

The Easiest Way to Run Apache® Spark™ Jobs

How-To Guide

The Easiest Way to Run Apache Spark Jobs

Recently, Databricks added a new feature, Jobs, to our cloud service. You can find a detailed overview of this feature in our [blog about jobs](#).

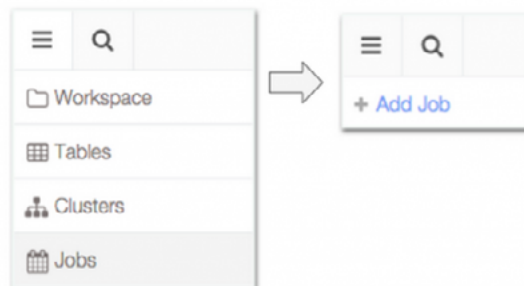
This feature allows you to programmatically run Apache Spark jobs on Amazon's EC2 easier than ever before. This how-to will provide a quick tour of this feature.

What is a Job?

The job feature is very flexible. You can run a job not only on any Spark JAR, but also notebooks you have created with Databricks. In addition, notebooks can be used as scripts to create sophisticated pipelines.

How to run a Job?

As shown below, Databricks offers an intuitive, easy to use interface to create a job.



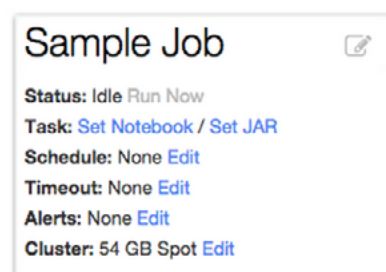
When creating a job, you will first need to specify the name of the job. By default, a job will use a new cluster size of 54GB each time it runs, however you will also have the option to change a few parameters of the cluster to fit your needs:

Cluster Type: New or existing cluster. If you choose to use a new cluster, Databricks will also automatically tear down the cluster once the job is completed.

Memory: Determines the performance of the job.

Spot Instance: You can choose to use Spot Instances to reduce your costs.

Next, you need to specify the notebook or the JAR you intend to run as a job, the input arguments of the job (both JARs and notebooks can take input arguments), and the job's configuration parameters: schedule, timeout, alerts, and the type of EC2 instances you would like the job to use. Next, we consider each of these configuration parameters in turn.



The Easiest Way to Run Apache® Spark™ Jobs

Scheduling

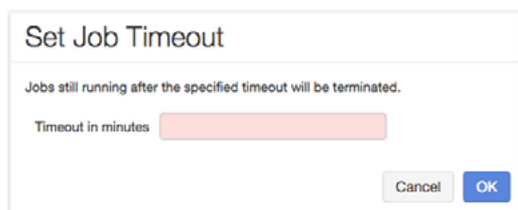
You can run any job periodically, by simply specifying the starting time and the interval, as shown below.



The 'Schedule Job' dialog box allows users to configure periodic job execution. It includes a frequency dropdown set to 'Every', a unit dropdown set to 'hour', and a starting time field set to '00:00' in 'US/Pacific' time. A 'show cron syntax' link is present. 'Cancel' and 'OK' buttons are at the bottom right.

Timeout

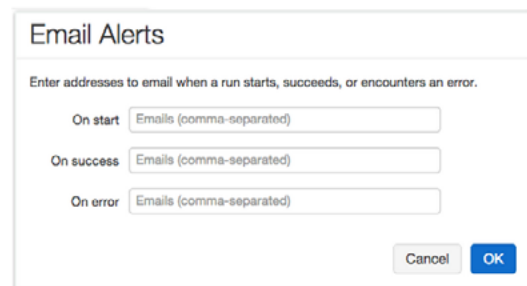
Optionally, you can set a timeout which specifies the time the job is allowed to run before being terminated. This feature is especially useful when handling runaway jobs, and to make sure an instance of a periodic job terminates before the next instance begins. If no timeout is specified and a job instance takes more than the scheduled period, no new instances are started before the current one terminates.



The 'Set Job Timeout' dialog box informs users that 'Jobs still running after the specified timeout will be terminated.' It features a 'Timeout in minutes' input field with a red border. 'Cancel' and 'OK' buttons are at the bottom right.

Alerts

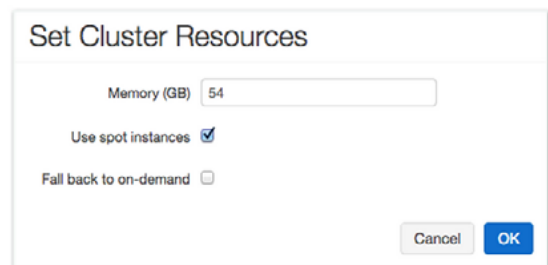
When running production jobs, it is critical get alerts when any significant event occurs. Databricks allows you to specify the events you would like to be alerted about via e-mail: when job starts, when it successfully finishes, or when it finishes with error.



The 'Email Alerts' dialog box prompts users to 'Enter addresses to email when a run starts, succeeds, or encounters an error.' It contains three input fields for 'On start', 'On success', and 'On error', each with a placeholder 'Emails (comma-separated)'. 'Cancel' and 'OK' buttons are at the bottom right.

Resource type

Finally, you can specify whether you would want to use spot or on-demand instances to run the job.



The 'Set Cluster Resources' dialog box allows users to configure job resources. It includes a 'Memory (GB)' input field set to '54'. There are two checkboxes: 'Use spot instances' (checked) and 'Fall back to on-demand' (unchecked). 'Cancel' and 'OK' buttons are at the bottom right.

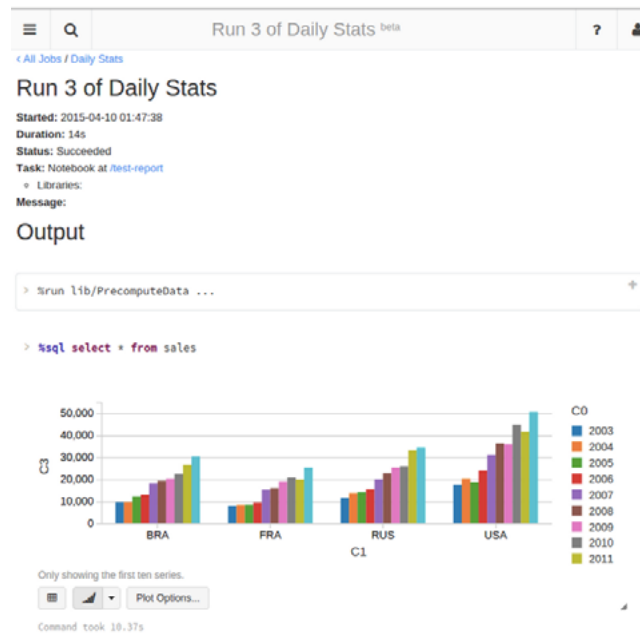
Completed runs

< Previous 20		Next 20 >			
Run	Start Time	Launched	Duration	Status	Message
Run 6	2015-02-06 13:17:38	Manually	1m 10s	Succeeded	
Run 5	2015-02-06 13:10:40	Manually	6m 39s	Cancelled	
Run 4	2015-02-06 12:23:56	Manually	44m 23s	Cancelled	
Run 3	2015-02-06 12:21:54	Manually	59s	Failed	
Run 2	2015-02-06 11:12:26	Manually	1h 6m 15s	Cancelled	
Run 1	2015-02-06 11:08:30	Manually	3m 52s	Cancelled	
< Previous 20		Next 20 >			

[illegible]

The Easiest Way to Run Apache® Spark™ Jobs

Similarly, the figure below shows the output of running a notebook as a job. Incidentally, the output is the same as running the notebook manually.



Summary

Databricks provides a powerful, yet easy to use feature to run not only Spark JARs compiled by any Spark install, but also notebooks created with Databricks. If you'd like to run your own jobs with Databricks, you can evaluate Databricks with a trial account now.

Additional Resources

Other Databricks how-tos can be found at:

[Analyzing Apache Access Logs with Databricks](#)

Evaluate Databricks with a trial account now:

databricks.com/try-databricks