SparkR: Interactive Data Science at Scale

Shivaram Venkataraman

Fast

Scalable

Flexible

DataFrames

Packages

Plots

Fast

DataFames

Scalable

+

Packages

Flexible

Plots

Outline

SparkR Distributed Lists (RDD)
Design Details
SparkR DataFrames
Roadmap

RDD

Parallel Collection

RDD

Parallel Collection

Transformations

map
filter
groupBy
…

Actions

count
collect
saveAsTextFile
…

R + RDD =

R + RDD = R2D2

R + RDD =
RRDD

R + RDD =
RRDD

lapply
lapplyPartition
groupByKey
reduceByKey
sampleRDD
collect
cache
filter
…
broadcast
includePackage
textFile
parallelize

Example: word_count.R

```
library(SparkR)
lines <- textFile(sc, “hdfs://my_text_file”)
```

Example: word_count.R

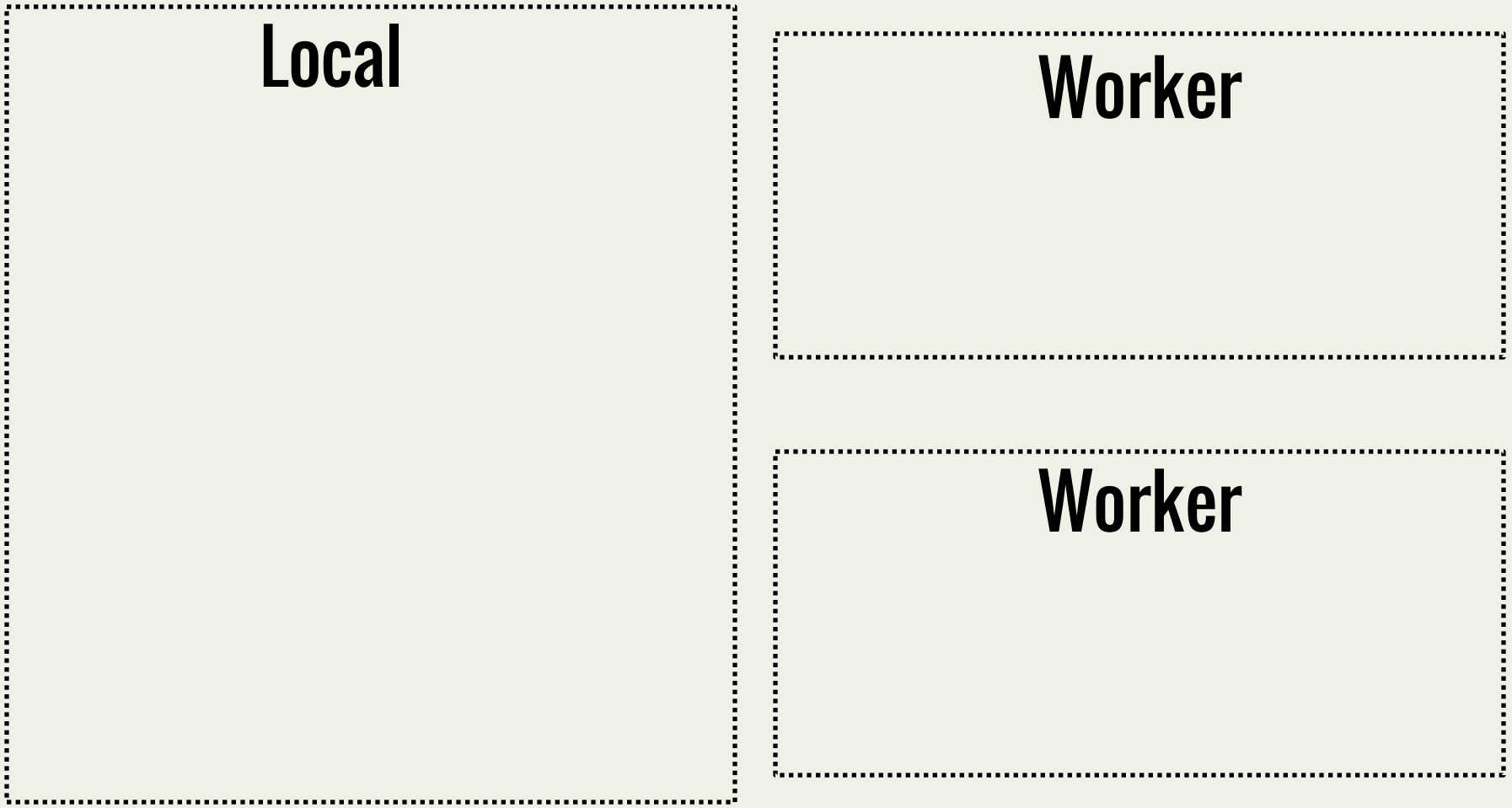
```
library(SparkR)
lines <- textFile(sc, “hdfs://my_text_file”)
words <- flatMap(lines,
                 function(line) {
                   strsplit(line, " ")[[1]]
                 })
wordCount <- lapply(words,
                    function(word) {
                        list(word, 1L)
                    })
```

Example: word_count.R

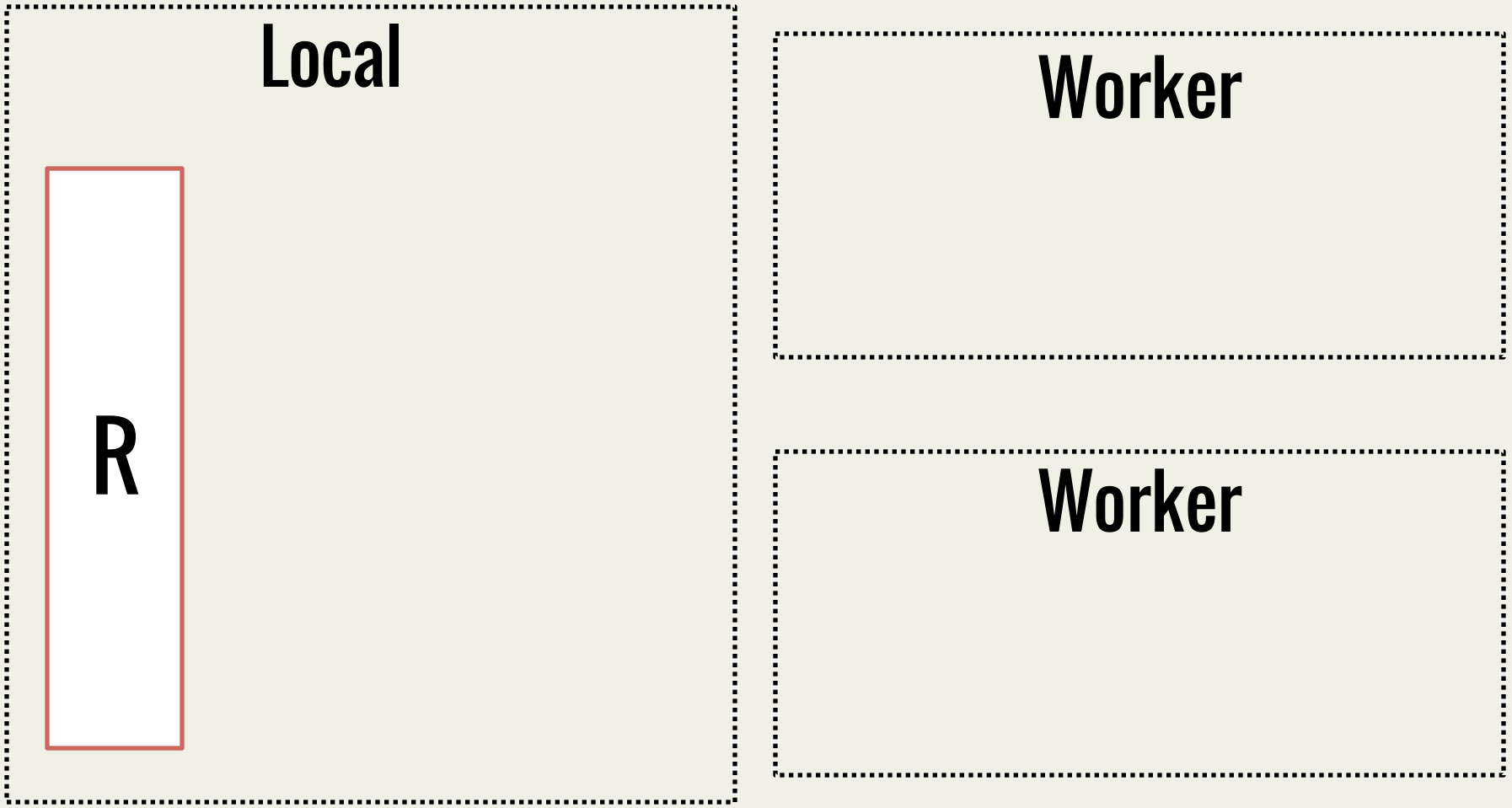
```
library(SparkR)
lines <- textFile(sc, “hdfs://my_text_file”)
words <- flatMap(lines,
                 function(line) {
                   strsplit(line, " ")[[1]]
                 })
wordCount <- lapply(words,
                    function(word) {
                        list(word, 1L)
                    })
counts <- reduceByKey(wordCount, "+", 2L)
output <- collect(counts)
```


How does this work ?

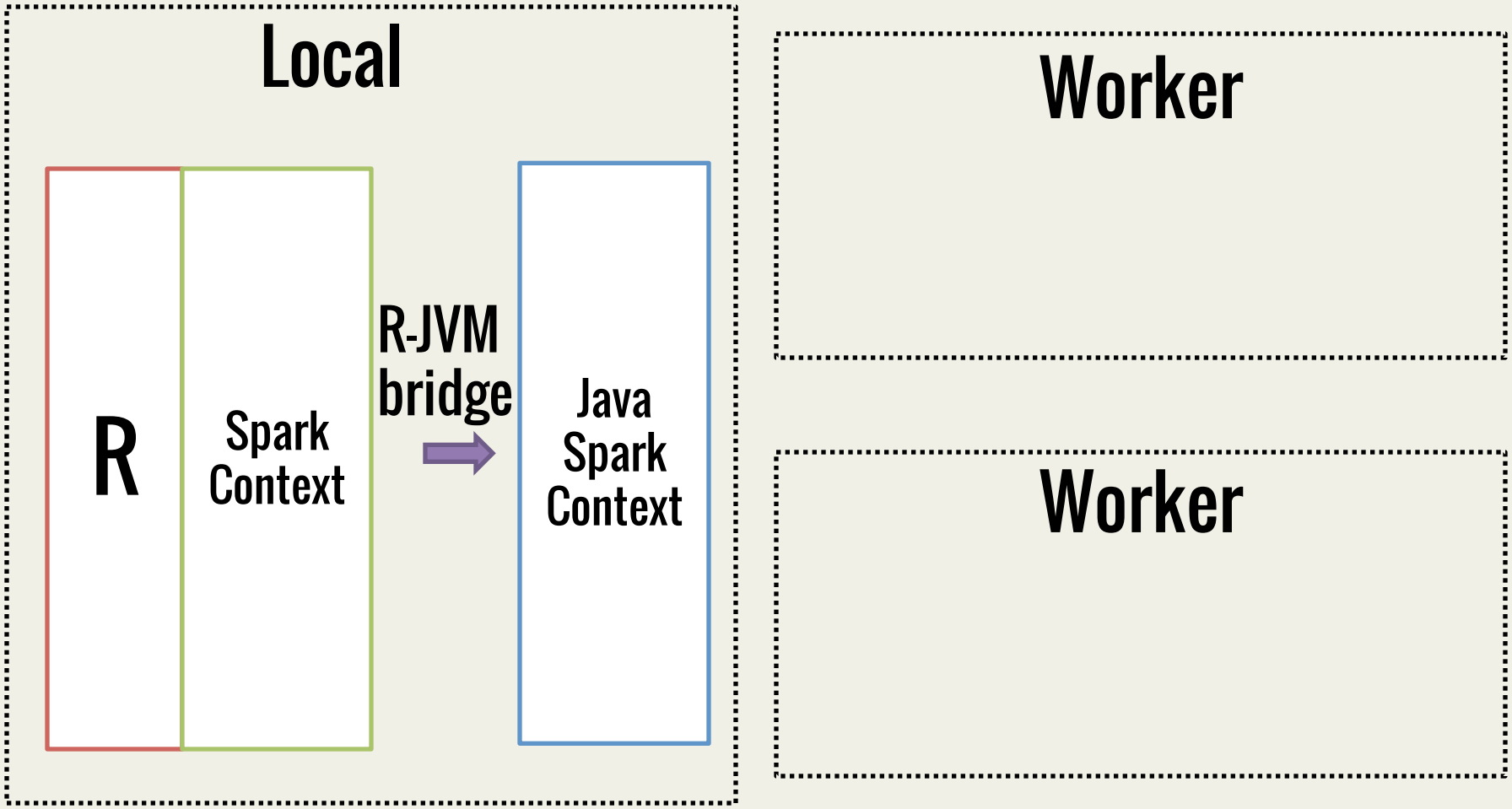
Dataflow



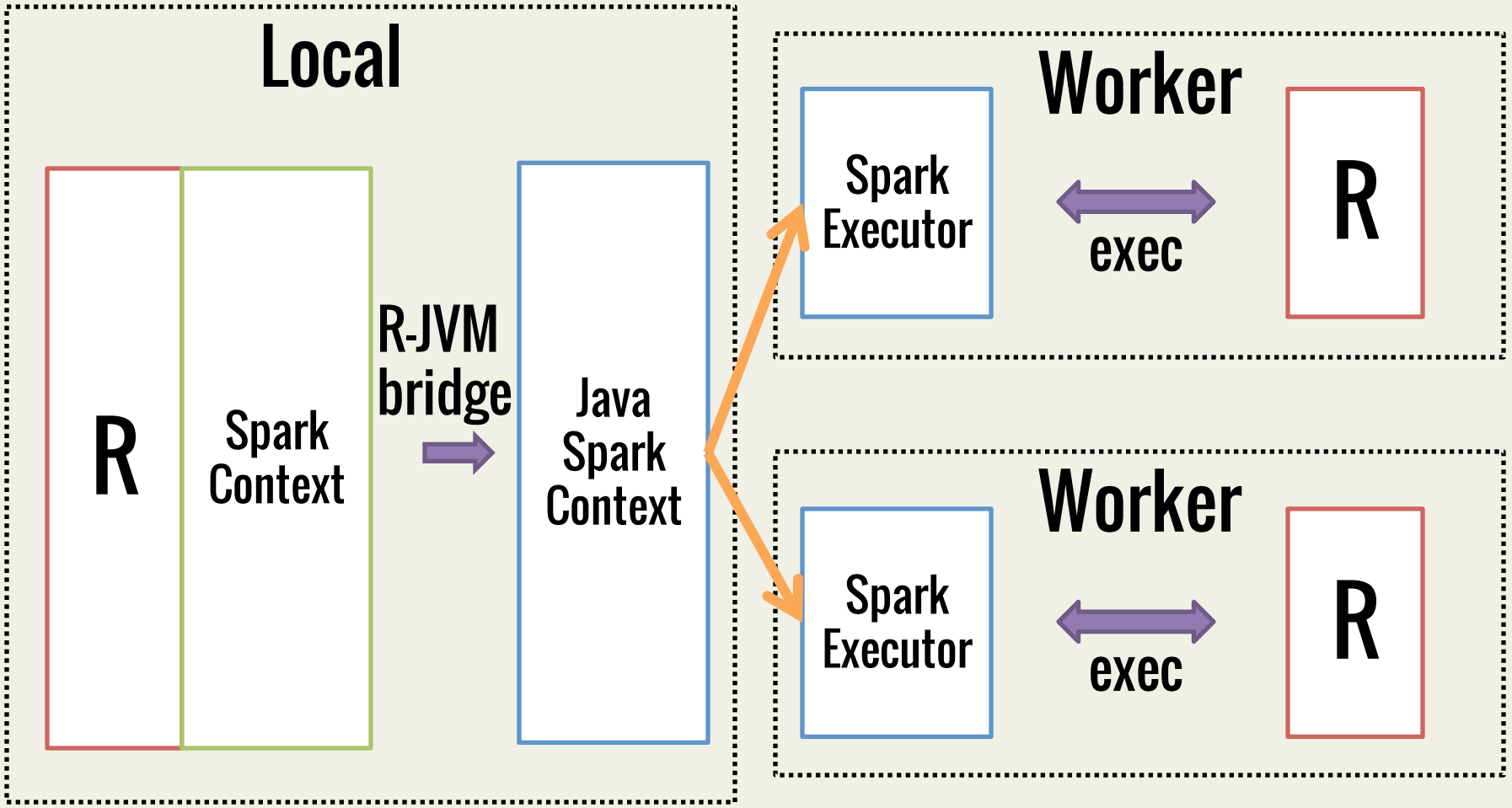
Dataflow



Dataflow


Local
R
Spark Context
R-JVM bridge
Java Spark Context
Worker
Worker

Dataflow


Local
R
Spark Context
R-JVM bridge
Java Spark Context
Worker
Spark Executor
exec
R
Worker
Spark Executor
exec
R

SparkR DataFrames

Need for DataFrames

Structured Data Processing

Read in CSV, JSON, JDBC etc.

Data source for Machine Learning

```
glm(a ~ b + c, data = df)
```

Functional transformations not intuitive

Spark SQL

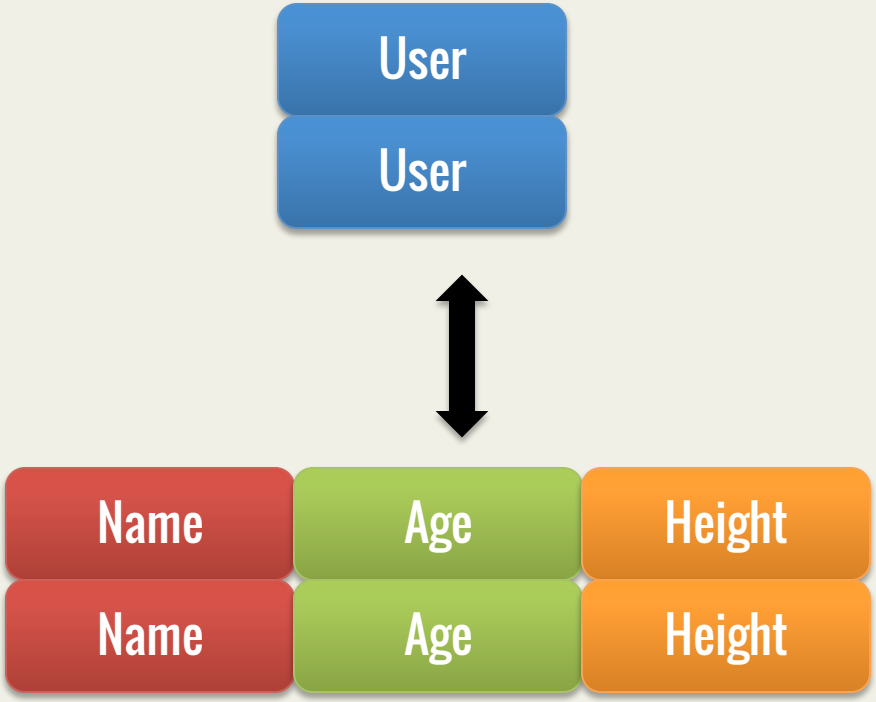
Imposes a schema on RDDs

Query Optimizer, Code Gen

Rich DataSources API

>> Hive, Parquet, JDBC, JSON

~~SchemaRDDs~~ DataFrames!



SparkR DataFrame Methods

Filter – Select some rows

```
filter(df, df$col1 > 0)
```

Project – Select some columns

```
df$col1 or df[“col”]
```

SparkR DataFrame Methods

Filter – Select some rows

Project – Select some columns

Aggregate – Group and Summarize data

```
groupDF <- groupBy(df, df$col1)

agg(groupDF, sum(groupDF$col2), max(groupDF$col3))
```

Sort – Sort data by a particular column

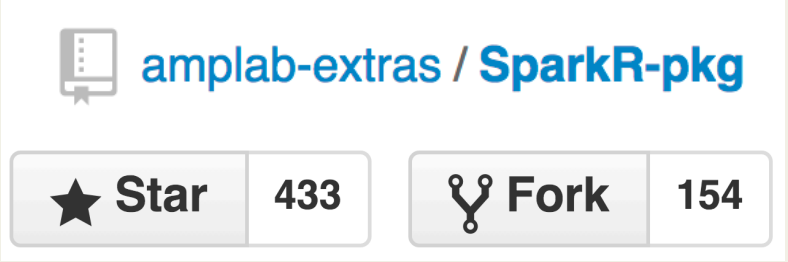
```
sortDF(df, asc(df$col1))
```

Demo

Developer Community

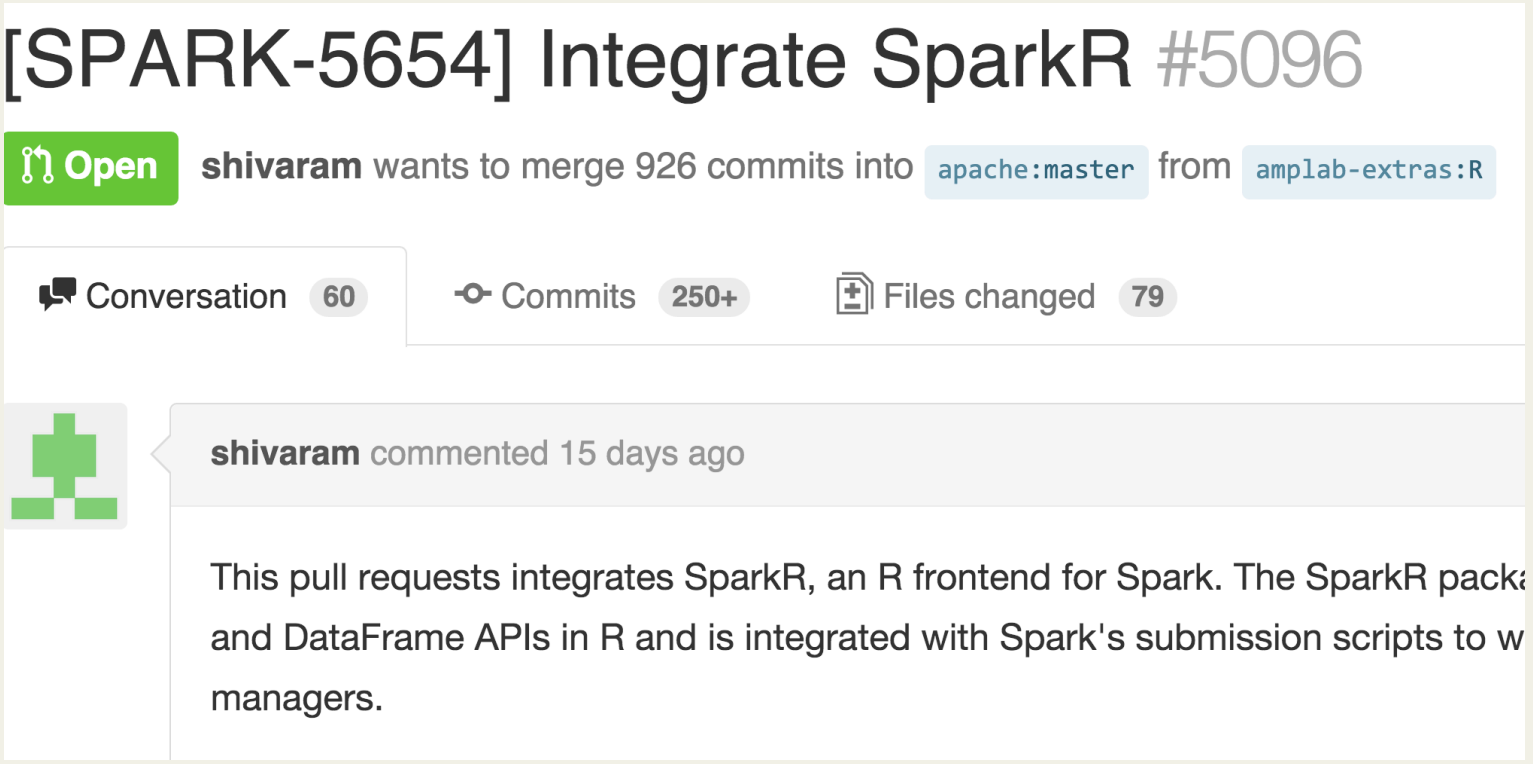
Originated in AMPLab

19 contributors
AMPLab, Alteryx,
Databricks, Intel



Merged with Spark !

Part of Apache Spark 1.4



Coming Soon: ML Pipelines

High-level APIs to do Machine learning

Example: glm, kmeans

Pipelines with featurizers, learning

Tokenize → TF-IDF → LogisticRegression

Extended models, summary methods

Coming Soon

APIs for Streaming, Time series analysis

Distributed matrix operations

<Your SparkR use case ?>

SparkR

RDD → distributed lists

Re-use existing packages

Distributed DataFrames

Combine scalability & utility

Shivaram Venkataraman shivaram@cs.berkeley.edu