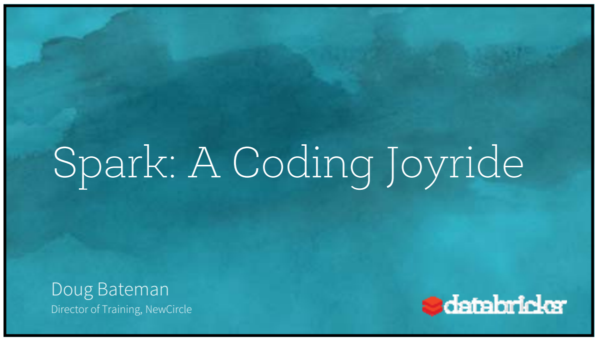
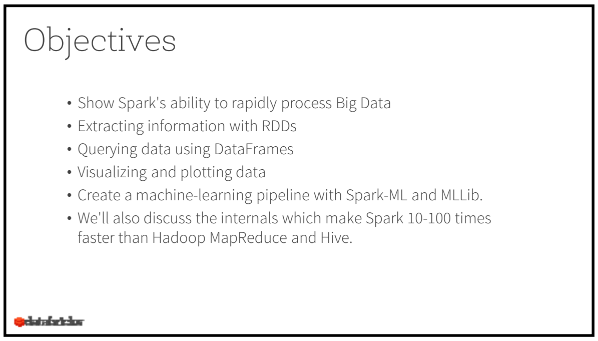


Spark: A Coding Joyride

Doug Bateman
Director of Training, NewCircle



Objectives

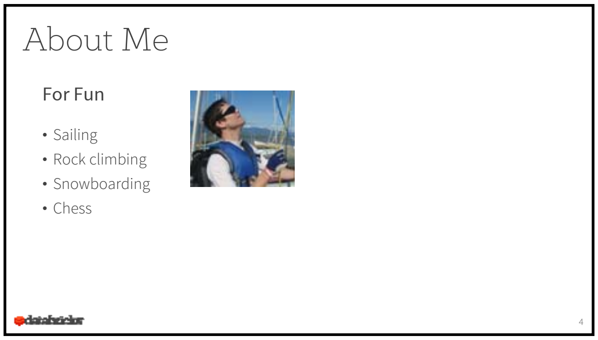
- Show Spark's ability to rapidly process Big Data
- Extracting information with RDDs
- Querying data using DataFrames
- Visualizing and plotting data
- Create a machine-learning pipeline with Spark-ML and MLLib.
- We'll also discuss the internals which make Spark 10-100 times faster than Hadoop MapReduce and Hive.

About Me

Engineer, Architect & Instructor

- Developing with Java since 1995 (Java 1.0)
- +15yrs as software developer, architect, and consultant.
- Director of Training at NewCircle
- Curriculum Lead at NewCircle

3



About Me

For Fun

- Sailing
- Rock climbing
- Snowboarding
- Chess

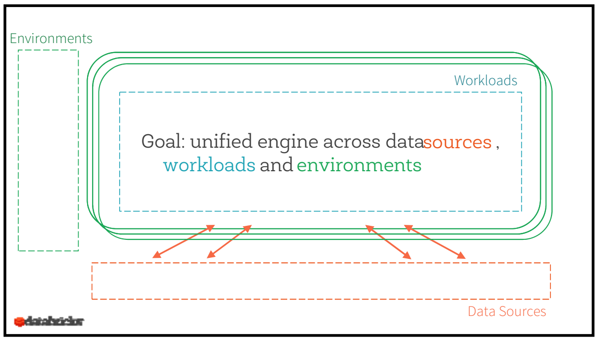
4

Who are you?

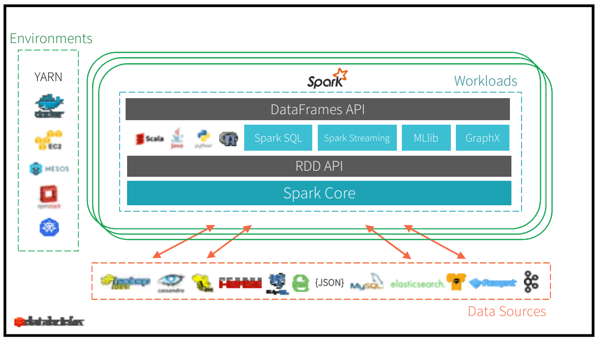
0) I am new to spark.

1) I have used Spark hands on before…

2) I have more than 1 year hands on experience with spark..

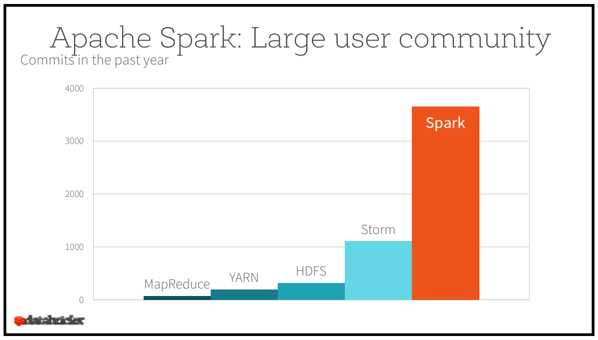


Goal: unified engine across datasources , workloads and environments



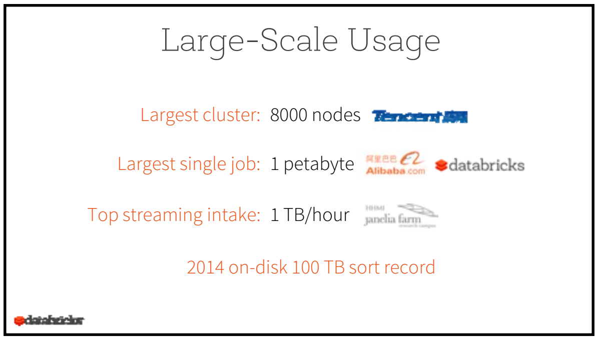
Spark – 100% open source and mature

Used in production by over 500 organizations. From fortune 100 to small innovators



Apache Spark: Large user community

Commits in the past year



Large-Scale Usage

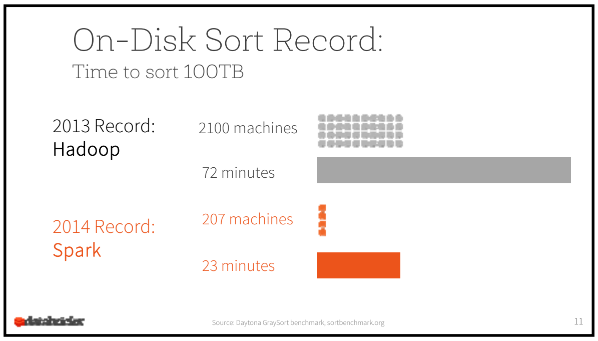
Largest cluster: 8000 nodes

Largest single job: 1 petabyte

Top streaming intake: 1 TB/hour

2014 on-disk 100 TB sort record

Tencent 腾讯
Alibaba.com
databricks
janelia farm
databricks



On-Disk Sort Record:

Time to sort 100TB

2013 Record:
Hadoop

2100 machines

72 minutes

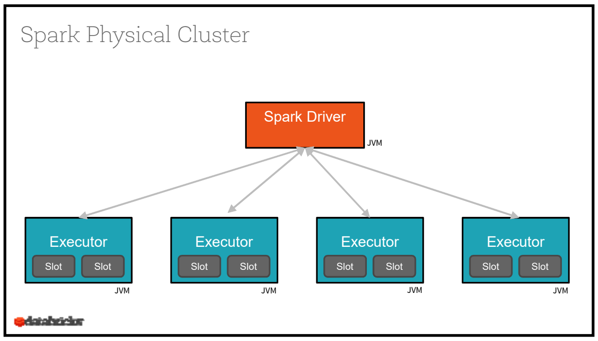
2014 Record:
Spark

207 machines

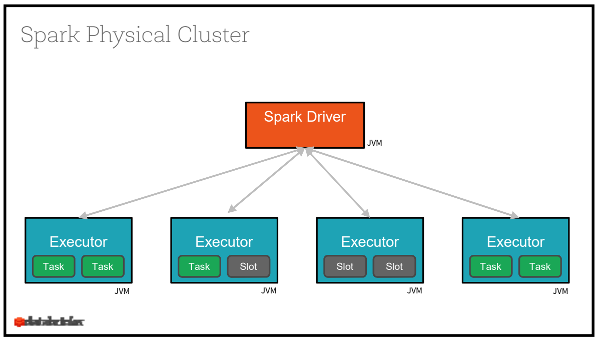
23 minutes

Source: Daytona GraySort benchmark, sortbenchmark.org

11

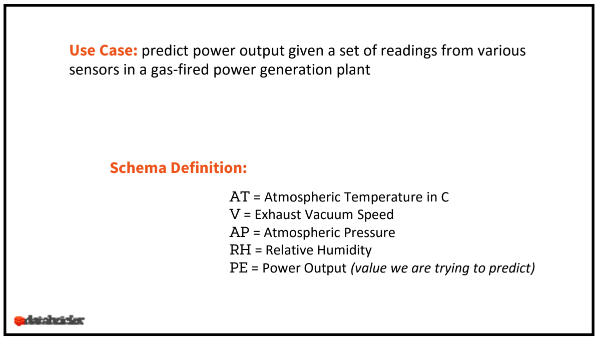


Spark Physical Cluster



Spark Physical Cluster

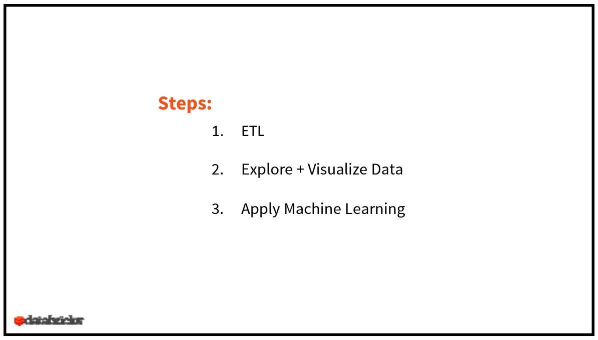
Power Plant Demo



Use Case: predict power output given a set of readings from various sensors in a gas-fired power generation plant

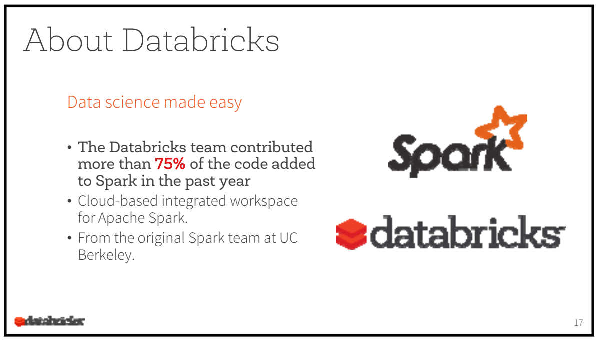
Schema Definition:

AT = Atmospheric Temperature in C
V = Exhaust Vacuum Speed
AP = Atmospheric Pressure
RH = Relative Humidity
PE = Power Output *(value we are trying to predict)*



Steps:

1. ETL
2. Explore + Visualize Data
3. Apply Machine Learning

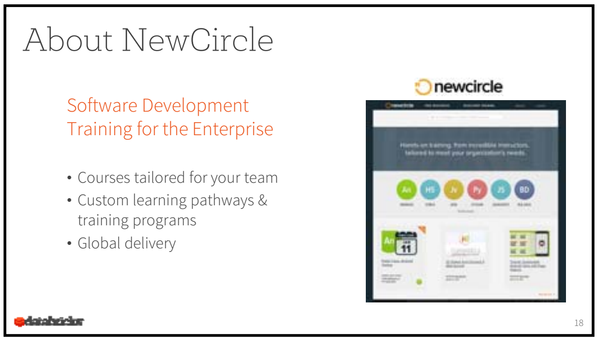


About Databricks

Data science made easy

- The Databricks team contributed more than 75% of the code added to Spark in the past year
- Cloud-based integrated workspace for Apache Spark.
- From the original Spark team at UC Berkeley.

17

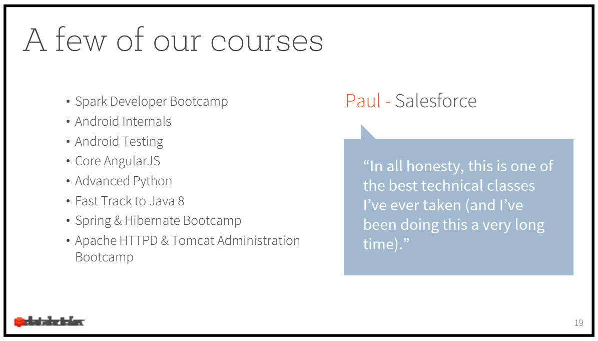


About NewCircle

Software Development Training for the Enterprise

- Courses tailored for your team
- Custom learning pathways & training programs
- Global delivery

18



A few of our courses

- Spark Developer Bootcamp
- Android Internals
- Android Testing
- Core AngularJS
- Advanced Python
- Fast Track to Java 8
- Spring & Hibernate Bootcamp
- Apache HTTPD & Tomcat Administration Bootcamp

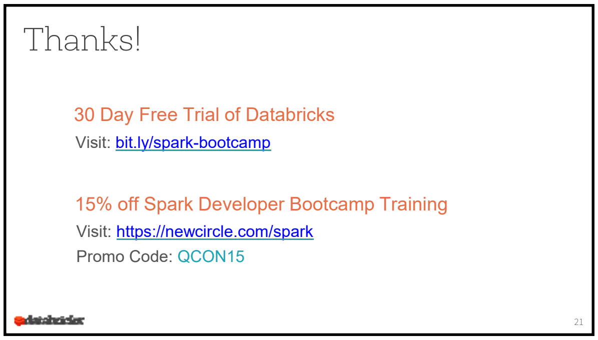
Paul - Salesforce

> "In all honesty, this is one of the best technical classes I've ever taken (and I've been doing this a very long time)."

19

Learn more at:

https://databricks.com/spark/training



Thanks!

30 Day Free Trial of Databricks

Visit: bit.ly/spark-bootcamp

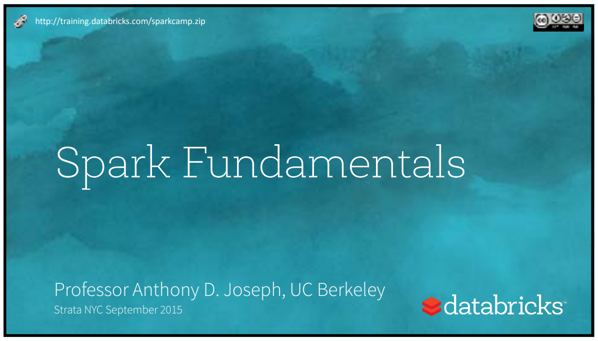
15% off Spark Developer Bootcamp Training

Visit: https://newcircle.com/spark

Promo Code: QCON15

21

Thank you.



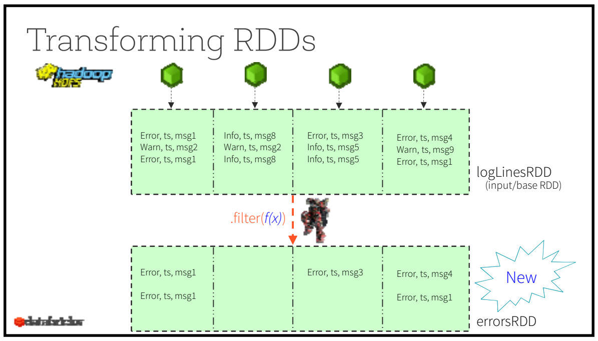
http://training.databricks.com/sparkcamp.zip

Spark Fundamentals

Professor Anthony D. Joseph, UC Berkeley

Strata NYC September 2015

databricks



Transforming RDDs

Error, ts, msg1
Warn, ts, msg2
Error, ts, msg1

Info, ts, msg8
Warn, ts, msg2
Info, ts, msg8

Error, ts, msg3
Info, ts, msg5
Info, ts, msg5

Error, ts, msg4
Warn, ts, msg9
Error, ts, msg1

logLinesRDD
(input/base RDD)

.filter(*f(x)*)

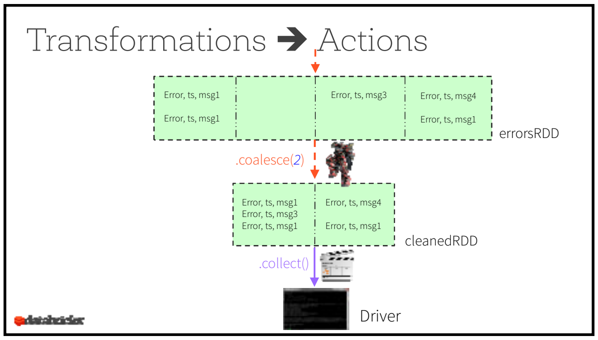
Error, ts, msg1
Error, ts, msg1

Error, ts, msg3

Error, ts, msg4
Error, ts, msg1

New

errorsRDD



Transformations ➔ Actions

Error, ts, msg1
Error, ts, msg1

Error, ts, msg3

Error, ts, msg4
Error, ts, msg1

errorsRDD

.coalesce(*2*)

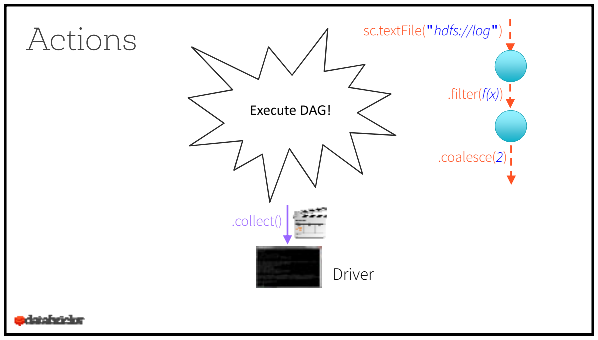
Error, ts, msg1
Error, ts, msg3
Error, ts, msg1

Error, ts, msg4
Error, ts, msg1

cleanedRDD

.collect()

Driver



Actions

Execute DAG!

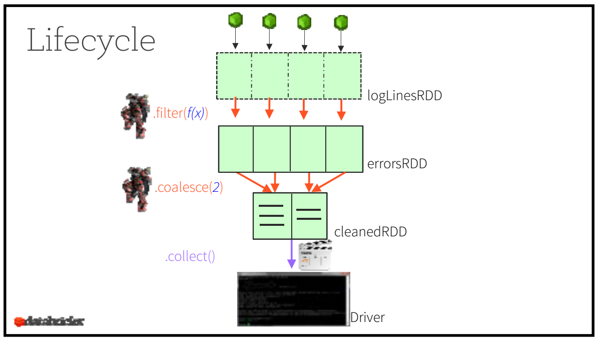
sc.textFile("*hdfs://log*")

.filter(*f(x)*)

.coalesce(*2*)

.collect()

Driver



Lifecycle

logLinesRDD

.filter(*f(x)*)

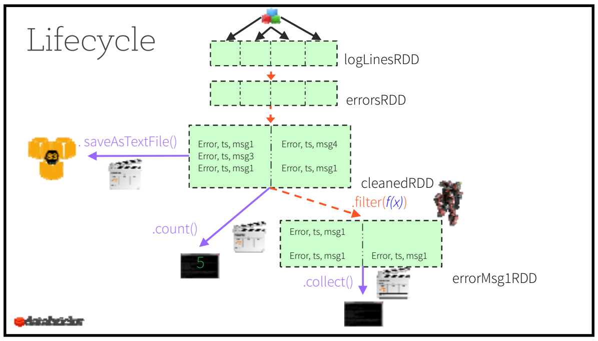
errorsRDD

.coalesce(*2*)

cleanedRDD

.collect()

Driver



Lifecycle

logLinesRDD

errorsRDD

. saveAsTextFile()

Error, ts, msg1
Error, ts, msg3
Error, ts, msg1

Error, ts, msg4
Error, ts, msg1

cleanedRDD

.filter(*f(x)*)

.count()

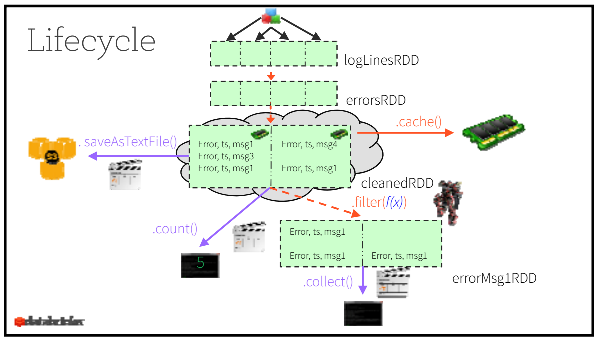
5

Error, ts, msg1
Error, ts, msg1

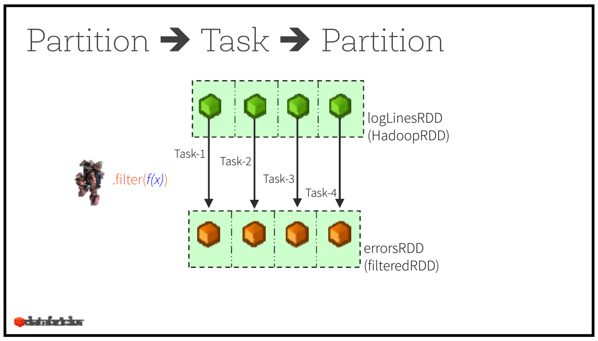
Error, ts, msg1

.collect()

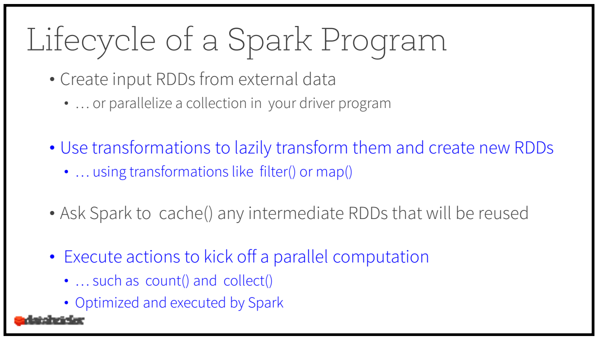
errorMsg1RDD



Lifecycle

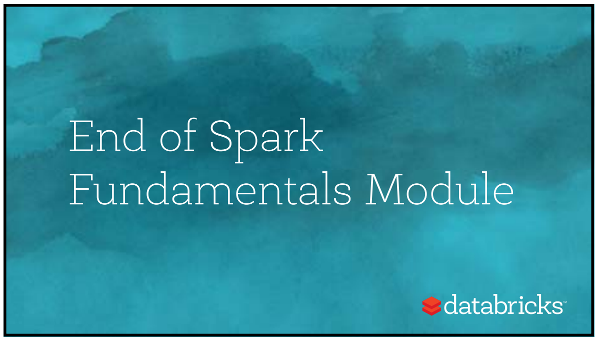


Partition ➔ Task ➔ Partition



Lifecycle of a Spark Program

- Create input RDDs from external data
  - … or parallelize a collection in your driver program
- Use transformations to lazily transform them and create new RDDs
  - … using transformations like filter() or map()
- Ask Spark to cache() any intermediate RDDs that will be reused
- Execute actions to kick off a parallel computation
  - … such as count() and collect()
  - Optimized and executed by Spark



End of Spark Fundamentals Module

databricks