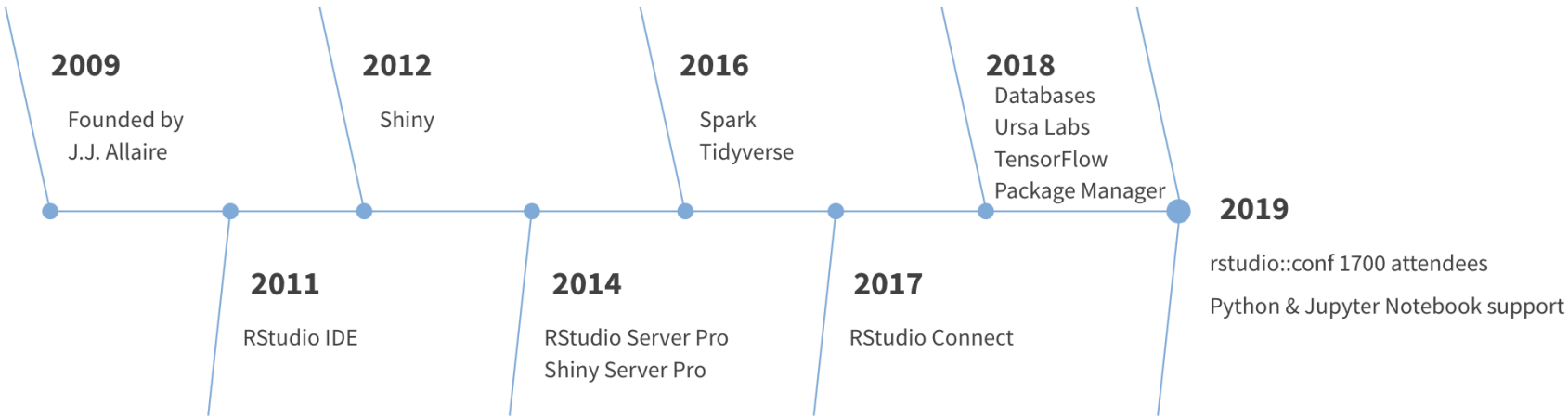
LINUX FOUNDATION AND SPARKLYR

JAVIER LURASCHI, RSTUDIO

OVERVIEW

ABOUT RSTUDIO



RSTUDIO'S MULTIVERSE TEAM

Authors of R packages to support Apache Spark, TensorFlow and MLflow.

Daniel Falbel
@dfalbel

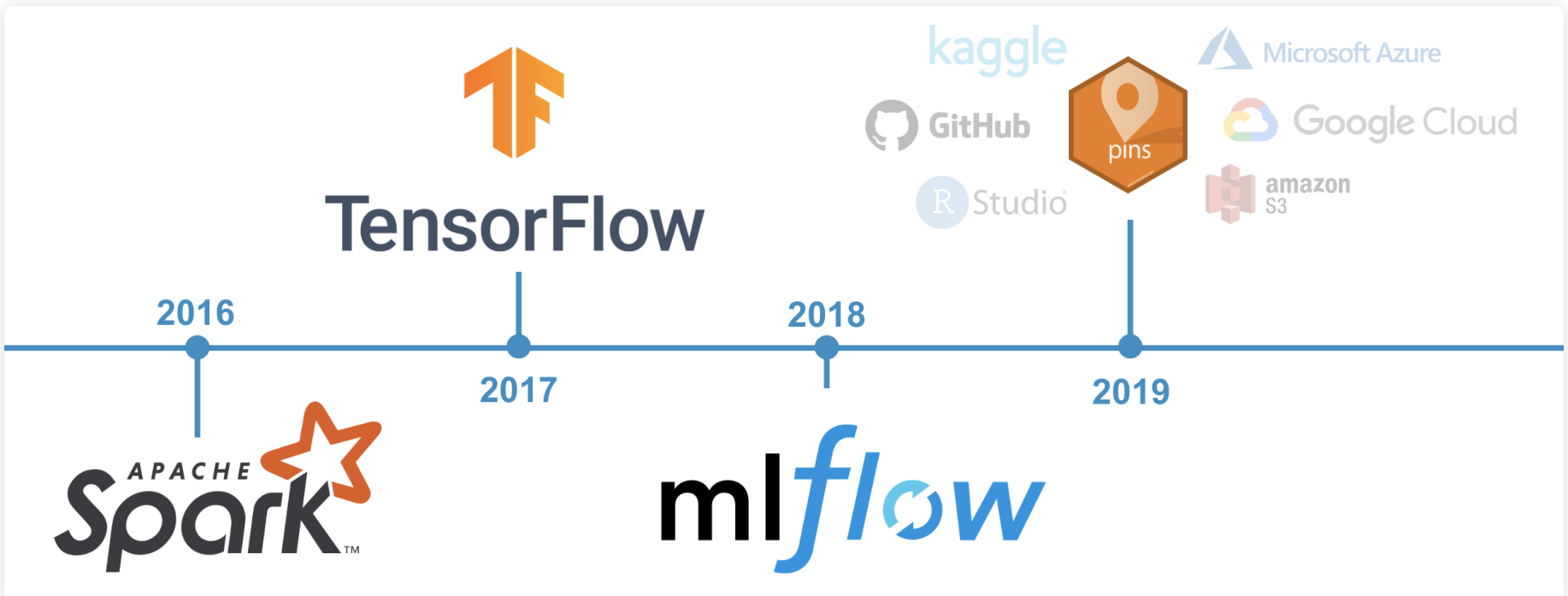
Sigrid Keydana
@zkajdan

Kevin Kuo
@kevinykuo

Javier Luraschi
@javierluraschi

MULTIVERSE TIMELINE

The multiverse team focuses on bringing relevant machine learning technologies to R users to empower and simplify data science workflows.



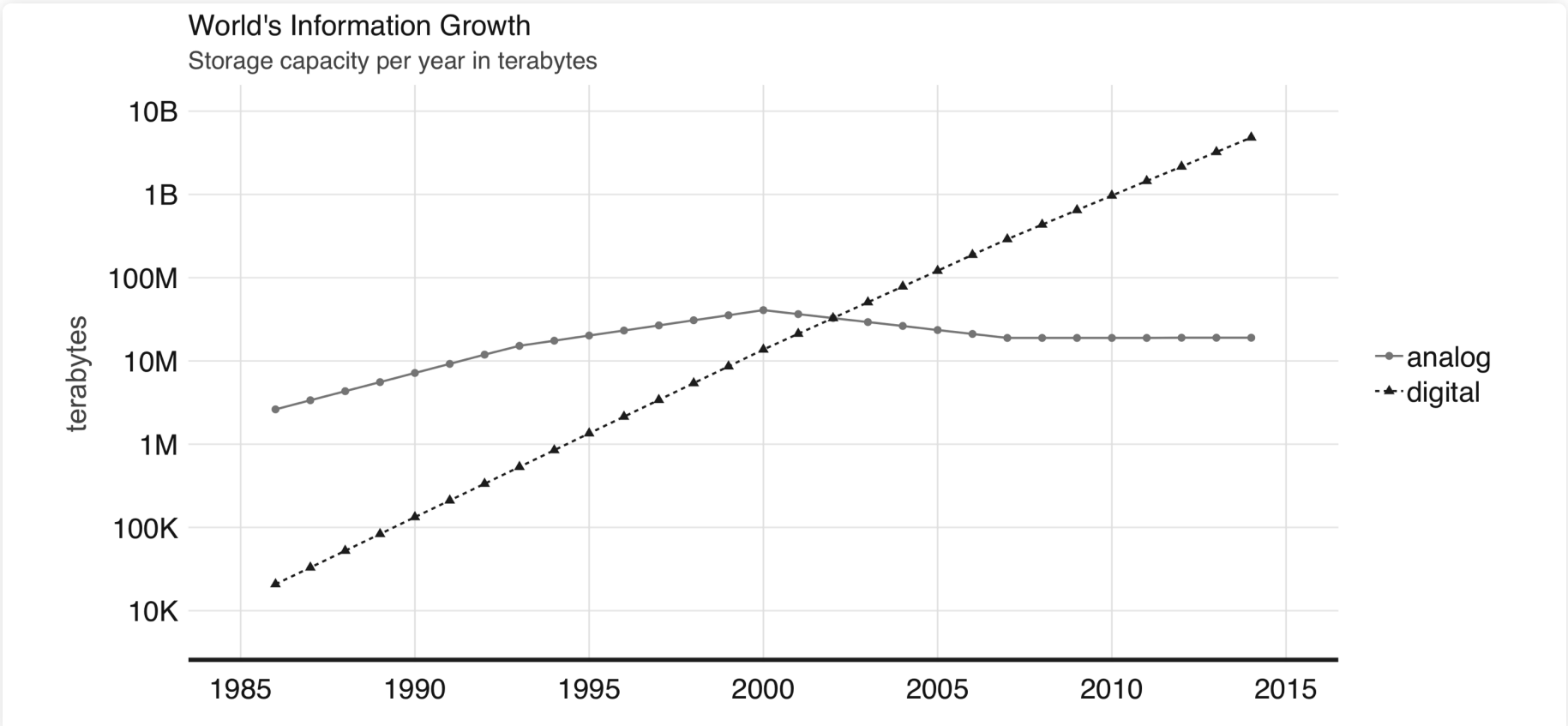
WHAT IS SPARK?

"Apache Spark™ is a unified analytics engine for large-scale data processing."

- **Unified**: Spark supports many libraries, clusters technologies and storage systems.
- **Analytics**: Analytics is the discovery and interpretation of data to produce and communicate information.
- **Engine**: Spark is expected to be efficient and generic.
- **Large-Scale**: One can interpret large-scale as cluster-scale, a set of connected computers working together.

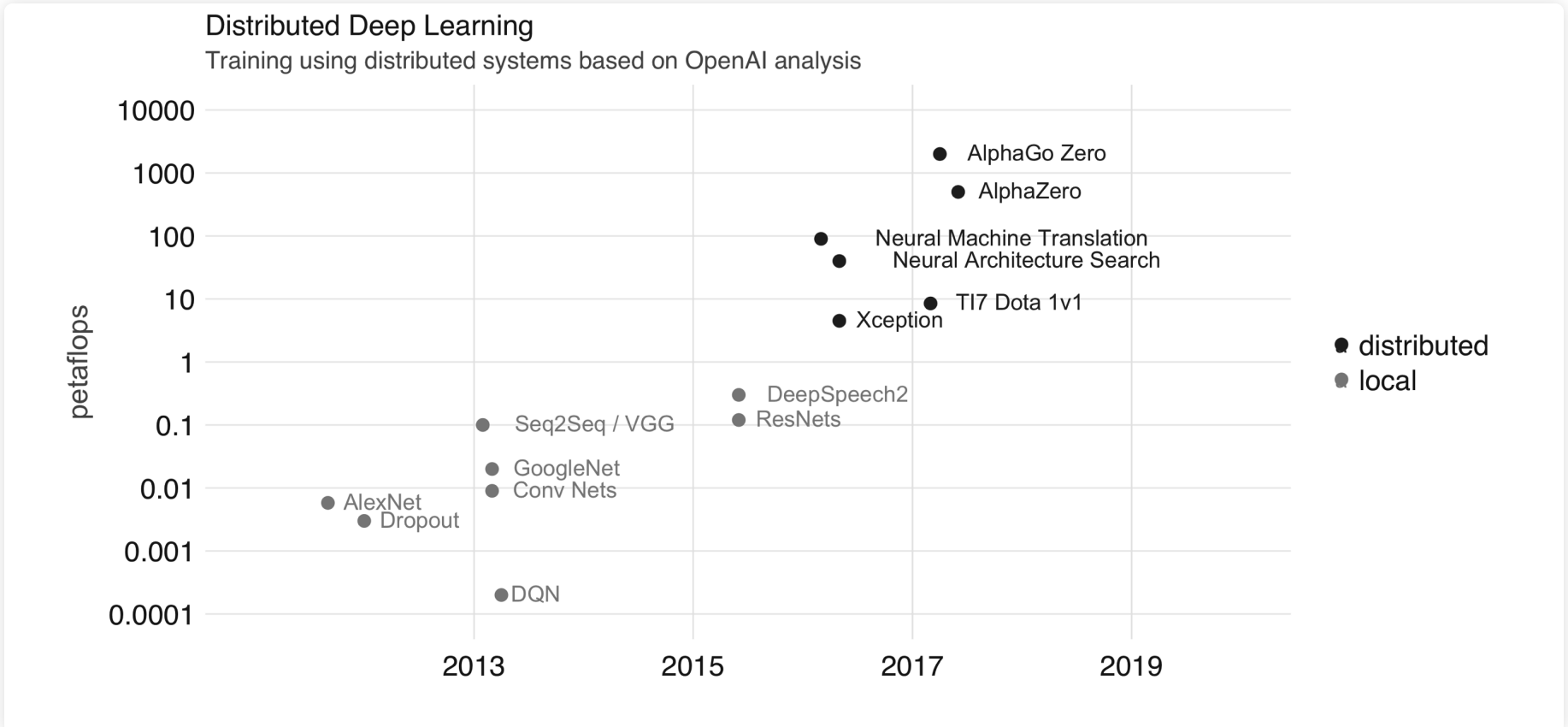
WHY SPARK?

Information grows at exponential rates.



WHAT'S NEXT?

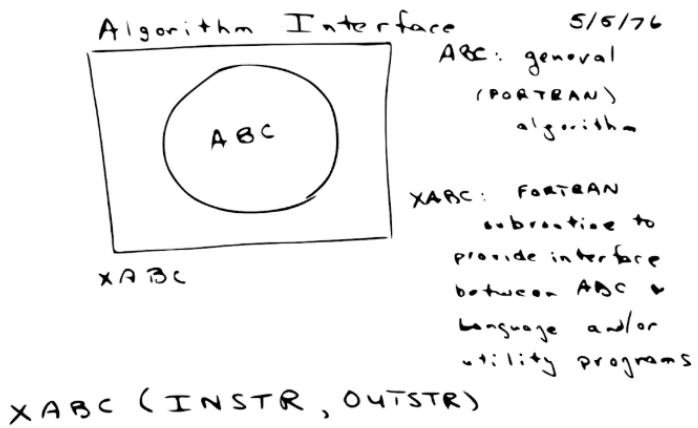
We see Spark supporting multiple projects: TensorFlow, MLflow, Tuning, etc.



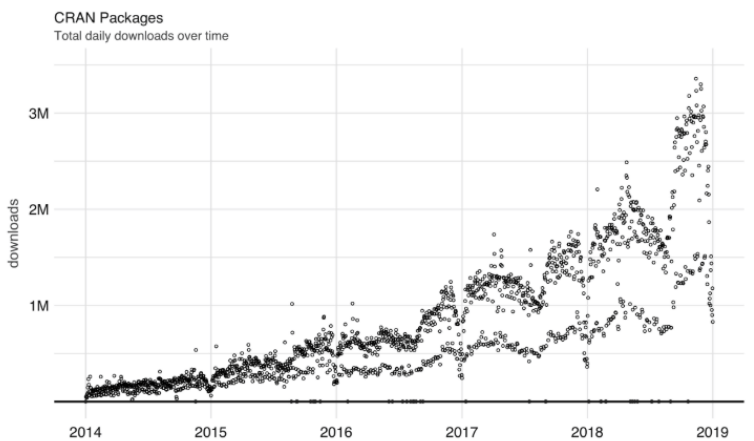
WHY R?

Josh Wills
@josh_wills

Data Scientist (n.): Person who is better at statistics than any software engineer and better at software engineering than any statistician.



"R is a **programming language** and free software environment for **statistical computing** and graphics."



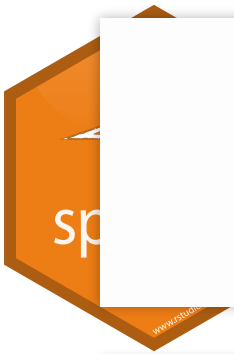
MODERN R

The tidyverse is an opinionated collection of R packages designed for data science. All packages share an underlying design philosophy, grammar, and data structures.

```r
library(tidyverse)
library(nycflights13)

flights %>%
  group_by(month, day) %>%
  summarise(count = n(), avg_delay = mean(dep_delay, na.rm = TRUE)) %>%
  filter(count > 1000)
```

SPARK AND R

In an ideal world, all R packages work with Spark, like magic. Such is the case for `dplyr` and `sparklyr`.



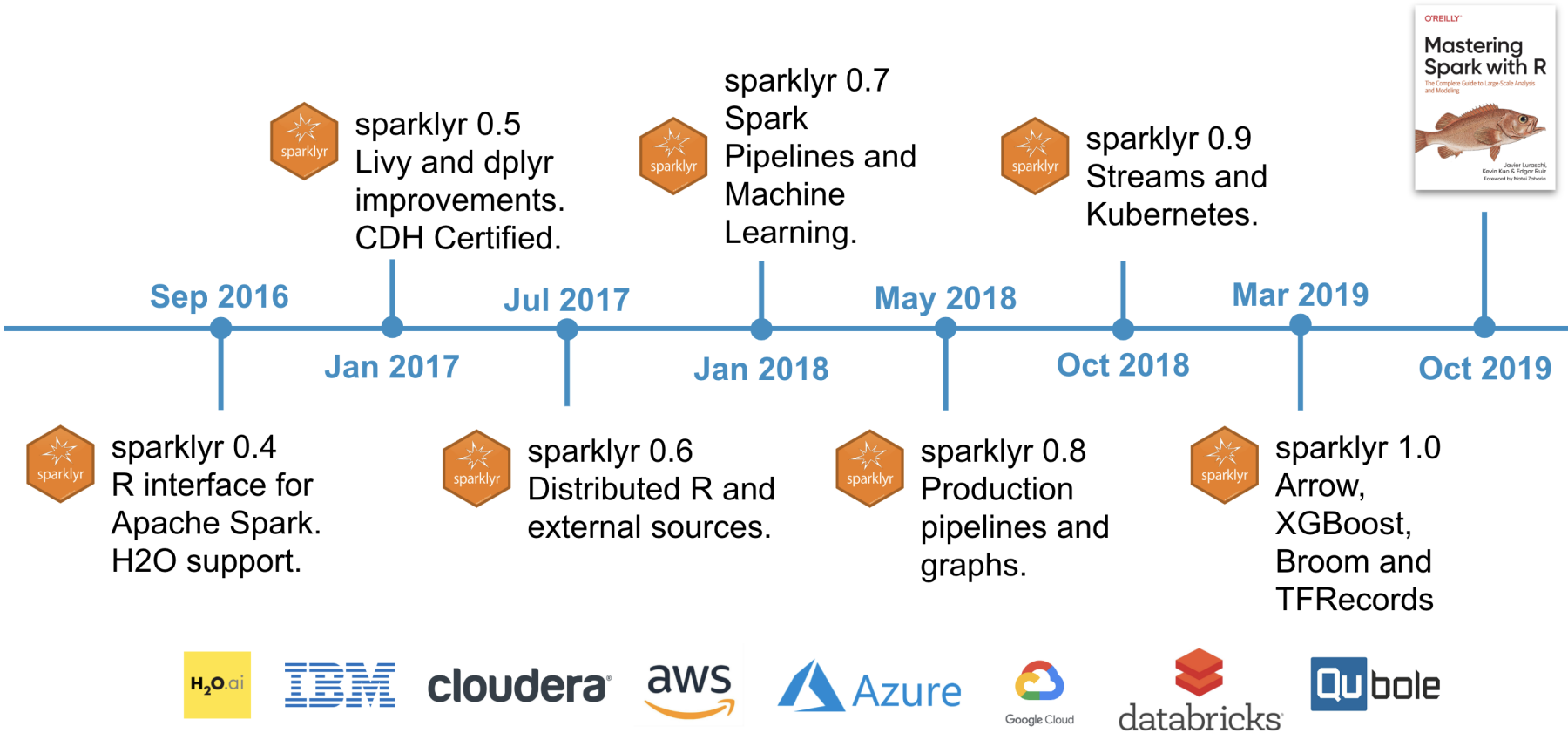
```r
library(sparklyr)
library(nycflights13)

sc <- spark_connect(master = "local|yarn|mesos|spark|livy")
flights <- copy_to(sc, flights)
```

```r
library(tidyverse)
library(nycflights13)

flights %>%
  group_by(month, day) %>%
  summarise(count = n(), avg_delay = mean(dep_delay, na.rm = TRUE)) %>%
  filter(count > 1000)
```

TIMELINE

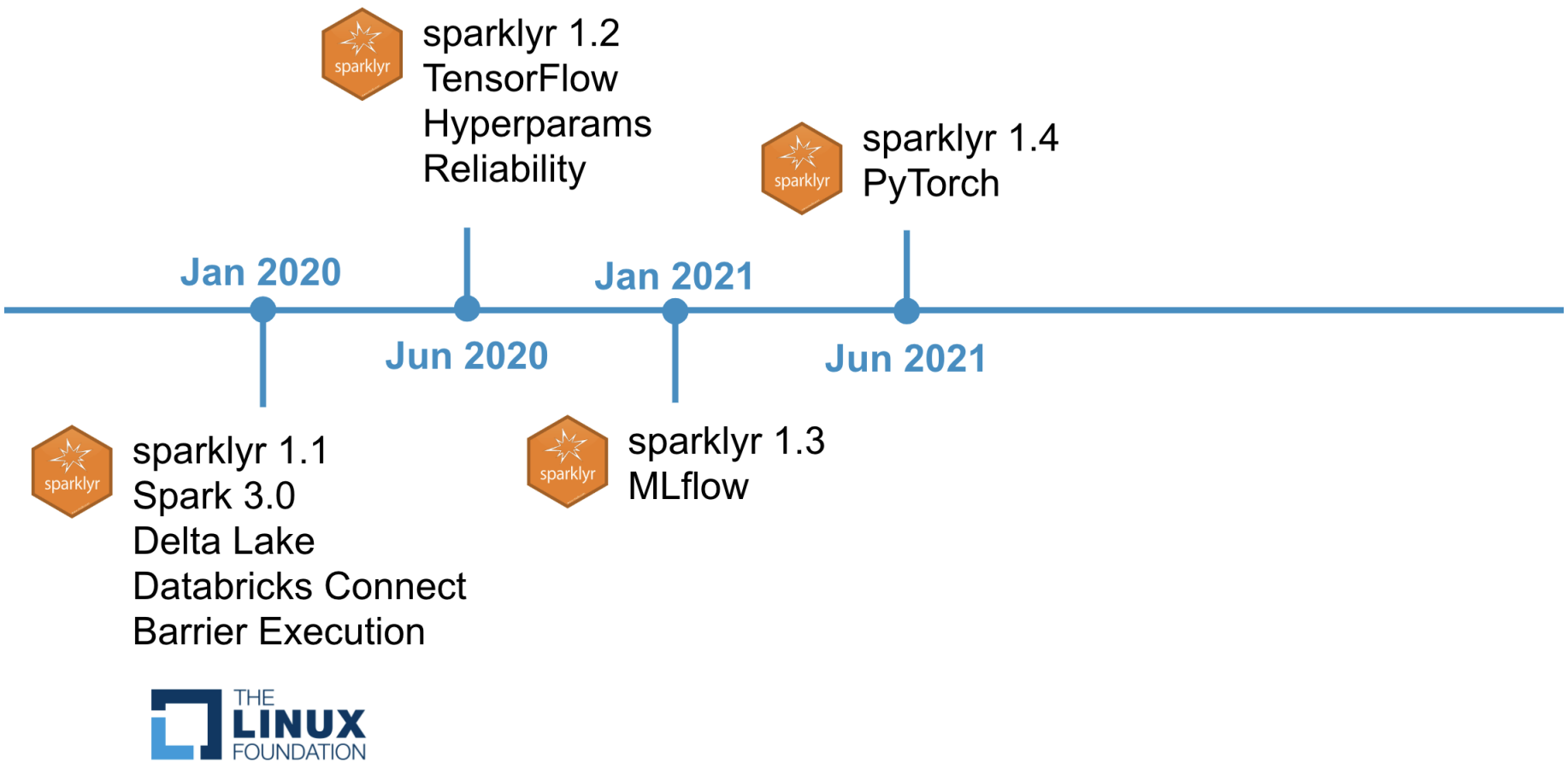
2016-2019

Timeline from launch to sparklyr 1.0.



BEYOND 2020

Aspirational direction beyond 2020.



THE LINUX FOUNDATION

USE CASES

SPARKLYR: R INTERFACE FOR APACHE SPARK

```r
spark_install()                                  # Install Apache Spark
sc <- spark_connect(master = "local")            # Connect to Spark cluster
```

```r
cars_tbl <- spark_read_csv(sc, "cars", "input/")     # Read data in Spark

summarize(cars_tbl, n = n())                         # Count records with dplyr
dbGetQuery(sc, "SELECT count(*) FROM cars")          # Count records with DBI
```

```r
ml_linear_regression(cars_tbl, mpg ~ wt + cyl)       # Perform linear regression

ml_pipeline(sc) %>%                                  # Define Spark pipeline
  ft_r_formula(mpg ~ wt + cyl) %>%                   # Add formula transformation
  ml_linear_regression()                             # Add model to pipeline
```

```r
spark_context(sc) %>% invoke("version")              # Extend sparklyr with Scala
```

```r
spark_apply(cars_tbl, nrow)                          # Extend sparklyr with R
```

```r
stream_read_csv(sc, "input/") %>%                    # Define Spark stream
  filter(mpg > 30) %>%                               # Add dplyr transformation
  stream_write_json("output/")                       # Start processing stream
```

MODELING ALGORITHMS

Some of the many modeling algorithms supported:

Algorithm	Function
Accelerated Failure Time Survival Regression	ml_aft_survival_regression()
Alternating Least Squares Factorization	ml_als()
Bisecting K-Means Clustering	ml_bisecting_kmeans()
Chi-square Hypothesis Testing	ml_chisquare_test()
Correlation Matrix	ml_corr()
Decision Trees	ml_decision_tree ()
Frequent Pattern Mining	ml_fpgrowth()
Gaussian Mixture Clustering	ml_gaussian_mixture()
Generalized Linear Regression	ml_generalized_linear_regression()
Gradient-Boosted Trees	ml_gradient_boosted_trees()
Isotonic Regression	ml_isotonic_regression()
K-Means Clustering	ml_kmeans()
Latent Dirichlet Allocation	ml_lda()
Linear Regression	ml_linear_regression()
Linear Support Vector Machines	ml_linear_svc()
Logistic Regression	ml_logistic_regression()
Multilayer Perceptron	ml_multilayer_perceptron()
Naive-Bayes	ml_naive_bayes()
One vs Rest	ml_one_vs_rest()
Principal Components Analysis	ml_pca()
Random Forests	ml_random_forest()
Survival Regression	ml_survival_regression()

FEATURE ENGINEERING

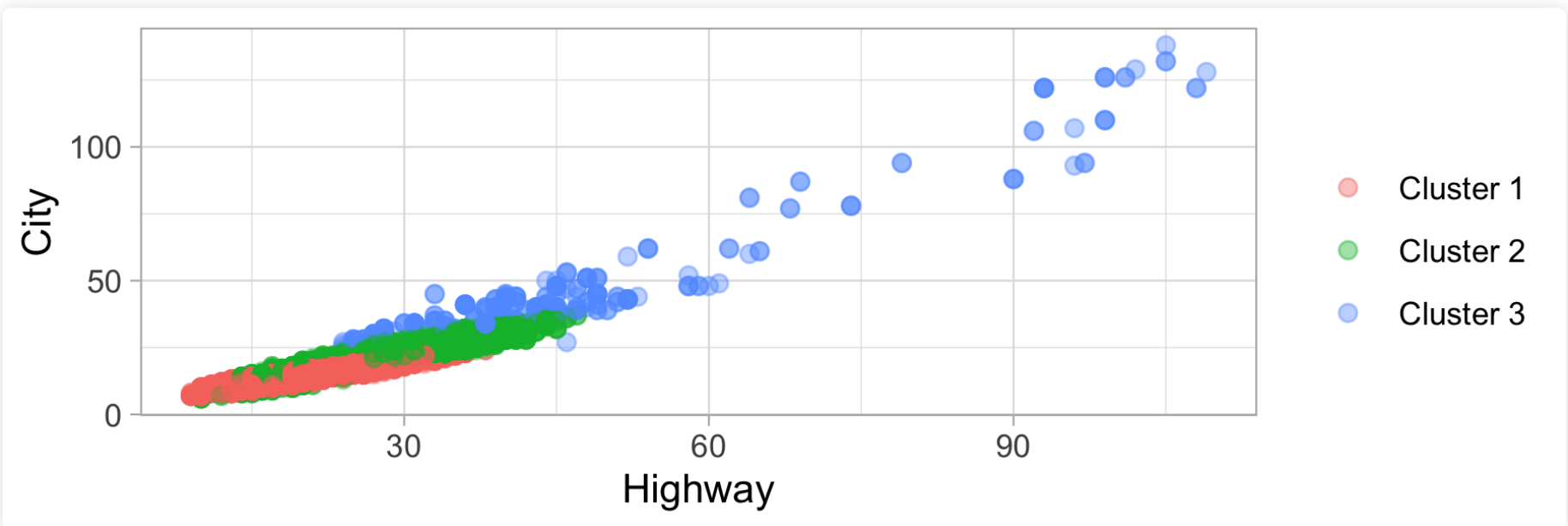
Some of the many feature engineering transformers:

Transformer	Function
Binarizer	ft_binarizer()
Bucketizer	ft_bucketizer()
Chi-Squared Feature Selector	ft_chisq_selector()
Vocabulary from Document Collections	ft_count_vectorizer()
Discrete Cosine Transform	ft_discrete_cosine_transform()
Transformation using dplyr	ft_dplyr_transformer()
Hadamard Product	ft_elementwise_product()
Feature Hasher	ft_feature_hasher()
Term Frequencies using Hashing	export(ft_hashing_tf)
Inverse Document Frequency	ft_idf()
Imputation for Missing Values	export(ft_imputer)
Index to String	ft_index_to_string()
Feature Interaction Transform	ft_interaction()
Rescale to [-1, 1] Range	ft_max_abs_scaler()
Rescale to [min, max] Range	ft_min_max_scaler()
Locality Sensitive Hashing	ft_minhash_lsh()
Converts to n-grams	ft_ngram()
Normalize using the given P-Norm	ft_normalizer()
One-Hot Encoding	ft_one_hot_encoder()
Feature Expansion in Polynomial Space	ft_polynomial_expansion()
Maps to Binned Categorical Features	ft_quantile_discretizer()
SQL Transformation	ft_sql_transformer()
Standardizes Features using Corrected STD	ft_standard_scaler()
Filters out Stop Words	ft_stop_words_remover()
Map to Label Indices	ft_string_indexer()

Transformer	Function
Splits by White Spaces	ft_tokenizer()
Transform Word into Code	ft_word2vec()

GAUSSIAN MIXTURE CLUSTERING

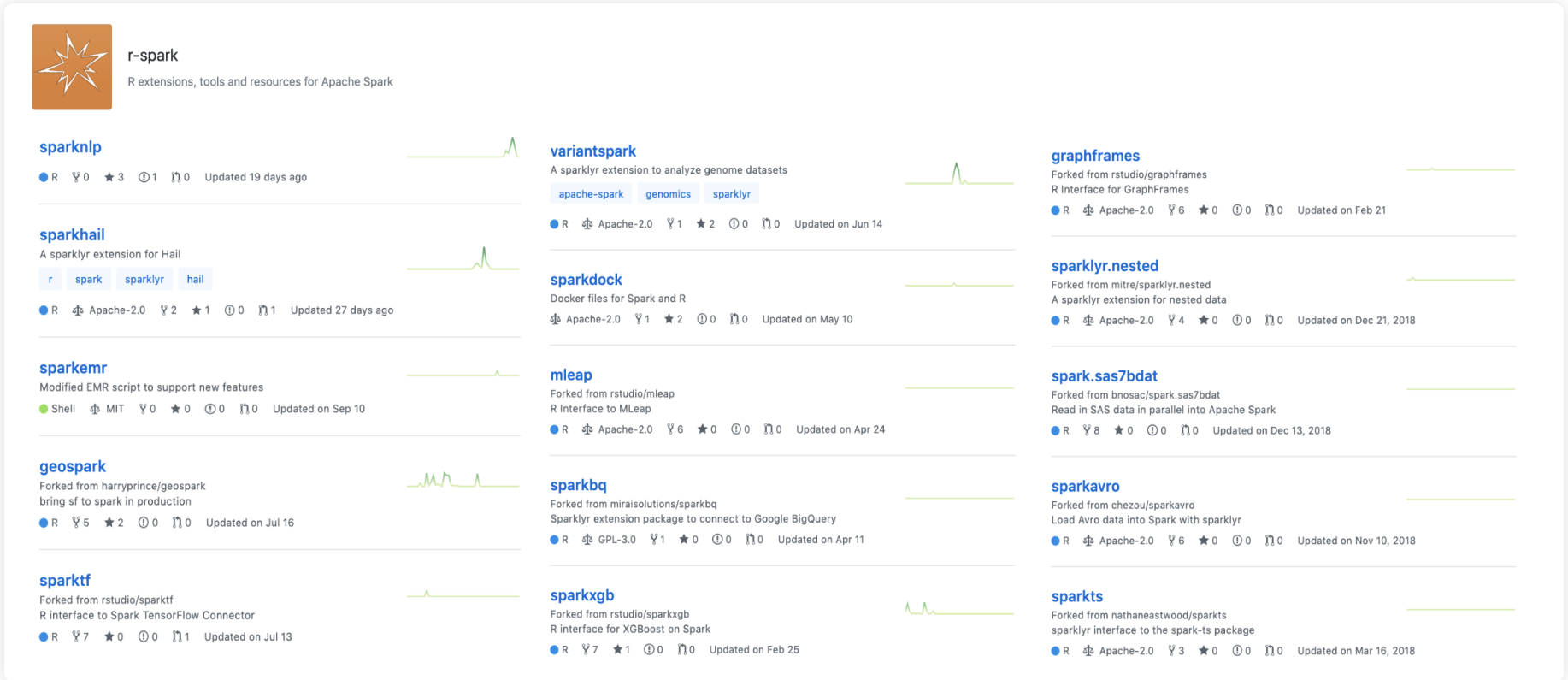
```
predictions <- copy_to(sc, fueleconomy::vehicles) %>%
  ml_gaussian_mixture(~ hwy + cty, k = 3) %>%
  ml_predict() %>% collect()

predictions %>%
  ggplot(aes(hwy, cty)) +
  geom_point(aes(hwy, cty, col = factor(prediction)), size = 2, alpha = 0.4) +
  scale_color_discrete(name = "", labels = paste("Cluster", 1:3)) +
  labs(x = "Highway", y = "City") + theme_light()
```



COMMUNITY

EXTENSIONS

About ~20 community extensions developed for sparklyr in the r-spark repo.



r-spark

R extensions, tools and resources for Apache Spark

sparknlp

R 0 ★ 3 1 0 Updated 19 days ago

sparkhail

A sparklyr extension for Hail

r spark sparklyr hail

R Apache-2.0 2 ★ 1 0 1 Updated 27 days ago

sparkemr

Modified EMR script to support new features

Shell MIT 0 ★ 0 0 0 Updated on Sep 10

geospark

Forked from harryprince/geospark
bring sf to spark in production

R 5 ★ 2 0 0 Updated on Jul 16

sparktf

Forked from rstudio/sparktf
R interface to Spark TensorFlow Connector

R 7 ★ 0 0 1 Updated on Jul 13

variantspark

A sparklyr extension to analyze genome datasets

apache-spark genomics sparklyr

R Apache-2.0 1 ★ 2 0 0 Updated on Jun 14

sparkdock

Docker files for Spark and R

Apache-2.0 1 ★ 2 0 0 Updated on May 10

mleap

Forked from rstudio/mleap
R Interface to MLeap

R Apache-2.0 6 ★ 0 0 0 Updated on Apr 24

sparkbq

Forked from miraisolutions/sparkbq
Sparklyr extension package to connect to Google BigQuery

R GPL-3.0 1 ★ 0 0 0 Updated on Apr 11

sparkxgb

Forked from rstudio/sparkxgb
R interface for XGBoost on Spark

R 7 ★ 1 0 0 Updated on Feb 25

graphframes

Forked from rstudio/graphframes
R Interface for GraphFrames

R Apache-2.0 6 ★ 0 0 0 Updated on Feb 21

sparklyr.nested

Forked from mitre/sparklyr.nested
A sparklyr extension for nested data

R Apache-2.0 4 ★ 0 0 0 Updated on Dec 21, 2018

spark.sas7bdat

Forked from bnosac/spark.sas7bdat
Read in SAS data in parallel into Apache Spark

R 8 ★ 0 0 0 Updated on Dec 13, 2018

sparkavro

Forked from chezou/sparkavro
Load Avro data into Spark with sparklyr

R Apache-2.0 6 ★ 0 0 0 Updated on Nov 10, 2018

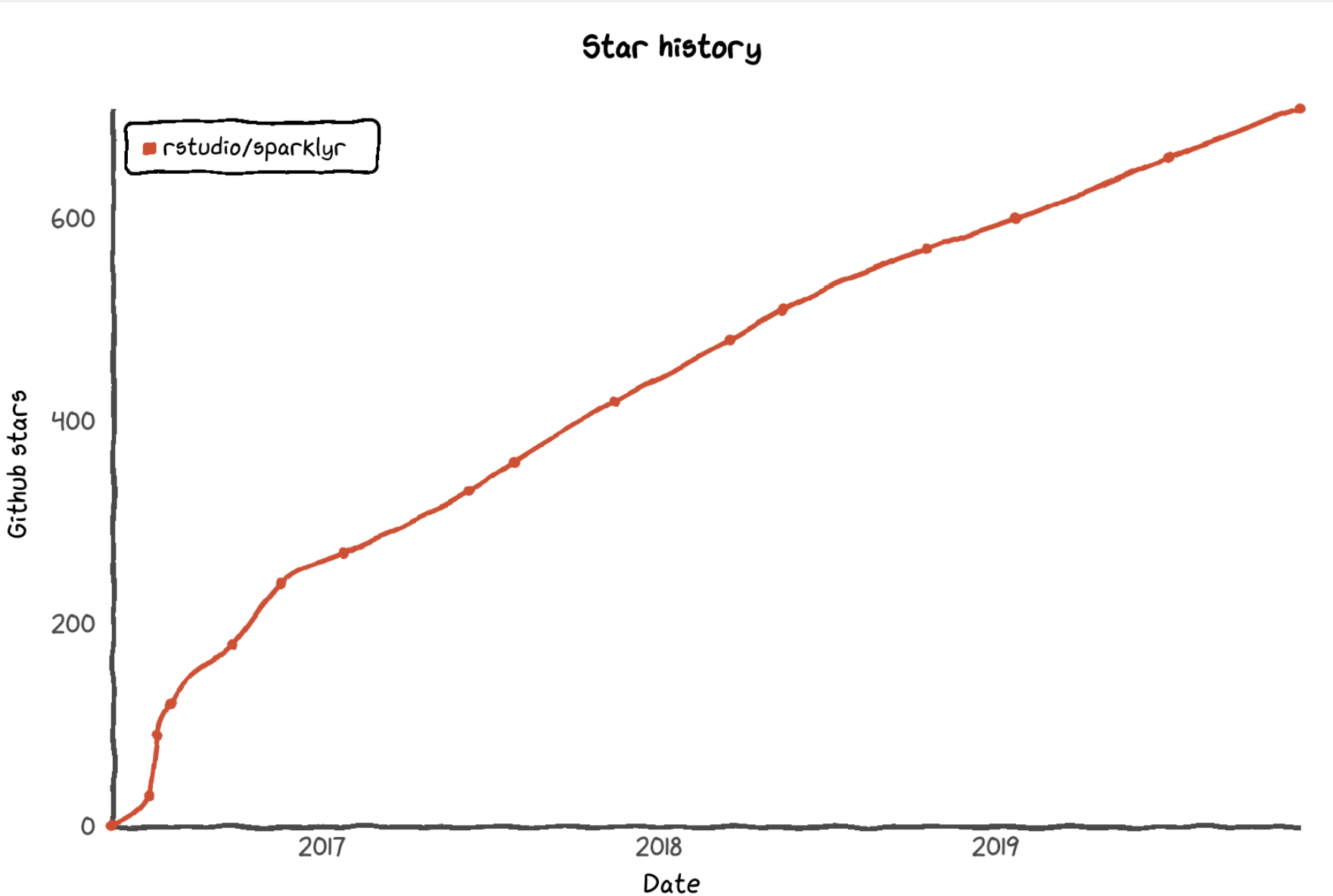
sparkts

Forked from nathaneastwood/sparkts
sparklyr interface to the spark-ts package

R Apache-2.0 3 ★ 0 0 0 Updated on Mar 16, 2018

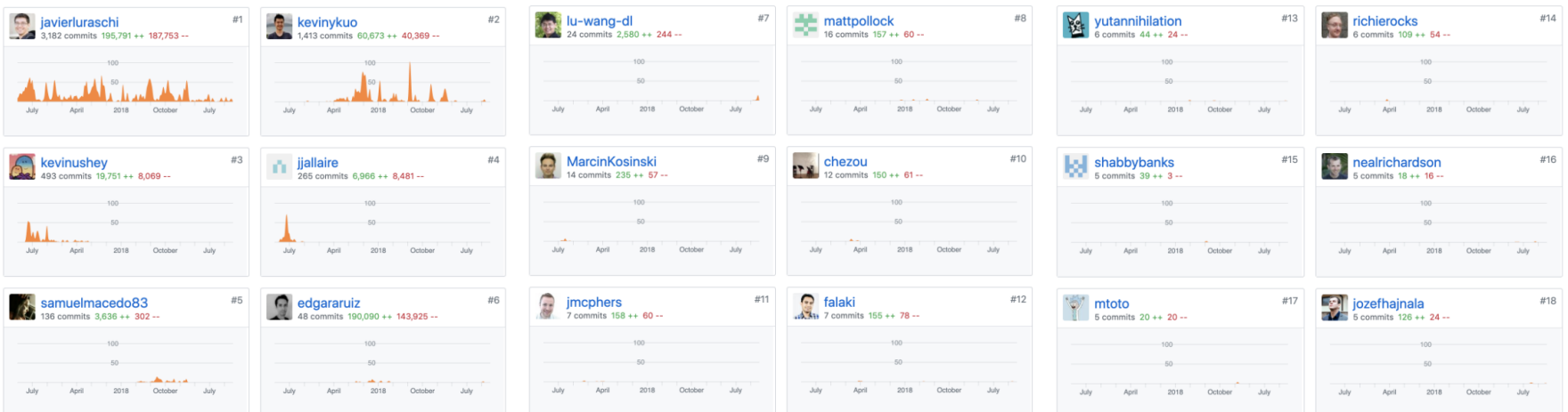
GITHUB STARS

Steady growth of GitHub stars over time.


Star history
rstudio/sparklyr
600
400
200
0
Github stars
2017
2018
2019
Date

PAST CONTRIBUTORS

Over 50+ contributors to the sparklyr repo.



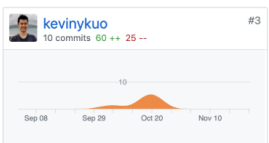
CURRENT CONTRIBUTORS

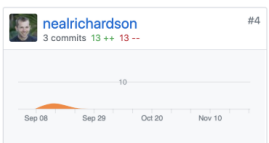
6+ organizations contributing in the last 3 months.

RStudio

databricks

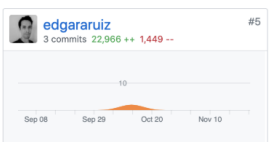
RStudio





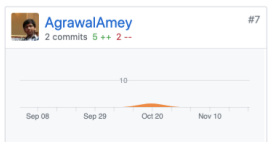
APACHE ARROW

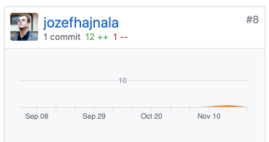
RStudio



databricks

Qubole





Augustus Consulting

Tokyo.R

databricks

TECHNICAL

CRAN RELEASES

Releasing to CRAN about every two months with major releases twice a year.

sparklyr: R Interface to Apache Spark

R interface to Apache Spark, a fast and general engine for big data processing, see <http://spark.apache.org>. This package supports connecting to local and remote Apache Spark clusters, provides a 'dplyr' compatible back-end, and provides an interface to Spark's built-in machine learning algorithms.

Version:	1.0.5
Depends:	R (≥ 3.2)
Imports:	assertthat, base64enc, config (≥ 0.2), DBI (≥ 0.6-1), dplyr (≥ 0.7.2), dbplyr (≥ 1.1.0), digest, forge, generics, httr (≥ 1.2.1), jsonlite (≥ 1.4), methods, openssl (≥ 0.8), purrr, r2d3, rappdirs, rlang (≥ 0.1.4), rprojroot, rstudioapi (≥ 0.10), tibble, tidyr, withr, xml2, ellipsis (≥ 0.1.0)
Suggests:	broom, ggplot2, janeaustenr, Lahman, mlbench, nnet, nycflights13, R6, RCurl, reshape2, shiny (≥ 1.0.1), testthat
Published:	2019-11-14
Author:	Javier Luraschi [aut, cre], Kevin Kuo [aut], Kevin Ushey [aut], JJ Allaire [aut], Samuel Macedo [ctb], RStudio [cph], The Apache Software Foundation [aut, cph]
Maintainer:	Javier Luraschi <javier at rstudio.com>
BugReports:	https://github.com/rstudio/sparklyr/issues
License:	Apache License 2.0 | file LICENSE
URL:	http://spark.rstudio.com
NeedsCompilation:	no
SystemRequirements:	Spark: 1.6.x or 2.x
Materials:	README NEWS
In views:	ModelDeployment
CRAN checks:	sparklyr results

Downloads:

Reference manual:	sparklyr.pdf
Package source:	sparklyr_1.0.5.tar.gz
Windows binaries:	r-devel: sparklyr_1.0.5.zip, r-devel-gcc8: sparklyr_1.0.5.zip, r-release: sparklyr_1.0.5.zip, r-oldrel: sparklyr_1.0.5.zip
OS X binaries:	r-release: sparklyr_1.0.5.tgz, r-oldrel: sparklyr_1.0.5.tgz
Old sources:	sparklyr archive

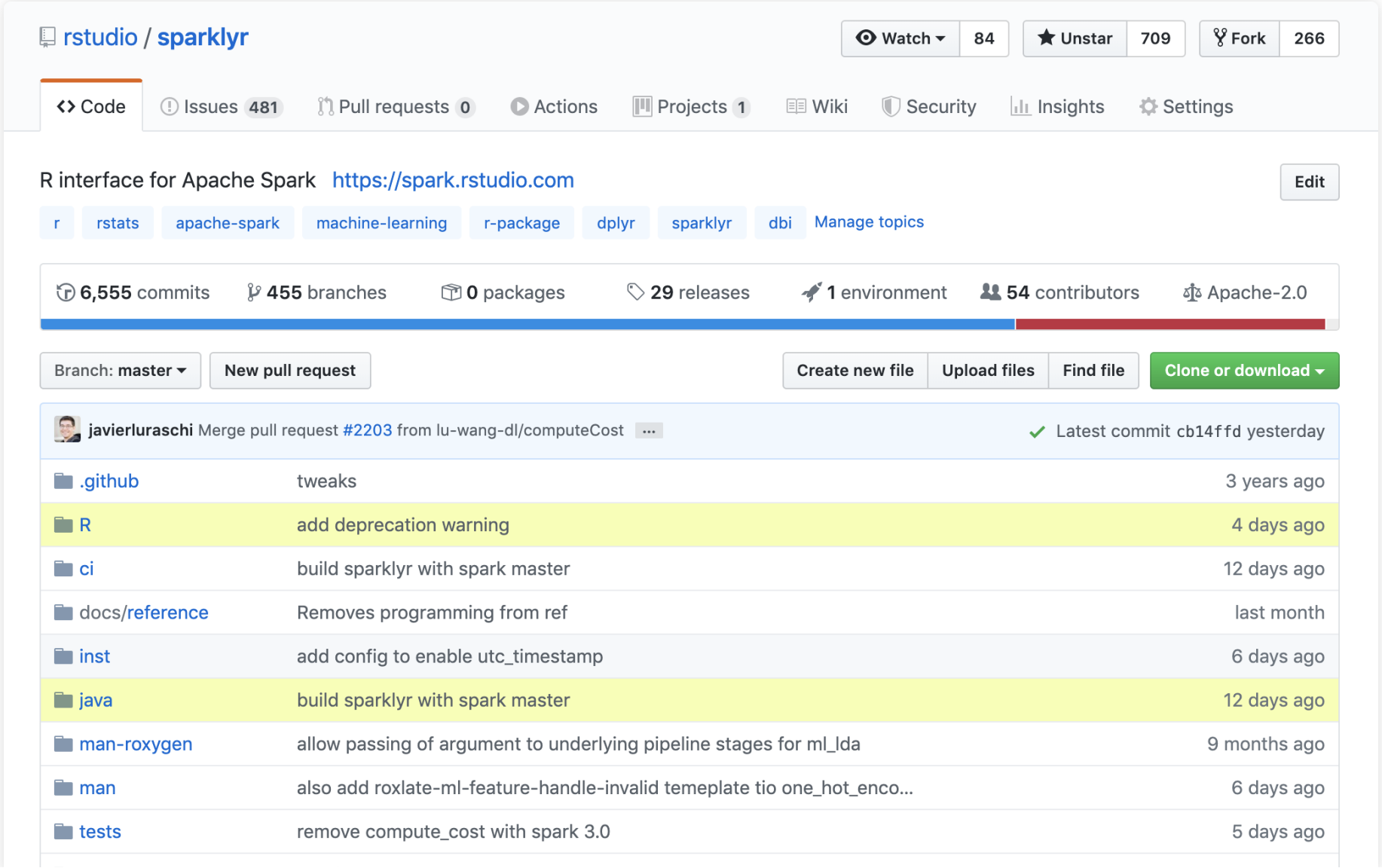
Reverse dependencies:

Reverse imports:	geospark, graphframes, mleap, rsparkling, shinyML, spark.sas7bdat, sparkavro, sparkbq, sparklyr.nested, sparktf, sparkwarc, sparkxgb, variantspark

Name	Last modified	Size	Description
Parent Directory		-	
sparklyr_0.4.tar.gz	2016-09-24 22:40	277K	
sparklyr_0.5.1.tar.gz	2016-12-19 15:56	716K	
sparklyr_0.5.2.tar.gz	2017-02-16 22:40	716K	
sparklyr_0.5.3.tar.gz	2017-03-09 18:09	716K	
sparklyr_0.5.4.tar.gz	2017-04-25 09:24	1.7M	
sparklyr_0.5.5.tar.gz	2017-05-26 08:19	1.7M	
sparklyr_0.5.6.tar.gz	2017-06-10 23:42	1.7M	
sparklyr_0.5.tar.gz	2016-12-18 11:23	716K	
sparklyr_0.6.0.tar.gz	2017-07-29 07:22	2.2M	
sparklyr_0.6.1.tar.gz	2017-08-06 18:35	2.2M	
sparklyr_0.6.2.tar.gz	2017-08-13 07:40	2.2M	
sparklyr_0.6.3.tar.gz	2017-09-19 18:08	2.3M	
sparklyr_0.6.4.tar.gz	2017-11-02 01:57	2.6M	
sparklyr_0.7.0.tar.gz	2018-01-23 09:49	2.9M	
sparklyr_0.8.0.tar.gz	2018-05-01 05:45	3.3M	
sparklyr_0.8.1.tar.gz	2018-05-02 09:59	3.4M	
sparklyr_0.8.2.tar.gz	2018-05-06 08:23	3.4M	
sparklyr_0.8.3.tar.gz	2018-05-12 07:53	3.4M	
sparklyr_0.8.4.tar.gz	2018-05-25 23:39	3.4M	
sparklyr_0.9.1.tar.gz	2018-09-27 07:00	3.6M	
sparklyr_0.9.2.tar.gz	2018-10-17 07:20	3.6M	
sparklyr_0.9.3.tar.gz	2018-11-29 07:00	3.6M	
sparklyr_0.9.4.tar.gz	2019-01-09 07:30	3.6M	
sparklyr_1.0.0.tar.gz	2019-02-25 09:30	3.4M	
sparklyr_1.0.1.tar.gz	2019-05-17 23:30	3.4M	
sparklyr_1.0.2.tar.gz	2019-07-04 08:33	3.4M	
sparklyr_1.0.3.tar.gz	2019-09-15 07:10	3.4M	
sparklyr_1.0.4.tar.gz	2019-10-04 23:40	3.4M	

GITHUB REPO

The sparklyr repo codebase is split into R (client) and Scala (server):



rstudio / sparklyr

Watch 84 | Unstar 709 | Fork 266

Code | Issues 481 | Pull requests 0 | Actions | Projects 1 | Wiki | Security | Insights | Settings

R interface for Apache Spark https://spark.rstudio.com

Edit

r | rstats | apache-spark | machine-learning | r-package | dplyr | sparklyr | dbi | Manage topics

6,555 commits | 455 branches | 0 packages | 29 releases | 1 environment | 54 contributors | Apache-2.0

Branch: master | New pull request | Create new file | Upload files | Find file | Clone or download

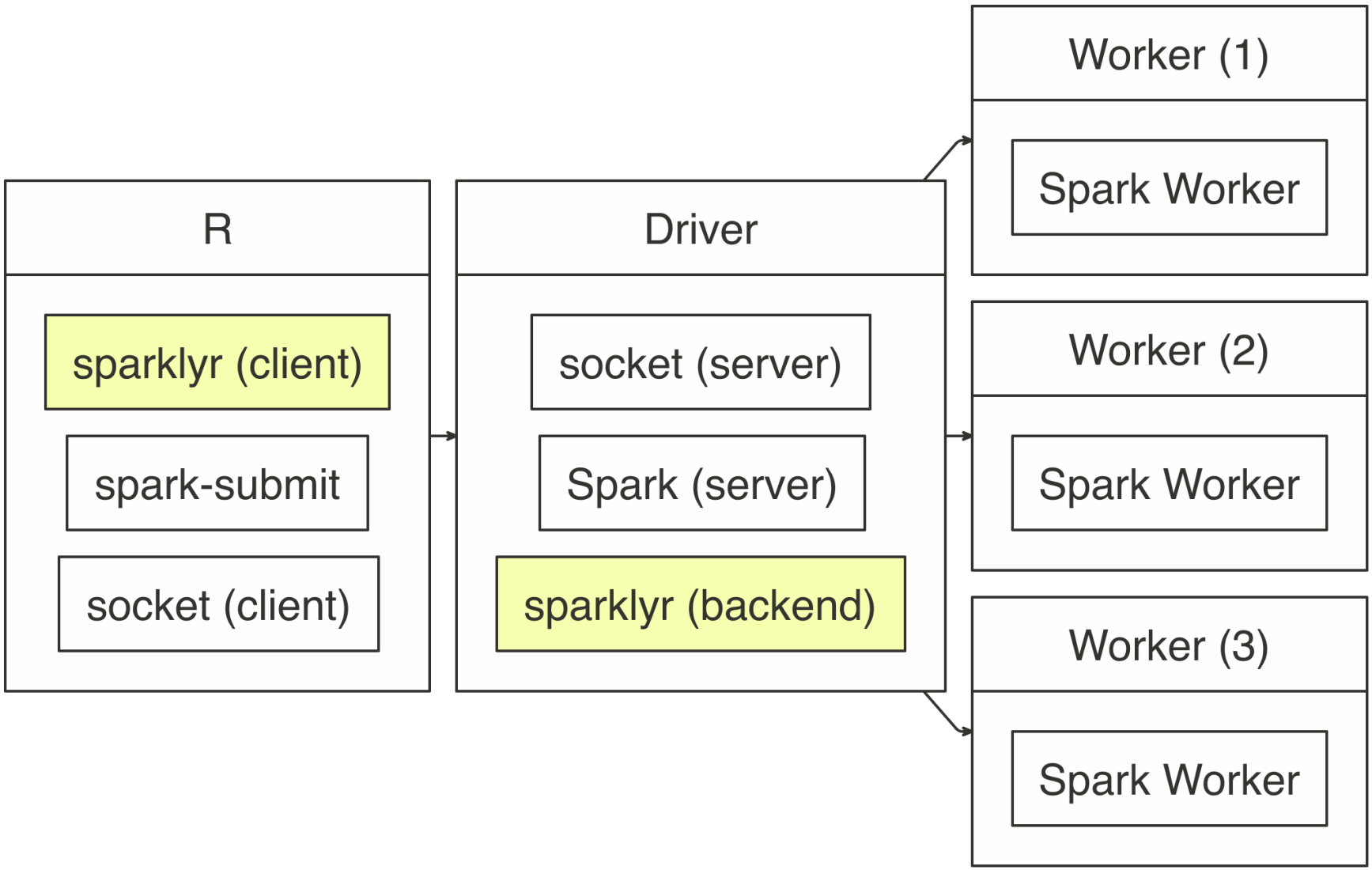
javierluraschi Merge pull request #2203 from lu-wang-dl/computeCost ... Latest commit cb14ffd yesterday

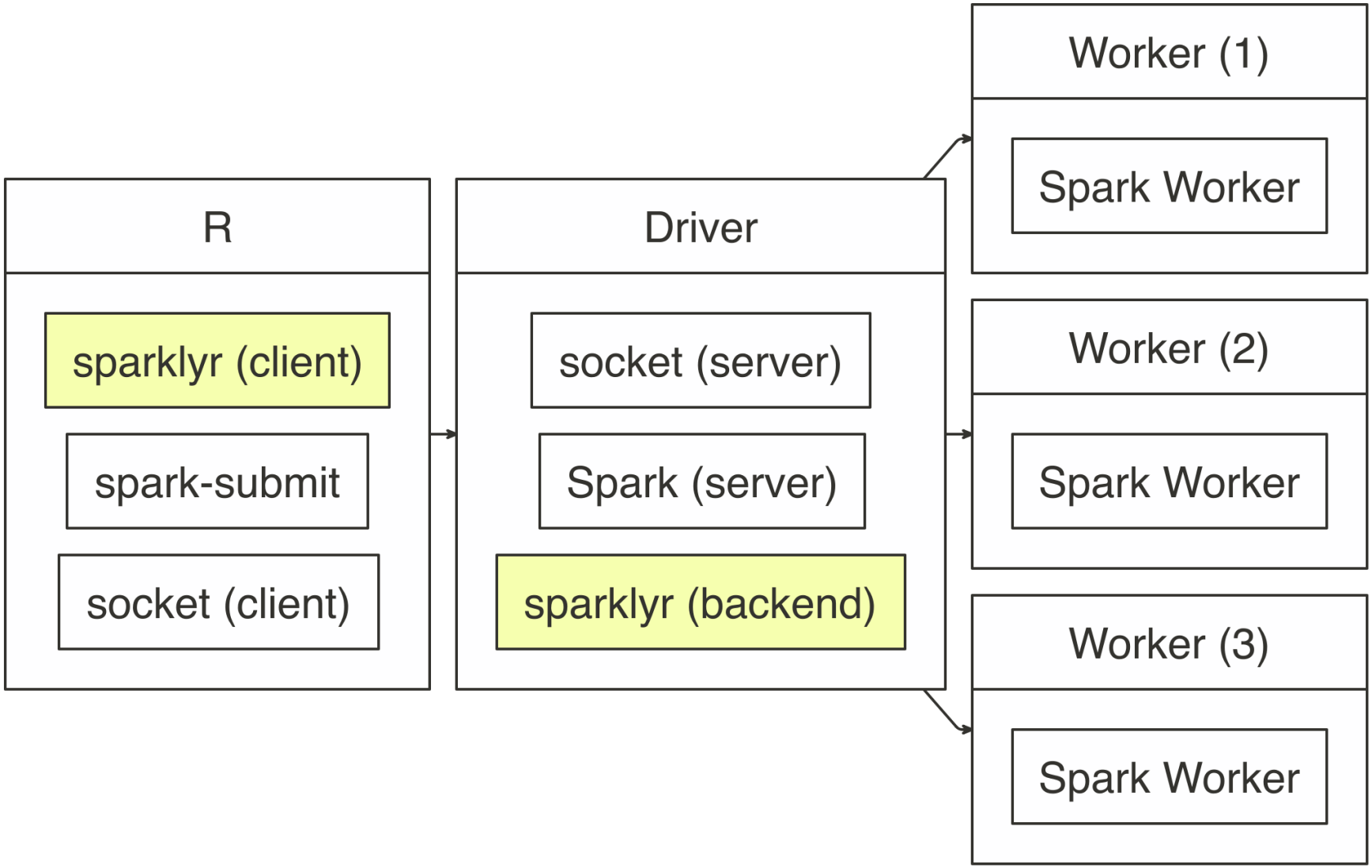
.github	tweaks	3 years ago
R	add deprecation warning	4 days ago
ci	build sparklyr with spark master	12 days ago
docs/reference	Removes programming from ref	last month
inst	add config to enable utc_timestamp	6 days ago
java	build sparklyr with spark master	12 days ago
man-roxygen	allow passing of argument to underlying pipeline stages for ml_lda	9 months ago
man	also add roxlate-ml-feature-handle-invalid temeplate tio one_hot_enco...	6 days ago
tests	remove compute_cost with spark 3.0	5 days ago

tools/readme rebuild readme 9 months ago

ARCHITECTURE OVERVIEW

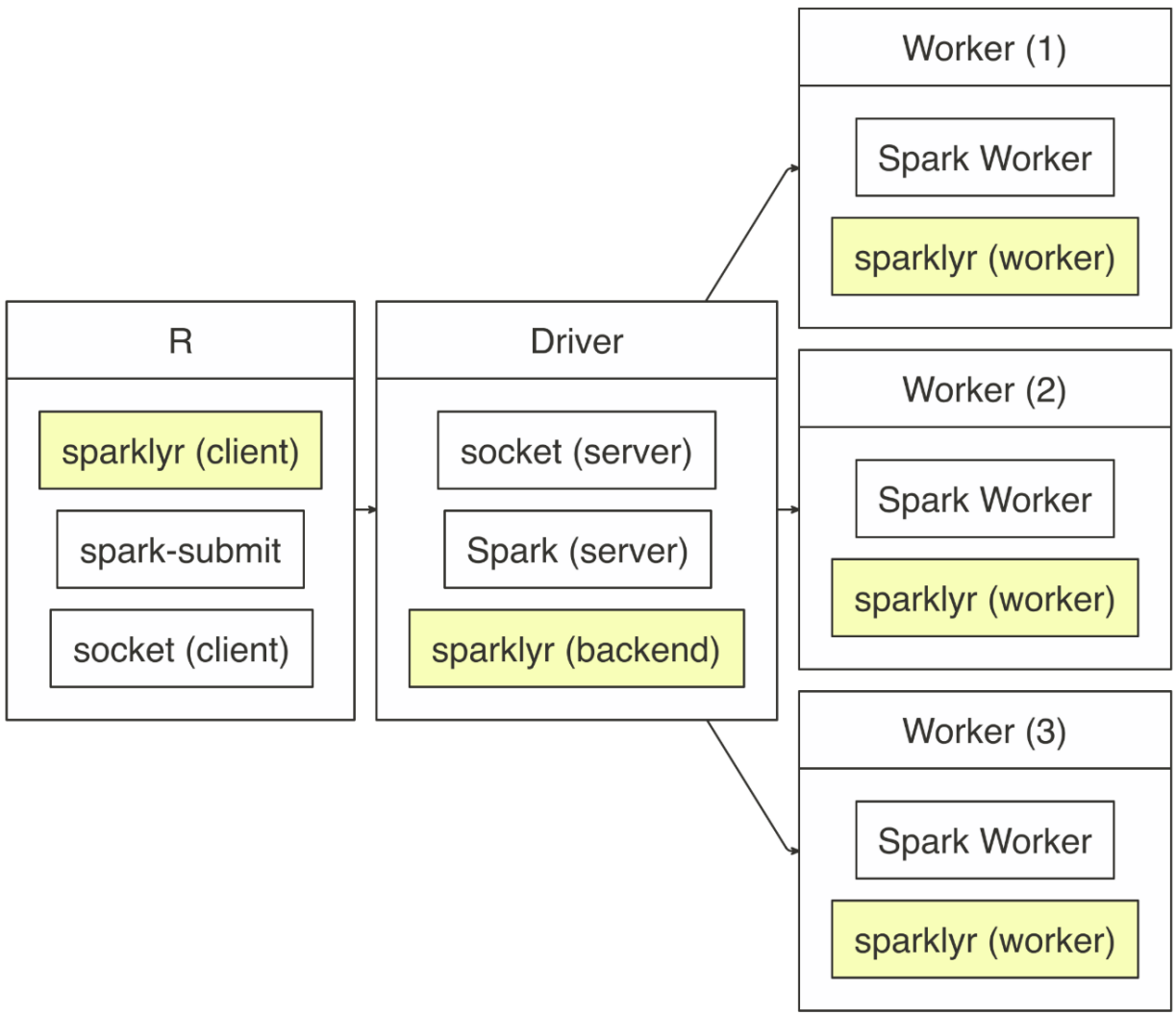
sparklyr is mostly an interface to Spark's driver node:




R
sparklyr (client)
spark-submit
socket (client)
Driver
socket (server)
Spark (server)
sparklyr (backend)
Worker (1)
Spark Worker
Worker (2)
Spark Worker
Worker (3)
Spark Worker

ARCHITECTURE OVERVIEW

Except for `spark_apply()` which enables distributing arbitrary R code:



THANKS!

NEXT STEPS

- Trademark
- GitHub Repo