



Which Data Broke My Code? Inspecting Spark Transformations

Vinod K. Nair, Director of Product Management @ Pepperdata

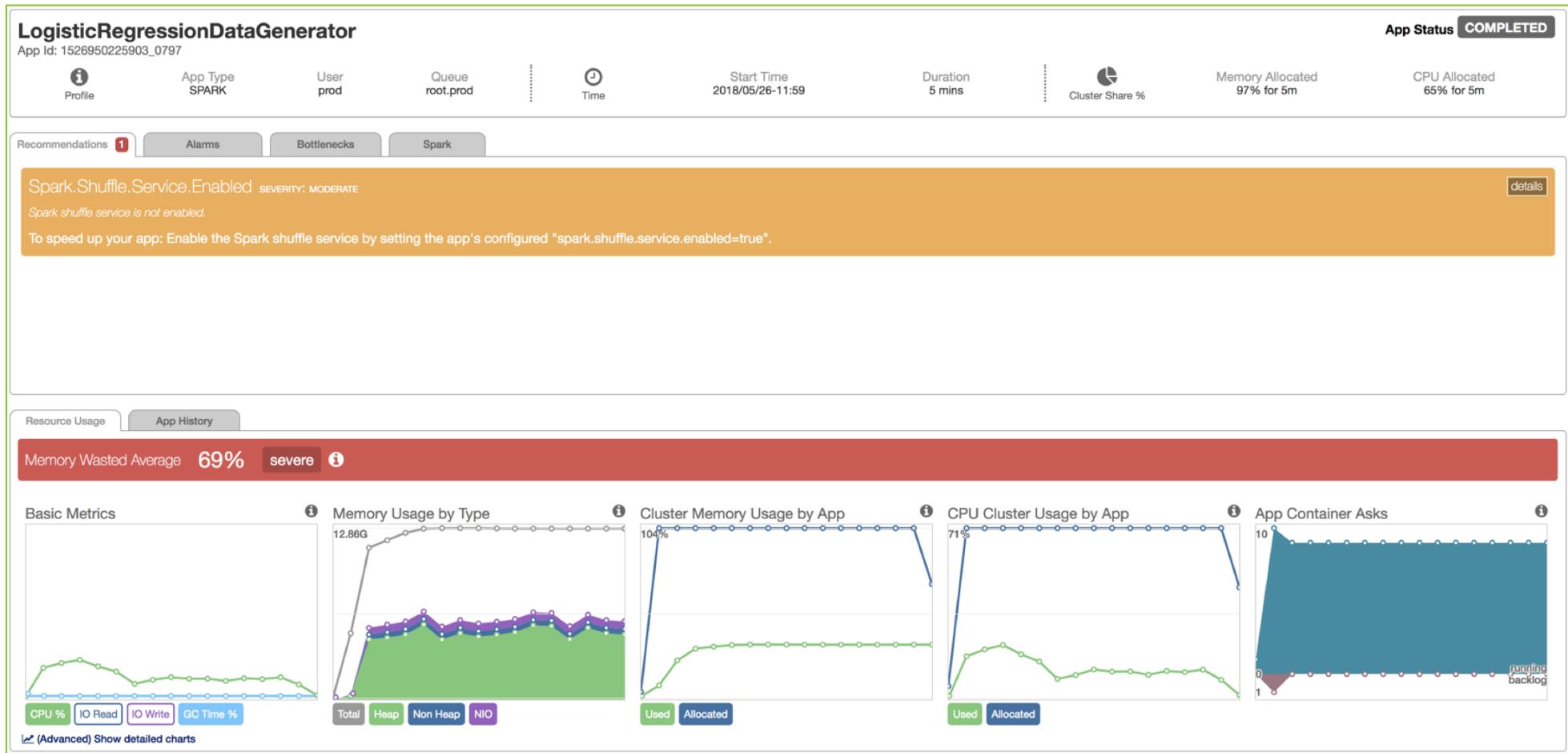
#DevSAIS12

Talk Outline

- Introduction
- Problem: ‘laziness’ makes debugging hard
- Solution: interactive inspection of RDDs
- Demo
- Q & A

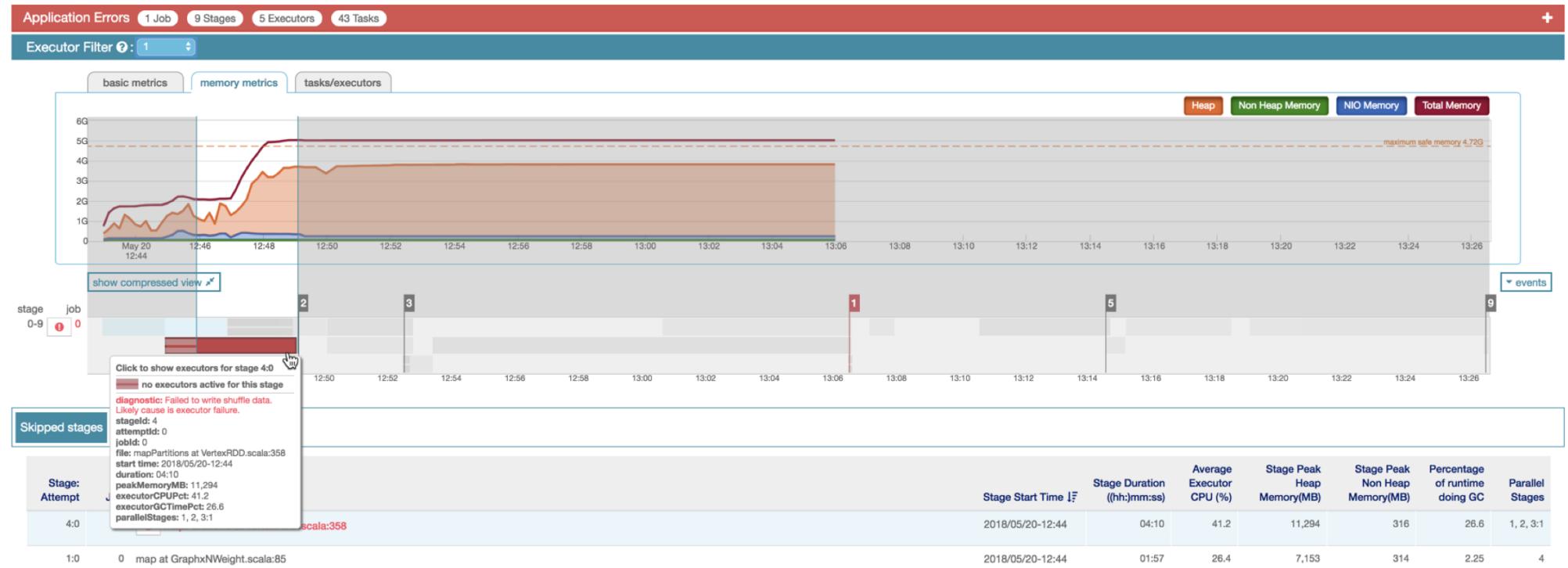
Introduction to Pepperdata

Application Performance Management (APM) for Spark (& Hadoop)



NWeightGraphX

App Id: 1525663230056_2131 App Configuration



Problem: ‘laziness’ makes debugging hard

RDD data unavailable until an ‘action’ triggers execution

Transformations are invisible

RDDs support two types of operations:

1. **transformations**, which create a new dataset from an existing one
2. **actions**, which return a value to the driver program after running a computation on the dataset

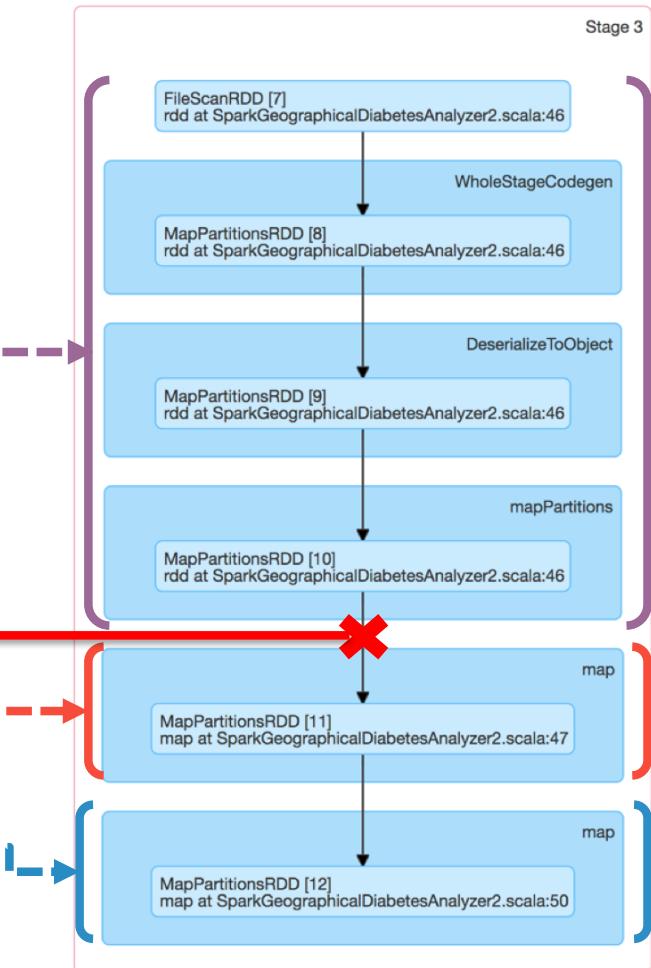
Transformations in Spark are *lazy*. They are only computed when an action requires a result to be returned to the driver program.

- <https://spark.apache.org/docs/latest/rdd-programming-guide.html>

```

36     val healthData = spark.read
37
38         .option("header", "true") // Use first line of all files as header
39
40         .option("inferSchema", "true") // Automatically infer data types
41
42         .csv("hdfs://tmp/health_data.csv")
43
44
45         // Filter to the relevant data aggregated by city, discard unused columns
46
47         val columnsToKeep = Seq("StateDesc", "Data_Value", "PopulationCount")
48
49         val filteredData = healthData.filter(healthData("GeographicLevel").==(("City")))
50
51         .filter(healthData("DataValueTypeID").==(("CrdPrv")))
52
53         .filter(healthData("MeasureId").==(("DIABETES")))
54
55
56         filteredData.take(10).foreach( println )
57
58
59         // Map the state to geographical region and convert the percentage to count of impacted people
60
61         val regionMappedImpactedData = filteredData.map(x => (getMappedRegionOrUnknown(x._1), x._2))
62
63         val reducedRegionMappedImpactedData = regionMappedImpactedData.reduceByKey(_ + _).sortByKey()
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99

```



Solution available today

- Sprinkle your code with print statements
- ‘hopefully’ catch the right transformation causing the problem
- If you don’t catch it – repeat process

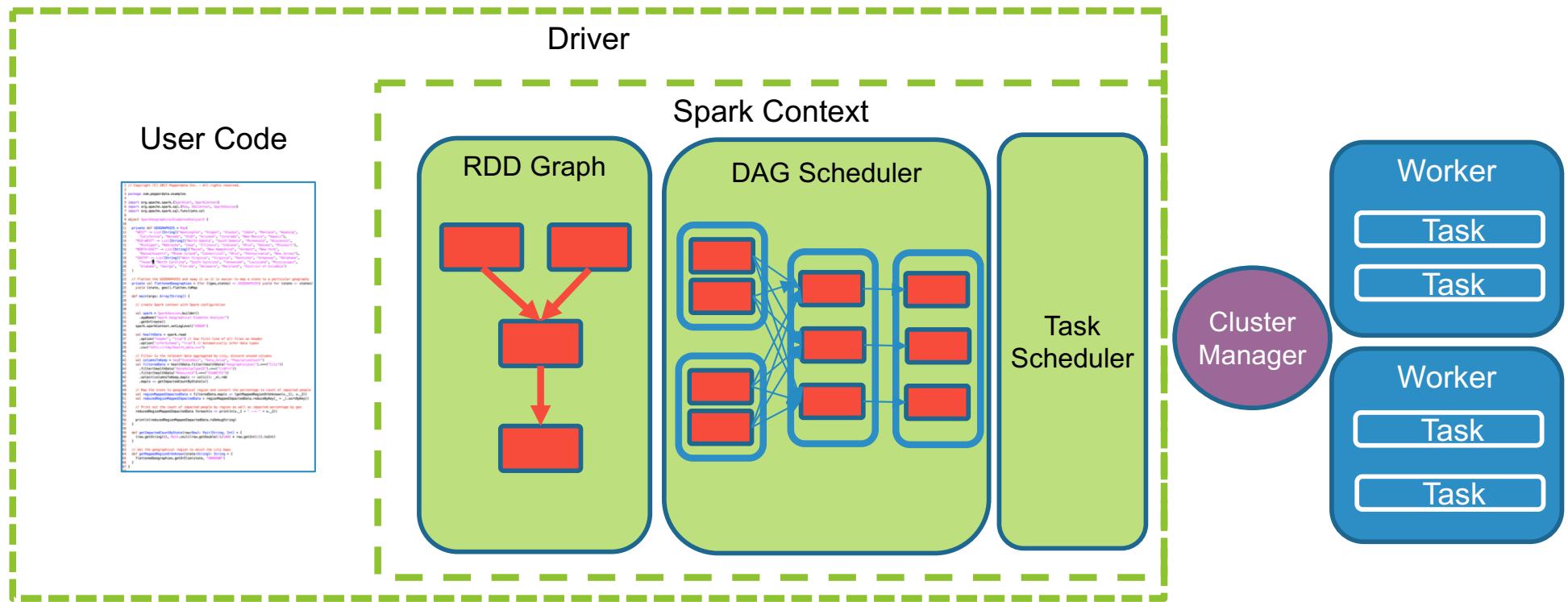
Our solution: interactive inspection of data in flight

Trigger an ‘action’ to enable inspection of any RDD in the DAG

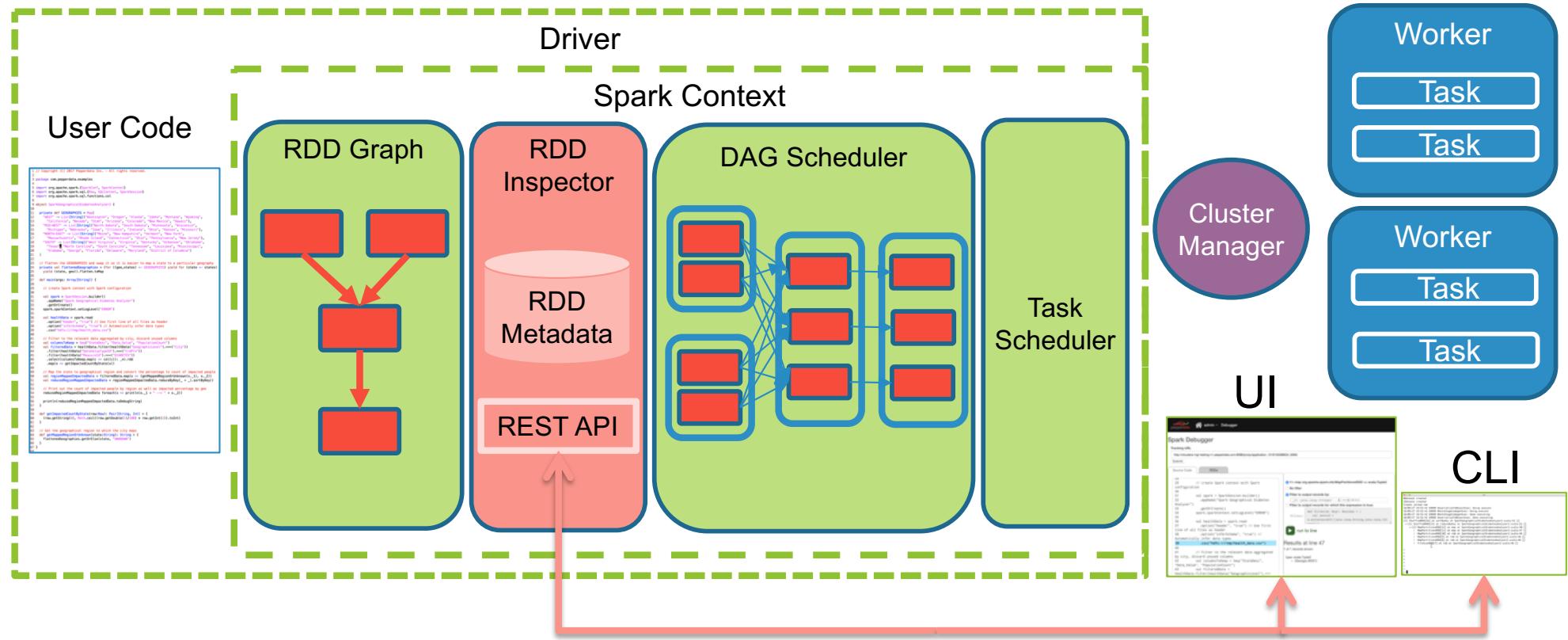
Solution requirements

- No user code changes required
- Work with any standard Spark distribution
- Provide a familiar interactive debugger interface

Solution overview



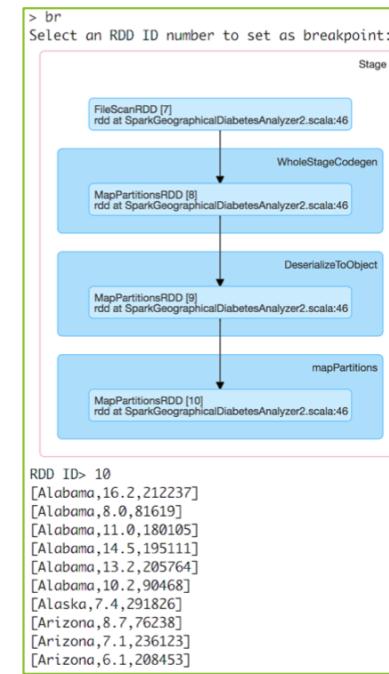
Solution overview



New Spark job to display RDD[10]



Original stage with RDD [10] as an intermediate transformation



Debug stage with RDD [10] as output

CLI command to inspect RDD[10]

ssh

```
(2) MapPartitionsRDD[5] at csv at SparkGeographicalDiabetesAnalyzer2.scala:39 □
|  hdfs://cloudera-mgr-testing-n1.pepperdata.com:8020/tmp/health_data.csv MapPartitionsRDD[4]
at csv at SparkGeographicalDiabetesAnalyzer2.scala:39 □
|  hdfs://cloudera-mgr-testing-n1.pepperdata.com:8020/tmp/health_data.csv HadoopRDD[3] at csv
at SparkGeographicalDiabetesAnalyzer2.scala:39 □
(2) MapPartitionsRDD[6] at csv at SparkGeographicalDiabetesAnalyzer2.scala:39 □
at csv at SparkGeographicalDiabetesAnalyzer2.scala:39 □
|  hdfs://cloudera-mgr-testing-n1.pepperdata.com:8020/tmp/health_data.csv HadoopRDD[3] at csv
at SparkGeographicalDiabetesAnalyzer2.scala:39 □
(2) MapPartitionsRDD[2] at csv at SparkGeographicalDiabetesAnalyzer2.scala:39 □
|  hdfs://cloudera-mgr-testing-n1.pepperdata.com:8020/tmp/health_data.csv MapPartitionsRDD[1]
at csv at SparkGeographicalDiabetesAnalyzer2.scala:39 □
|  hdfs://cloudera-mgr-testing-n1.pepperdata.com:8020/tmp/health_data.csv HadoopRDD[0] at csv
at SparkGeographicalDiabetesAnalyzer2.scala:39 □
(2) ShuffledRDD[16] at sortByKey at SparkGeographicalDiabetesAnalyzer2.scala:51 □
+- (2) ShuffledRDD[13] at reduceByKey at SparkGeographicalDiabetesAnalyzer2.scala:51 □
  +- (2) MapPartitionsRDD[12] at map at SparkGeographicalDiabetesAnalyzer2.scala:50 □
    |  MapPartitionsRDD[11] at map at SparkGeographicalDiabetesAnalyzer2.scala:47 □
    |  MapPartitionsRDD[10] at rdd at SparkGeographicalDiabetesAnalyzer2.scala:46 □
    |  MapPartitionsRDD[9] at rdd at SparkGeographicalDiabetesAnalyzer2.scala:46 □
    |  MapPartitionsRDD[8] at rdd at SparkGeographicalDiabetesAnalyzer2.scala:46 □
    |  FileScanRDD[7] at rdd at SparkGeographicalDiabetesAnalyzer2.scala:46 □
(2) MapPartitionsRDD[15] at sortByKey at SparkGeographicalDiabetesAnalyzer2.scala:51 □
|  MapPartitionsRDD[14] at sortByKey at SparkGeographicalDiabetesAnalyzer2.scala:51 □
|  ShuffledRDD[13] at reduceByKey at SparkGeographicalDiabetesAnalyzer2.scala:51 □
+- (2) MapPartitionsRDD[12] at map at SparkGeographicalDiabetesAnalyzer2.scala:50 □
  |  MapPartitionsRDD[11] at map at SparkGeographicalDiabetesAnalyzer2.scala:47 □
  |  MapPartitionsRDD[10] at rdd at SparkGeographicalDiabetesAnalyzer2.scala:46 □
  |  MapPartitionsRDD[9] at rdd at SparkGeographicalDiabetesAnalyzer2.scala:46 □
  |  MapPartitionsRDD[8] at rdd at SparkGeographicalDiabetesAnalyzer2.scala:46 □
  +-- RDD ID> 10
    [Alabama,16.2,212237]
    [Alabama,8.0,81619]
    [Alabama,11.0,180105]
    [Alabama,14.5,195111]
    [Alabama,13.2,205764]
    [Alabama,10.2,90468]
    [Alaska,7.4,291826]
    [Arizona,8.7,76238]
    [Arizona,7.1,236123]
    [Arizona,6.1,208453]
  > □
```

Action

Spark Jobs (?)

User: vnair
Total Uptime: 10 min
Scheduling Mode: FIFO
Completed Jobs: 6

Event Timeline

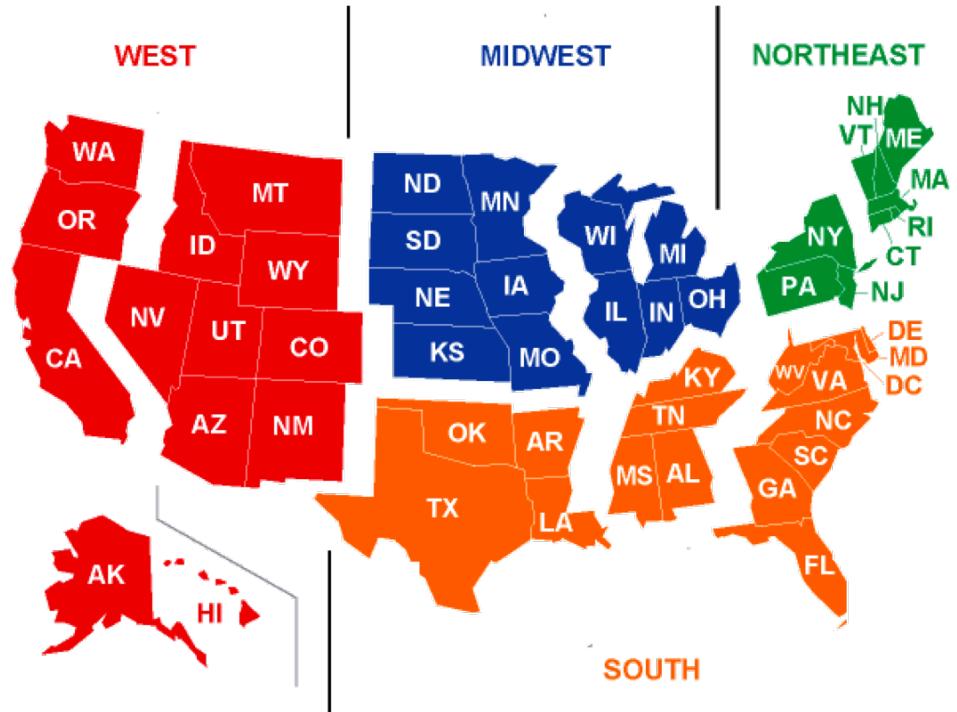
Job Id	Description	Submitted	Duration	Stages: Succeeded/Total	Tasks (for all stages): Succeeded/Total
5	DebuggerClient\$	2018/05/26 20:17:37	0.2 s	1/1	1/1
4	foreach at SparkGeographicalDiabetesAnalyzer2.scala:54	2018/05/26 20:07:59	0.2 s	2/2 (1 skipped)	4/4 (2 skipped)
3	sortByKey at SparkGeographicalDiabetesAnalyzer2.scala:51	2018/05/26 20:07:56	4 s	2/2	4/4
2	csv at SparkGeographicalDiabetesAnalyzer2.scala:39	2018/05/26 20:07:46	7 s	1/1	2/2
1	csv at SparkGeographicalDiabetesAnalyzer2.scala:39	2018/05/26 20:07:46	0.1 s	1/1	1/1
0	csv at SparkGeographicalDiabetesAnalyzer2.scala:39	2018/05/26 20:07:44	2 s	1/1	1/1

Demo

Interactive ‘debugger’ experience to inspect any RDD in the DAG

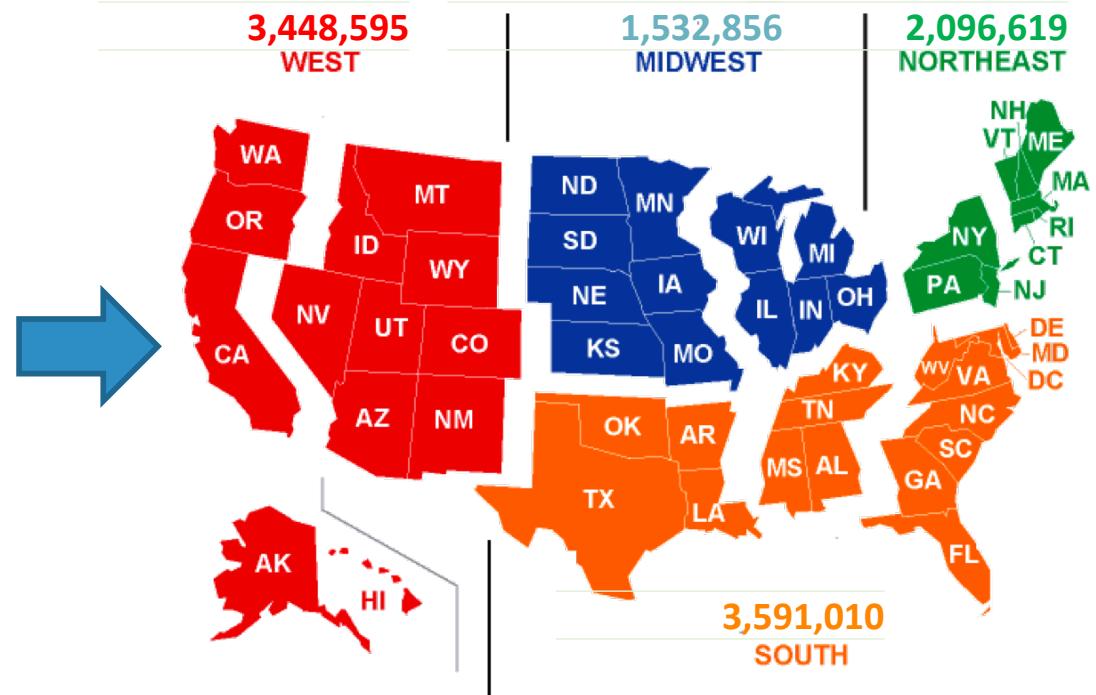
Analyze CDC's census data

- CDC's 500 cities project
(data.gov: 500 cities local data for better health)
- Input:
Major chronic diseases by city
- Output:
Diabetes in adults by US region



Distribution of diabetics by region

500_Cities_Local_Data_for_Better_Health.csv
2014, IA, Iowa, Davenport, Census Tract, BRFSS, Health Outcomes, 1919000-19163010101, Cancer (excluding skin cancer) among adults aged >=18 Years, %, CrdPrv, Crude prevalence,, Estimates suppressed for population less than 50, 3, "(41.60035282, -90.4980671216)", HLTHOUT, CANCER, 1919000, 19163010101, Cancer (except skin)
2014, IA, Iowa, Davenport, Census Tract, BRFSS, Health Outcomes, 1919000-19163010202, Cancer (excluding skin cancer) among adults aged >=18 Years, %, CrdPrv, Crude prevalence,, Estimates suppressed for population less than 50, 23, "(41.6051687821, -90.5841494896)", HLTHOUT, CANCER, 1919000, 19163010202, Cancer (except skin)
2014, IA, Iowa, Davenport, Census Tract, BRFSS, Health Outcomes, 1919000-19163010300, Cancer (excluding skin cancer) among adults aged >=18 Years, %, CrdPrv, Crude prevalence,, Estimates suppressed for population less than 50, 9, "(41.600491047, -90.6777509264)", HLTHOUT, CANCER, 1919000, 19163010300, Cancer (except skin)
2014, IA, Iowa, Davenport, Census Tract, BRFSS, Health Outcomes, 1919000-19163010600, Cancer (excluding skin cancer) among adults aged >=18 Years, %, CrdPrv, Crude prevalence, 3.7, 3.6, 3.9, . . . , 2422, "(41.5268094476, -90.5656367434)", HLTHOUT, CANCER, 19163010600, Cancer (except skin)
2014, IA, Iowa, Davenport, Census Tract, BRFSS, Health Outcomes, 1919000-19163010700, Cancer (excluding skin cancer) among adults aged >=18 Years, %, CrdPrv, Crude prevalence, 4.5, 4.4, 4.7, . . . , 1527, "(41.5293018757, -90.5806944595)", HLTHOUT, CANCER, 1919000, 19163010700, Cancer (except skin)
2014, IA, Iowa, Davenport, Census Tract, BRFSS, Health Outcomes, 1919000-19163010800, Cancer (excluding skin cancer) among adults aged >=18 Years, %, CrdPrv, Crude prevalence, 6.4, 6.1, 6.7, . . . , 3277, "(41.5280925473, -90.5944178646)", HLTHOUT, CANCER, 1919000, 19163010800, Cancer (except skin)
2014, IA, Iowa, Davenport, Census Tract, BRFSS, Health Outcomes, 1919000-19163010900, Cancer (excluding skin cancer) among adults aged >=18 Years, %, CrdPrv, Crude prevalence, 4.9, 4.7, 5.1, . . . , 1970, "(41.5194046979, -90.5858947174)", HLTHOUT, CANCER, 1919000, 19163010900, Cancer (except skin)
2014, IA, Iowa, Davenport, Census Tract, BRFSS, Health Outcomes, 1919000-19163011000, Cancer (excluding skin cancer) among adults aged >=18 Years, %, CrdPrv, Crude prevalence, 5.7, 5.4, 6.0, . . . , 2725, "(41.5047261649, -90.6115509526)", HLTHOUT, CANCER, 1919000, 19163011000, Cancer (except skin)
2014, IA, Iowa, Davenport, Census Tract, BRFSS, Health Outcomes, 1919000-19163011100, Cancer (excluding skin cancer) among adults aged >=18 Years, %, CrdPrv, Crude prevalence, 6.9, 6.5, 7.3, . . . , 3272, "(41.5355053826, -90.6078471834)", HLTHOUT, CANCER, 1919000, 19163011100, Cancer (except skin)
2014, IA, Iowa, Davenport, Census Tract, BRFSS, Health Outcomes, 1919000-19163011200, Cancer (excluding skin cancer) among adults aged >=18 Years, %, CrdPrv, Crude prevalence, 4.8, 4.6, 5.1, . . . , 2281, "(41.5367571624, -90.5962736008)", HLTHOUT, CANCER, 1919000, 19163011200, Cancer (except skin)



```
ssh  
-bash-4.2$ /usr/local/spark/bin/spark-submit --executor-memory 512m --deploy-mode client --class com.pepperdata.examples.SparkGeographicalDiabetesAnalyzer2 ~/cityaggregator-0.2.jar
```

Region	Adult Population with Diabetes
WEST	3,448,595
MID WEST	1,532,856
NORTH EAST	2,096,619
SOUTH	3,407,528
UNKNOWN	183,482

RDD transformations through the app

```
2014,CA,California,Alameda,City,BRFSS,Health  
Outcomes,0600562,Diagnosed diabetes among adults aged >=18  
Years,%,AgeAdjPrv,Age-adjusted  
prevalence,8.1,7.9,8.2,,73812,"(37.7650849031,-  
122.266489842)",HLTHOUT,DIABETES,0600562,,Diabetes  
...  
RDD [5]
```

Region	Impacted pop.
West	
Mid West	
North East	
South	RDD [16]

Filter ("DIABETES") & Map (State, Impacted pop.)

State	Impacted pop.
Alabama	205,764
Alabama	90,468
Alaska	291,826
Arizona	76,238
...	RDD [11]

Map
(Region, Impacted pop.)

Region	Impacted pop.
South	27,161
South	9,228
West	21,596
West	6,633
...	RDD [12]

Spark Debugger UI

The screenshot shows the Spark Debugger UI interface. On the left, the 'Source Code' tab is selected, displaying Scala code for a 'Spark Geographical Diabetes Analyzer'. A red box highlights line 47, which contains a 'map' transformation with a breakpoint. A callout box points to this line with the text "'Breakpoint' on any transformation'". On the right, the 'RDDs' tab is selected, showing a list of RDDs and their transformations. A red box highlights the 'No filter' option under 'Filter to output records by'. Another red box highlights the 'Filter to output records for which this expression is true:' input field. A callout box points to this section with the text 'URL Spark Web UI', 'List of RDDs', and 'RDD Filter'. Below this, a button labeled 'Run to line' with a play icon is shown. A callout box points to the results table with the text 'First 10 records matching filter'. The results table lists 10 records as 'scala.Tuple2' pairs.

```
def main(args: Array[String]) {  
    // create Spark context with Spark configuration  
    val spark = SparkSession.builder()  
        .appName("Spark Geographical Diabetes Analyzer")  
        .getOrCreate()  
    spark.sparkContext.setLogLevel("ERROR")  
  
    val healthData = spark.read  
        .option("header", "true") // Use first line of all files as header  
        .option("inferSchema", "true") // Automatically infer data types  
        .csv("hdfs:///tmp/health_data.csv")  
  
    // Filter to the relevant data aggregated by ci  
    val columnsToKeep = Seq("StateDesc", "Data_Value")  
    val filteredData = healthData.filter(healthData.  
        .filter(healthData("DataValueTypeID") === ("C"))  
        .filter(healthData("MeasureId"). === ("IABETES"))  
        .select(columnsToKeep).map(x => (x._1, x._2)))  
    .map(x => getImpactedCountByState(x))  
  
    // Map the state to geographical region and convert the percentage to count of impacted people  
    val regionMappedImpactedData = filteredData.map(x => (getMappedRegionOrUnknown(x._1), x._2))  
    val reducedRegionMappedImpactedData = regionMappedImpactedData.reduceByKey(_ + _).sortByKey()  
  
    // Print out the count of impacted people by region as well as impacted percentage by geo  
    reducedRegionMappedImpactedData foreach(x => println(x._1 + " -> " + x._2))  
  
    println(reducedRegionMappedImpactedData.toDebugString)  
}  
  
def getImpactedCountByState(row: Row): Pair[String, Int] = {  
    (row.getString(0), Math.ceil((row.getDouble(1)/100) * row.getInt(2)).toInt)  
}  
  
// Get the geographical region to which the city maps
```

URL Spark Web UI

List of RDDs

RDD Filter

'Breakpoint' on any transformation

Run to line

Results at line 47

10 of 10 records shown

type: scala.Tuple2

- (Alabama,34383)
- (Alabama,6530)
- (Alabama,19812)
- (Alabama,28292)
- (Alabama,27161)
- (Alabama,9228)
- (Alaska,21596)
- (Arizona,6633)
- (Arizona,16765)
- (Arizona,12716)

Spark Debugger Demo



To recap, you can...

- View data sets as they are transformed in your Spark App
 - no code changes are required
 - it works with any Spark distribution, and
 - it uses a familiar debugger interface to ‘set’ breakpoints and view RDDs in flight

What's next ?

Roadmap for Pepperdata's Spark Debugger

Areas of focus going forward

- UX Improvements
- Attach to a running app (streaming use case)
- Pause a job on hitting a condition
- Spark SQL support

To learn more visit the  booth (#407)