask you for your Secret Access Key.

The Access Key ID is not a secret, and anyone could use your Access Key ID in requests to AWS. To provide proof that you truly are the sender of the request, you also include a digital signature calculated using your Secret Access Key. The sample code handles this for you.

Your Access Key ID and Secret Access Key display when you create your AWS account. They are not e-mailed to you. If you need to see them again, you can view them at any time from your AWS account.

## To view your AWS security credentials

1. Go to <http://aws.amazon.com/security-credentials>.  
If you're not logged in, you are asked to. If you are logged in, the **Security Credentials** page displays.

## Security Credentials


AWS provides a number of ways for you to securely access your account and begin using our services. Below, you will find the list of credentials you need to identify yourself as a valid user of your account, as well as additional security options that enable you to further protect your account, manage your credentials, and control access.


This page contains the following information. Click to jump down:

- ↓ Access Credentials
- ↓ Your AWS Account ID, E-mail Address and Password
- ↓ Amazon Web Services Multi-Factor Authentication

## Access Credentials

In order to start using Amazon Web Services you must first identify yourself as the sender of a request to the given service. This is accomplished by sending a digital signature that is derived from a pair of public/private access keys or a valid security certificate.

 Access Keys

 X.509 Certificates

**Access Key ID**  
Your Access Key ID identifies you as the party responsible for service requests. Use your Access Key ID as the value of the `AWSAccessKeyId` parameter in requests you send to Amazon Web Services (when required).

**Secret Access Key**  
Each Access Key ID has a Secret Access Key associated with it. Use your Secret Access Key to calculate a signature to include in requests to web services that require authenticated requests. Your Secret Access Key is a secret, and should be known only by you and AWS. You should never include your Secret Access Key in your requests to AWS. You should never e-mail your Secret Access Key to anyone. It is important to keep your Secret Access Key confidential to protect your account.

2. Click **Access Key ID** or **Secret Access Key** to display your IDs.

# Signing Up for Amazon S3

In this guide, you will upload data to Amazon S3. Before you can use Amazon S3, you must sign up to use the service.



### Tip

If you already have an Amazon S3 account and are familiar with bucket creation and permissions, you can skip to the section [Using the Elastic MapReduce Console \(p. 13\)](#)

## To sign up for Amazon S3

1. Go to <http://aws.amazon.com/s3> and click **Sign Up for Amazon S3**.
2. Take one of the following actions depending on whether or not you associated a credit card with your Amazon.com account.

If you already have a credit card associated with the account:	If you don't have a credit card associated with the account:
<ul style="list-style-type: none"><li>• Review the displayed pricing and credit card information and click <b>Complete Sign Up</b>.</li></ul>	<ol style="list-style-type: none"><li>1. Enter information for a valid credit card and click <b>Continue</b>.</li><li>2. Enter the billing address to use with the credit card and click <b>Continue</b>.</li><li>3. Review the displayed pricing and credit card information you provided and click <b>Complete Sign Up</b>.</li></ol>

AWS sends you a confirmation e-mail.

## Amazon S3 Bucket Creation

Before you can use the examples in this guide you must create a bucket with write permissions where Elastic MapReduce can upload results. The following tools make creating a bucket, putting data in the bucket, and retrieving data from the bucket easy:

The following tools make creating a bucket and retrieving data from it easy:

- S3curl  
For more information, go to <http://developer.amazonwebservices.com/connect/entry.jspa?externalID=128>.
- S3 Firefox Organizer  
For more information, go to <https://addons.mozilla.org/en-US/firefox/addon/3247>.
- S3 tool  
For more information, go to <http://developer.amazonwebservices.com/connect/entry.jspa?externalID=739>.
- Bucket Explorer  
For more information, go to <http://www.bucketexplorer.com/>.
- CloudBerry  
For more information, go to <http://cloudberrylab.com/?page=cloudberry-explorer-amazon-s3>.

For more information on creating an Amazon S3 bucket, go to the [Amazon Simple Storage Service Getting Started Guide](#). Consult each tool's documentation for more information.



### Tip

Make a note of your bucket name so that you can use it in the sample requests.

## Naming Conventions for Amazon S3 Buckets

The bucket you create must be DNS friendly. We recommend that you use bucket names that conform with the following DNS requirements:

- Bucket names should not contain underscores (\_)
- Bucket names should be between 3 and 63 characters long
- Bucket names should not end with a dash
- Bucket names cannot contain dashes next to periods (e.g. "my-.bucket.com" and "my.-bucket" are invalid)

For more details, go to <http://docs.amazonwebservices.com/AmazonS3/latest/index.html?BucketRestrictions.html>.



### Tip

Make a note of your bucket name so that you can use it in this guide.

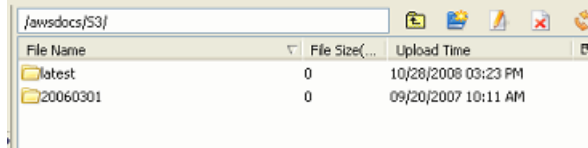
## How to Assign Bucket Permissions

After creating a bucket, make sure to set appropriate permissions on it. Typically, you give the owner read and write access and authenticated users read access.

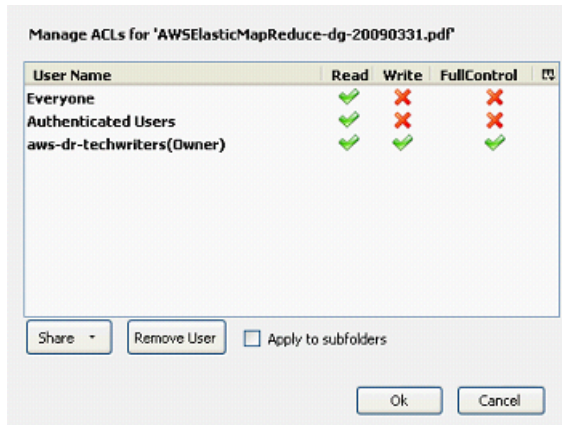
How you give permission depends on the tool. Here we show you using Amazon S3 FireFox Organizer.

### To set permissions on a bucket using S3Fox

1. Open S3Fox and navigate to the bucket you created.



2. Right-click on your bucket, select the permissions you want to give to each user type, and click **OK**.



You've now created your bucket and assigned it permissions. You will use this bucket in the following tutorial as the place where you will upload your processed data.

The next section is a tutorial that takes you step-by-step through using the Elastic MapReduce console to create and view a job flow. The tutorial is written procedurally so you should follow it from beginning to end. After completing the tutorial, you should have a good feel for some of the major tasks you can complete using Elastic MapReduce.

Alternately, you can skip the console tutorial and jump right to the last section that provides links to code samples, application examples, forums, and other resources designed to help you learn Elastic MapReduce. To skip to more advanced topics, see [Implementing Elastic MapReduce in a Live Environment](#) (p. 21)

# Using the Elastic MapReduce Console

---

## Topics

- [Sample Data and Mapper \(p. 13\)](#)
- [How to Create a Job Flow \(p. 14\)](#)
- [How to View Details of the Job Flow \(p. 19\)](#)
- [Additional Console Tutorials \(p. 19\)](#)
- [Send Us Your Feedback Now \(p. 20\)](#)
- [What's Next? \(p. 20\)](#)

The AWS Management Console provides an easy way to perform Hadoop processing on data sets. This section is a tutorial that walks you step-by-step through using the console to create and then view a job flow.

This tutorial uses a JAR to implement a MapReduce program. You can also use Hive and Pig, which use a higher-level, SQL like language. For more information, go to:

- Running Hive on Amazon ElasticMap Reduce—<http://developer.amazonwebservices.com/connect/entry.jspa?externalID=2857>
- Pig Video Tutorial—<http://s3.amazonaws.com/awsVideos/AmazonElasticMapReduce/ElasticMapReduce-PigTutorial.html>

## Sample Data and Mapper

So that you can actually work through the tutorial in this section, we created a sample mapper executable and loaded it along with sample data into the following Amazon S3 buckets:

- **Input data**—`elasticmapreduce/samples/wordcount/input`
- **Mapper**—`elasticmapreduce/samples/wordcount/wordSplitter.py`
- **Reducer**—`aggregate`





### Note

The Aggregate library that comes with Hadoop includes the reducer, `aggregate`, which provides many basic reducer aggregations, such as sum, max, and min. For more information, go to [Working with the Hadoop Aggregate Package](#).

You need the URI of the bucket you created in [Creating an Amazon S3 Bucket \(p. 11\)](#) that will hold the processed data. The format of the output URI is `http://[BucketYouCreated]/demo/output`.

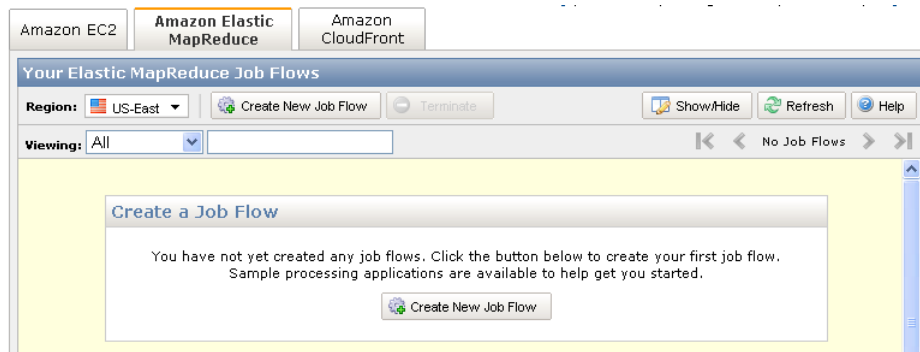
## How to Create a Job Flow

This section explains how to create a job flow using the Elastic MapReduce console.

### To create a job flow

1. Go to <https://console.aws.amazon.com/elasticmapreduce/home>.

The Elastic MapReduce console appears.



2. Click **Create a New Job Flow**.

The **Create a New Job Flow** page appears.

## Amazon Elastic MapReduce Getting Started Guide

### How to Create a Job Flow

**Create a New Job Flow** Cancel

**Define Job Flow** **Specify Parameters** **Configure EC2 Instances** **Review**

Creating a job flow to process your data using Amazon Elastic MapReduce is simple and quick. Let's begin by giving your job flow a name and selecting its type. If you don't already have an application you'd like to run on Amazon Elastic MapReduce, samples are available to help you get started.

**Job Flow Name\*:**

The name can be anything you like and doesn't need to be unique. It's a good idea to name the job flow something descriptive.

**Type\*:** ☒ **Streaming**  
A Streaming job flow allows you to write single-step mapper and reducer functions in a language other than java.

☐ **Custom Jar (advanced)**  
A custom jar on the other hand gives you more complete control over the function of Hadoop but must be a compiled java program. Amazon Elastic MapReduce supports custom jars developed for Hadoop 0.18.3.

☐ **Pig Program**  
Pig is a SQL-like language built on top of Hadoop. This option allows you to define a job flow that runs a Pig script, or set up a job flow that can be used interactively via SSH to run Pig commands.

☐ **Sample Applications**  
Select a sample application and click Continue. Subsequent forms will be filled with the necessary data to create

Word count is a Python application that counts occurrences of each word in provided documents. [Learn more and view license](#)

Continue \* Required field

3. Enter a name in the **Job Flow Name** field.  
We recommend that you use a descriptive name. It does not need to be unique.
4. Select **Sample Applications**, **Word Count**, and click **Continue**.  
The **Specify Parameters** page appears.

### Create a New Job Flow

✓ DEFINE JOB FLOW
○ SPECIFY PARAMETERS
○ CONFIGURE EC2 INSTANCES
○ REVIEW

Specify Mapper and Reducer functions to run within the Job Flow. The mapper and reducers may be either (i) class names referring to a mapper or reducer class in Hadoop or (ii) locations in Amazon S3. ([Click Here](#) for a list of available tools to help you upload and download files from Amazon S3.) The format for specifying a location in Amazon S3 is bucket\_name/path\_name. The location should point to an executable program, for example a python program. Extra arguments are passed to the Hadoop streaming program and can specify such as additional files to be loaded into the distributed cache.

**Input Location\*:**

The URL of the Amazon S3 Bucket that contains the input files.

**Output Location\*:**

The URL of the Amazon S3 Bucket to store output files. Should be unique.

**Mapper\*:**

The name of the mapper executable located in the Input Location.

**Reducer\*:**

The name of the reducer executable located in the Input Location.

**Extra Args:**

< Back
Continue 
\* Required

5. Use the following table to enter values in the fields and then click **Continue**.

In This Box...	Do this...
Input Location	Enter elasticmapreduce/samples/wordcount/input
Output Location	Enter the location of the Amazon S3 bucket you created in the setup. The value must be in the form: <span style="color: red;">[bucketName]</span> / <span style="color: red;">[path]</span>
Mapper	Enter elasticmapreduce/samples/wordcount/wordSplitter.py
Reducer	Enter aggregate
Args	(Enter nothing.)

The Configure EC2 Instances page appears.

## Amazon Elastic MapReduce Getting Started Guide

### How to Create a Job Flow

**Create a New Job Flow** Cancel

DEFINE JOB FLOW

SPECIFY PARAMETERS

CONFIGURE EC2 INSTANCES

REVIEW

Enter the number and type of EC2 instances you'd like to run your job flow on.

**Number of Instances\*:**

The number of EC2 instances to run in your Hadoop cluster.  
If you wish to run more than 20 instances, please complete the [limit request form](#).

**Type of Instance\*:**

The type of EC2 instances to run in your Hadoop cluster ([learn more about instance types](#)).

---

[Hide Advanced Options](#)

**Amazon S3 Log Path:**

The log path is a location in Amazon S3 where Elastic MapReduce will upload the log files for each step in the job flow. It will take a few minutes after the step has completed for the logs to appear. If you do not specify a path, the log files will not be uploaded.

**Amazon EC2 Key Pair:**

The Key Pair is the name of an Amazon EC2 Private Key that you have previously created when using Amazon EC2. It is a handle you can use to SSH into the master node of the Amazon EC2 cluster (without a password).

< Back Continue > \* Required field

6. Use the following table to enter values in the fields and then click **Continue**.

In this box...	Do this...
Number of Instances	Enter 1
Type of Instance	Select Small (m1-small)
Amazon S3 Log Path	(Enter nothing.)
EC2 Key Pair	(Enter nothing.)

The **Review** page appears.

### Create a New Job Flow

DEFINE JOB FLOW
SPECIFY PARAMETERS
CONFIGURE EC2 INSTANCES
REVIEW

Please review the details of your job flow and click "Create Job Flow" when you are ready to launch your Hadoop Cluster.

**Job Flow Name:** My Job Flow

**Type:** [Edit Job Flow De](#)

---

**Input Location:** s3n://kensbucket/hadoop/input

**Output Location:** s3n://kensbucket/hadoop/output

**Mapper:** s3n://kensbucket/kens-mapper.py

**Reducer:** s3n://kensbucket/kens-reducer.py

**Extra Args:**

[Edit Job Flow Para](#)

---

**Number of Instances:** 3

**Type of Instance:** m1.small

**Amazon S3 Log Path:**

**Amazon EC2 Key Pair:**

[Edit EC2 Configs and Advanced](#)

---

[< Back](#)

Create Job Flow
▶

**Note:** Once you click "Create Job Flow", your EC2 instances will be launched and you will be charged accordingly.

7. Review the information.

If this is true...	Do this...
One or more entries are incorrect	Click <b>Back</b> one or more times to revise any of the values
All entries are correct	Click <b>Create Job Flow</b>

When the job creation succeeds, you get the following

Create a New Job Flow
Cancel ✕

✔ Your Job Flow has been created.

Note: Your job flow may take a few minutes to launch, depending on the type of processing job you are running.

[View my job flows and check on job flow status](#)

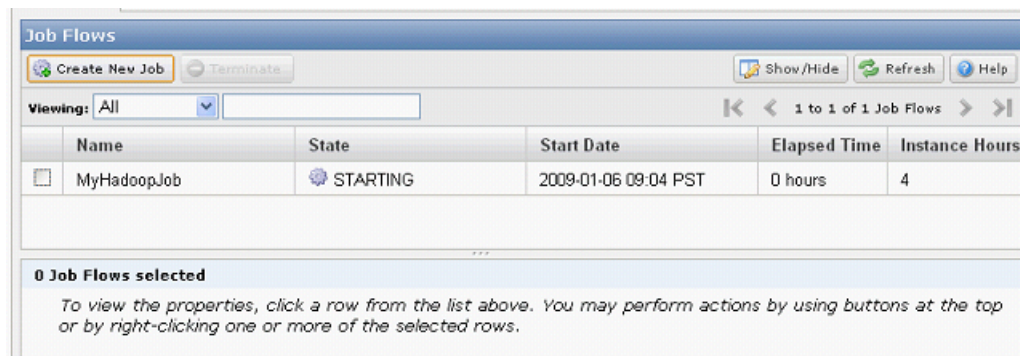
Close
▶

\* Required field

confirmation.

8. Click **Close**.

The console reflects the new job flow. The maximum lifetime of a job flow is 2 weeks. Elastic MapReduce returns an error if processing doesn't complete within two weeks.

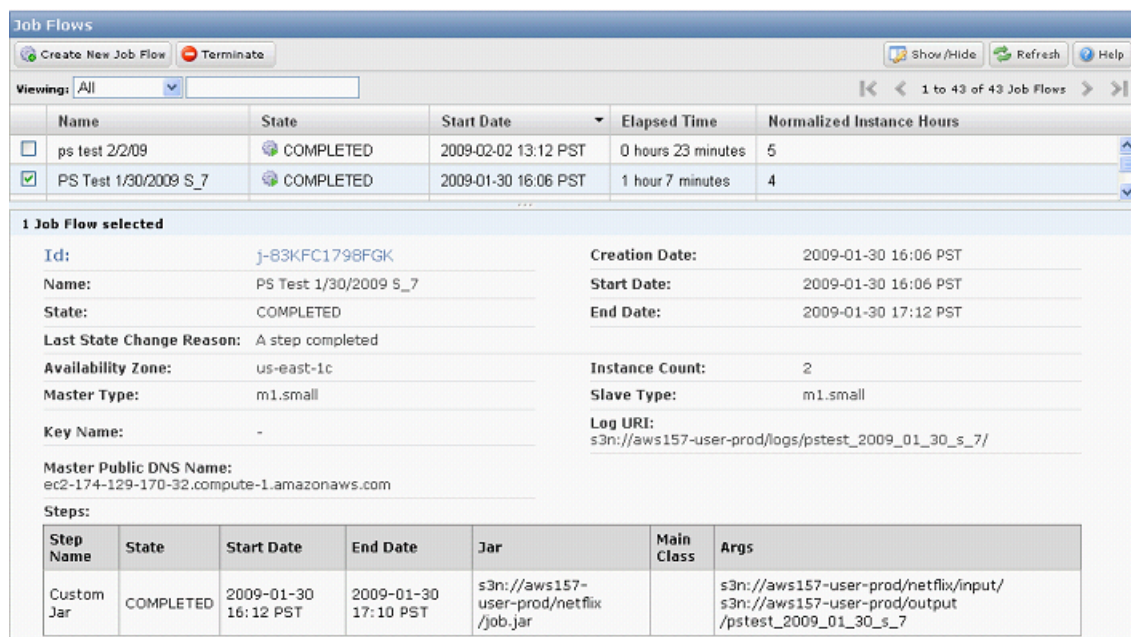


## How to View Details of the Job Flow

Once you start a job flow, you can monitor its status and get extended information about its execution. This section explains how to view the details of a job flow using the Elastic MapReduce console.

### To view the details of a job flow

- On the **Job Flows** page, select the check box next to the job flow you created. The **Job Flow selected** pane appears and provides detailed information about the selected job flow.



## Additional Console Tutorials

For additional console tutorials, go to [Introduction to Amazon Elastic MapReduce](#), [Finding Similar Items with Amazon Elastic MapReduce](#), [Python, and Hadoop Streaming](#), and [Using a Pig Script with the Console Video Tutorial](#).

## Send Us Your Feedback Now

Your input is important to us to help make our documentation helpful and easy to use. Please take a minute to give us your feedback on how well we were able to help you get started with Amazon Elastic MapReduce. Just click this [Feedback](#) link. Thank you.

## What's Next?

Now that you've used this guide to run your first job flow, you've become familiar with the architecture of the system, some of its basic functionality, the kind of responses you can expect, and the scope of the data you can process. The following section explains how to learn more about Elastic MapReduce and how to implement advanced Elastic MapReduce functionality in your applications.

# Where Do I Go from Here?

---

In the preceding tutorial you learned how to complete basic Elastic MapReduce tasks. This section answers common questions and tells you where to get more information to gain a deeper understanding of Elastic MapReduce. You can find answers to most of your questions in the [Amazon Elastic MapReduce Developer Guide](#).

## **Where do I find code samples for Elastic MapReduce?**

To find Elastic MapReduce code samples and sample applications, go to [Sample Data Processing Applications](#).

## **Where can I find the Elastic MapReduce command line interface that makes debugging job flows easier?**

To download the Elastic MapReduce command line interface, go to [Amazon Elastic MapReduce Ruby Client](#) and click **Download**. For information about command line interface tutorials, go to [Amazon Elastic MapReduce Ruby Client](#).

## **Where can I ask questions about Elastic MapReduce and get answers for free?**

The [AWS Support Center](#) is the home page for AWS Technical Support. The page includes links to our [Discussion Forums](#) where you can ask questions of fellow developers and Amazon support personnel. On the same page, you'll find links to Elastic MapReduce Technical FAQs, and the Service Status page. To get answers using the Elastic MapReduce documentation, go to the [Amazon Elastic MapReduce Developer Guide](#).

## **Can I process compressed files with Elastic MapReduce?**

Elastic MapReduce supports operating on the following compressed file formats: .gz, .deflate. For more information, see [Hadoop Data Compression](#).

## **Can I use Apache Pig with Elastic MapReduce?**

Elastic MapReduce supports Apache Pig. For more information, go to the [Pig tutorial](#) that shows how to analyze log files, and the [Pig video tutorial](#) that shows how to use Pig in interactive and batch modes. For more information about using Pig and Elastic MapReduce, go to [Introduction to Amazon Elastic MapReduce](#).

## **Can I use cascading with Elastic MapReduce?**

Yes, you can use cascading. For more information about a sample cascading application, go to [Cascading.Multitool](#).



### **I need to use multiple steps in my job flow. How do I do that?**

You can add steps to a running job flow to debug an application or to add an extra processing step. For more information, go to [Job Flows with Multiple Steps](#). To include multiple steps in your `RunJobFlow` request, go to [How to Add Steps to a Job Flow](#).

### **Can I tune the EC2 cluster that's running my job flow?**

Elastic MapReduce enables you to specify the number and kind of EC2 instances in the cluster, which is the primary means of affecting the speed with which your job flow completes. Elastic MapReduce by default sets many Hadoop parameters. Some of these parameter values can be overridden by parameter values set in a `RunFlowJob` request. For more information, see [EC2 Cluster Tuning](#).

### **How can I debug my job flow using Hadoop logs?**

To monitor the progress of the job flow, you can SSH into the master node and either look at the Hadoop log files directly or access the user interface that Hadoop publishes to the web server running on the master node. For more information, see [How to Debug Job Flows Using Log Files](#). For more information about using SSH to examine Hadoop log files, go to [How to Monitor Job Flow Status Using SSH](#).

### **Can I upload an application or additional data to be used by my job flow?**

You can upload an application or data into Amazon S3 for Elastic MapReduce to use. You supply the bucket path to the application or data using the `args` parameter in a `RunJobFlow` request. For more information, go to [How to Use Additional Files and Libraries With the Mapper or Reducer and DistributedCache](#).

### **Where can I see extended product FAQs**

The [Amazon Elastic MapReduce Technical FAQ](#) covers the top 20 questions developers have asked about Elastic MapReduce. [Amazon Elastic MapReduce](#) is the primary web page for information about Amazon Elastic MapReduce.

### **Does Elastic MapReduce support Pig or Hive?**

Elastic MapReduce supports both. For more information, go to:

- Running Hive on Amazon ElasticMap Reduce—<http://developer.amazonwebservices.com/connect/entry.jspa?externalID=2857>
- Pig Video Tutorial—<http://s3.amazonaws.com/awsVideos/AmazonElasticMapReduce/ElasticMapReduce-PigTutorial.html>

### **Where can I learn about the Elastic MapReduce API?**

The Elastic MapReduce API enables you to process data using Hadoop programmatically. This allows you to integrate Hadoop processing into other applications you have or to create your own console or command line interface. The API enables you to use the full functionality of Elastic MapReduce. For more information, go to [Starting and Managing Job Flows Using the API](#).

### **How do I cancel my registration with Elastic MapReduce?**

You can cancel your registration with Amazon Elastic MapReduce at any time.

### **To cancel your registration with Amazon Elastic MapReduce**

1. Go to [aws.amazon.com](http://aws.amazon.com).
2. Point to **Your Account** and click **Account Activity**.  
The **Account Activity** page displays.

3. Click the **View/Edit Service** link under Amazon Elastic MapReduce.  
The **View/Edit Service** page displays.
4. Click the link, **cancel this service**.  
This link is typically in the last sentence of the opening paragraph.

## Send Us Your Feedback Now

Your input is important to us to help make our documentation helpful and easy to use. Please take a minute to give us your feedback on how well we were able to help you get started with Elastic MapReduce. Just click this [Feedback](#) link. Thank you.

# Document Conventions

---

This section lists the common typographical and symbol use conventions for AWS technical publications.

## Typographical Conventions

This section describes common typographical use conventions.

Convention	Description/Example
Call-outs	<p>A call-out is a number in the body text to give you a visual reference. The reference point is for further discussion elsewhere.</p> <p>You can use this resource regularly. <b>1</b></p>
Code in text	<p>Inline code samples (including XML) and commands are identified with a special font.</p> <p>You can use the command <code>java -version</code>.</p>
Code blocks	<p>Blocks of sample code are set apart from the body and marked accordingly.</p> <pre># ls -l /var/www/html/index.html -rw-rw-r-- 1 root root 1872 Jun 21 09:33 /var/www/html/ index.html # date Wed Jun 21 09:33:42 EDT 2006</pre>
Emphasis	<p>Unusual or important words and phrases are marked with a special font.</p> <p>You <i>must</i> sign up for an account before you can use the service.</p>
Internal cross references	<p>References to a section in the same document are marked.</p> <p>See <a href="#">Document Conventions (p. 24)</a>.</p>
Logical values, constants, and regular expressions, abstracta	<p>A special font is used for expressions that are important to identify, but are not code.</p> <p>If the value is <code>null</code>, the returned response will be <code>false</code>.</p>

Convention	Description/Example
Product and feature names	Named AWS products and features are identified on first use. Create an <i>Amazon Machine Image</i> (AMI).
Operations	In-text references to operations. Use the <code>GetHITResponse</code> operation.
Parameters	In-text references to parameters. The operation accepts the parameter <i>AccountID</i> .
Response elements	In-text references to responses. A container for one <code>CollectionParent</code> and one or more <code>CollectionItems</code> .
Technical publication references	References to other AWS publications. If the reference is hyperlinked, it is also underscored. For detailed conceptual information, see the <i>Amazon Mechanical Turk Developer Guide</i> .
User entered values	A special font marks text that the user types. At the password prompt, type <b>MyPassword</b> .
User interface controls and labels	Denotes named items on the UI for easy identification. On the <b>File</b> menu, click <b>Properties</b> .
Variables	When you see this style, you must change the value of the content when you copy the text of a sample to a command line. % ec2-register <i>&lt;your-s3-bucket&gt;</i> /image.manifest See also the following symbol convention.

## Symbol Conventions

This section describes the common use of symbols.

Convention	Symbol	Description/Example
Mutually exclusive parameters	(Parentheses   and   vertical   bars)	Within a code description, bar separators denote options from which one must be chosen.  <code>% data = hdfread (start   stride   edge)</code>
Optional parameters XML variable text	[square brackets]	Within a code description, square brackets denote completely optional commands or parameters.  <code>% sed [-n, -quiet]</code>  Use square brackets in XML examples to differentiate them from tags.  <code>&lt;CustomerId&gt;[ ID]&lt;/CustomerId&gt;</code>
Variables	<arrow brackets>	Within a code sample, arrow brackets denote a variable that must be replaced with a valid value.  <code>% ec2-register &lt;your-s3-bucket&gt;/image.manifest</code>