SPARKLER

A web-crawler on Apache Spark

Thamme Gowda
@thammegowda

Karanjeet Singh
@_karanjeet

Dr. Chris Mattmann
@chrismattmann

https://github.com/USCDataScience/sparkler

ABOUT

Information Retrieval and Data Science (IRDS) Group

University of Southern California, Los Angeles, CA

Home page: https://irds.usc.edu Email: irds-L@mymaillists.usc.edu

Thamme Gowda

Graduate Student
@thammegowda

Karanjeet Singh

Graduate Student
@karanjeet_tw

Dr. Chris Mattmann

Director, IRDS
@chrismattmann

OVERVIEW

- About Sparkler
- Motivations for building Sparkler
- Sparkler technology stack, internals
- Features of Sparkler
- Dashboard
- Demo
- What's Next ?

ABOUT: SPARKLER

- New Open Source Web Crawler
  - A bot program that can fetch resources from the web
- Name: **Spark** Crawl**er**
- Inspired by Apache Nutch
- Like Nutch: Distributed crawler that can scale horizontally
- Unlike Nutch: Runs on top of Apache Spark
- Easy to deploy and easy to use

MOTIVATION #1

- Challenges in **DARPA MEMEX***
  - •MEMEX System has crawlers to fetch *deep and dark web data*
  - •ML based analysis to assist law keeping agencies
  - •Crawls are blackbox, we wanted real-time progress reports
- Dr. Chris Mattmann was considering an upgrade since 3 years
- Technology upgrade needed

* http://memex.jpl.nasa.gov/

WHY A NEW CRAWLER?

Modern Hadoop cluster has no Hadoop (Map-Reduce) left in it!

https://twitter.com/cutting/status/796566255830503424

MOTIVATION #2

- Challenges at **DATOIN**
  - •Intro: Datoin.com is a distributed text analytics platform
  - •Late 2014 - migrated the infrastructure from Hadoop Map Reduce to Apache Spark
  - •But the crawler component (powered by Apache Nutch) was left behind
- Met Dr. Chris Mattmann at USC in Web Search Engines class
  - •Enquired about his thoughts for running Nutch on Spark

SPARKLER: TECH STACK

- Batch crawling (similar to Apache Nutch)
- Apache Solr as crawl database
- Multi module Maven project with OSGi bundles
- Stream crawled content through Apache Kafka
- Parses everything using Apache Tika
- Crawl visualization - Banana

Maven™

APACHE felix

Solr

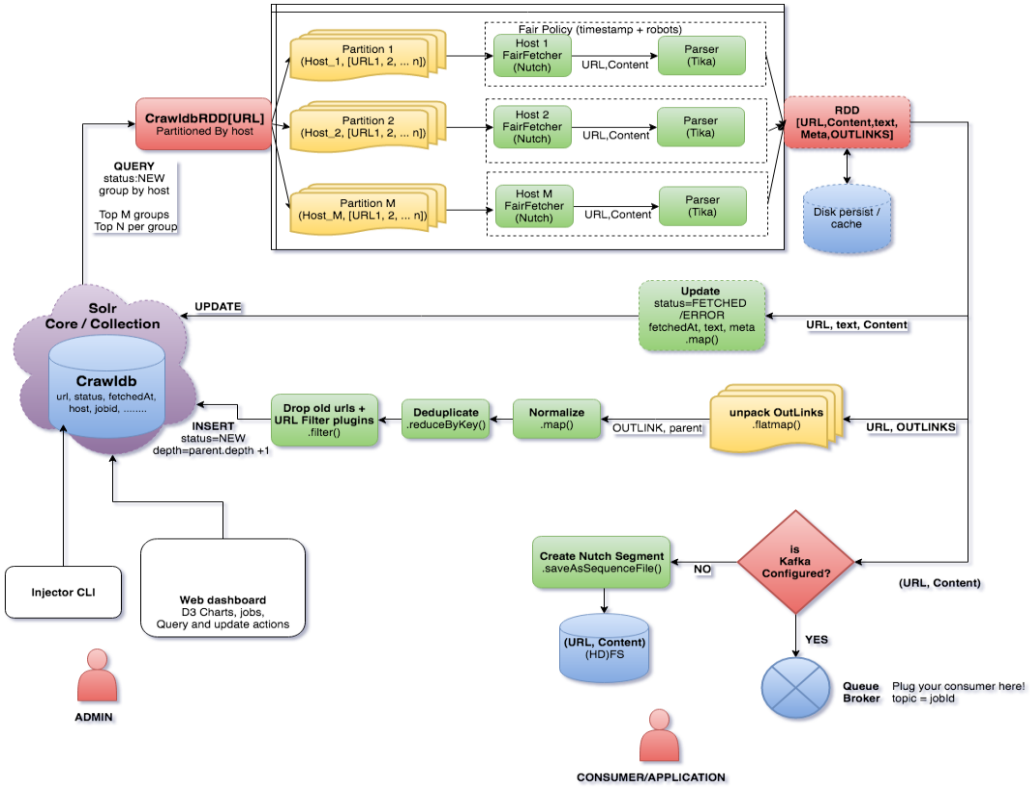
nutch

kafka

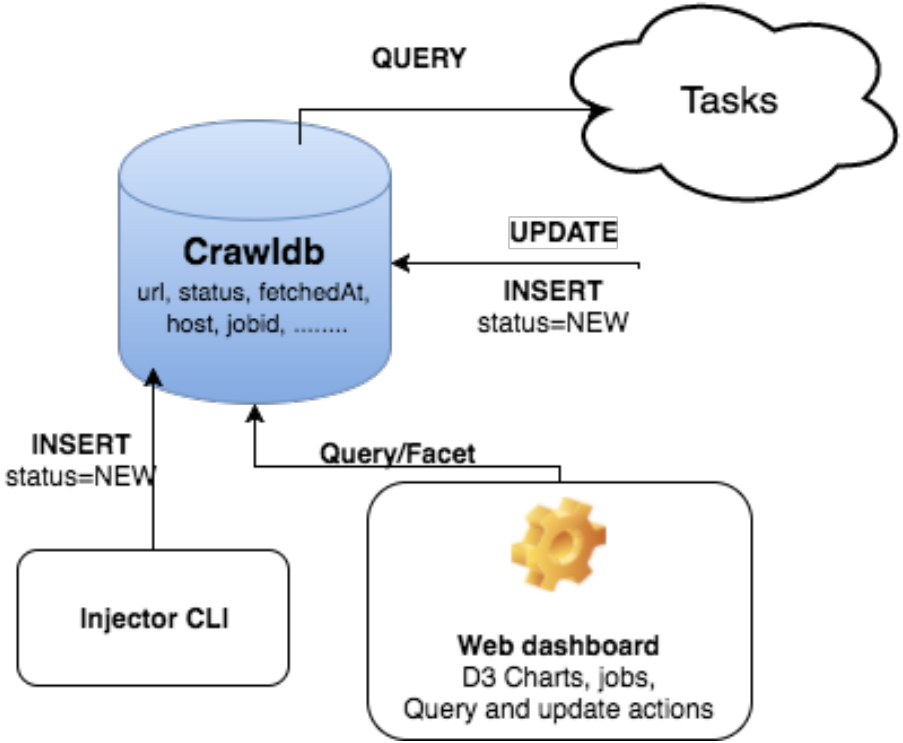
Apache™ Tika

Spark

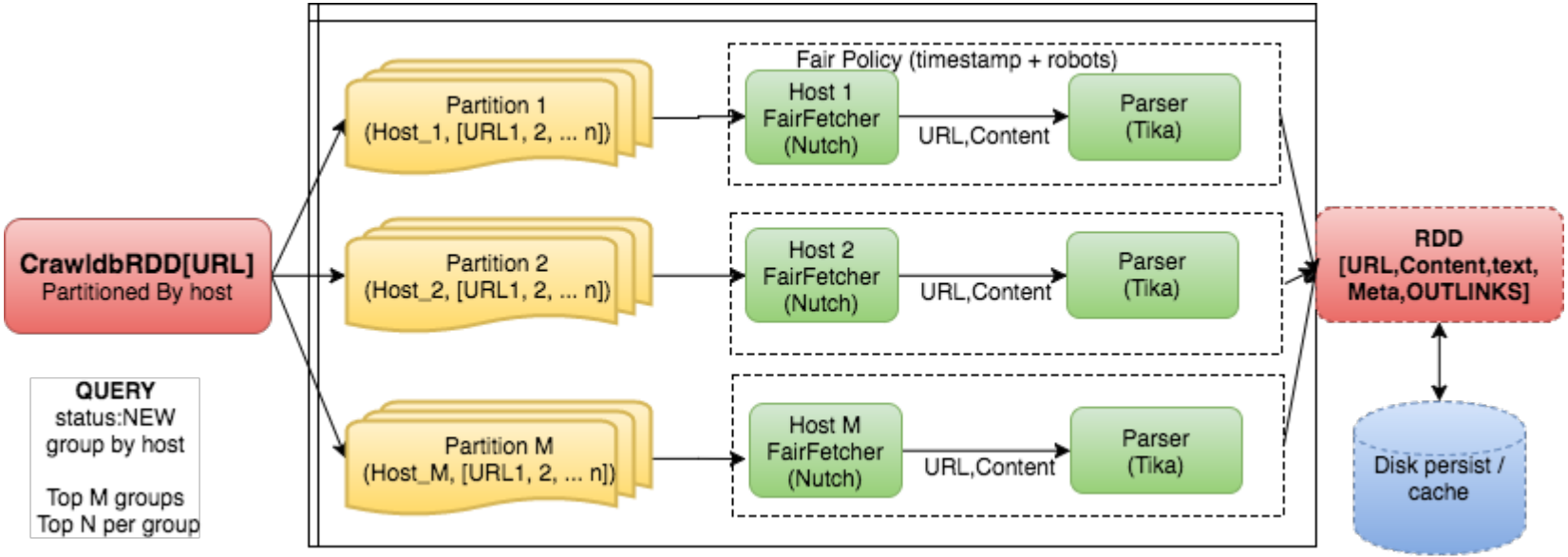
SPARKLER: INTERNALS & WORKFLOW



SPARKLER: CRAWLDB

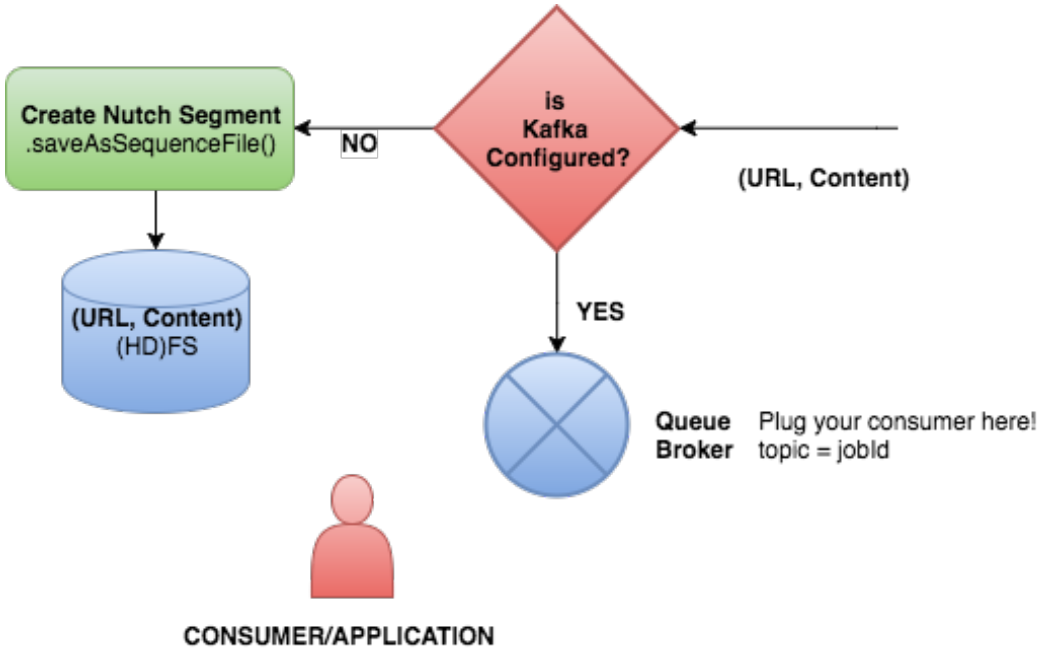


SPARKLER: RDD



SPARKLER: LINKS PIPELINE

SPARKLER: OUTPUT CONSUMPTION



SPARKLER: FEATURES

SPARKLER #1: Lucene/Solr powered Crawldb

- Crawldb needed indexing
  - For real time analytics
  - For instant visualizations
- This is internal data structure of sparkler
  - Exposed over REST API
  - Used by Sparkler-ui, the web application
- We chose Apache Solr
- Standalone Solr server or Solr cloud? *Yes!*
- Glued the crawldb and spark using CrawldbRDD

SPARKLER #2: URL Partitioning

- Politeness
  - •Doesn't hit same server too many times in distributed mode
- First version
  - •Group by: Host name
  - •Sort by: depth, score
- Customization is easy
  - •Write your own Solr query
  - •Take advantage of boosting to alter the ranking
- Partitions the dataset based on the above criteria
- Lazy evaluations and delay between the requests
  - •Performs parsing instead of waiting
  - •Inserts delay only when it is necessary

SPARKLER #3: OSGI Plugins

- Plugins Interfaces are inspired by Nutch
- Plugins are developed as per Open Service Gateway Interface (OSGI)
- We chose Apache Felix implementation of OSGI
- Migrated a plugin from Nutch
  - Regex URL Filter Plugin → The most used plugin in Nutch
- Added JavaScript plugin (described in the next slide)
- //TODO: Migrate more plugins from Nutch
  - Mavenize nutch [NUTCH-2293]

SPARKLER #4: JavaScript Rendering

- Java Script Execution* has first class support
  - •Allows Sparkler to crawl the Deep/Dark web too
- Distributable on Spark Cluster without pain
  - •Pure JVM based JavaScript engine
- This is an implementation of **FetchFunction**
- **FetchFunction**
  - •Stream<URL> → Stream<Content>
  - •Note: URLS are grouped by host
  - •Preserves cookies and reuses sessions for each iteration

* JBrowserDriver by MachinePublishers

Thanks to: Madhav Sharan
Member of USC IRDS

SPARKLER #5: Output in Kafka Streams

- Crawler is sometimes input for the applications that does deeper analysis
  - Can't fit all those deeper analysis into crawler
- Integrating to such applications made easy via Queues
- We chose Apache Kafka
  - Suits our need
    - Distributable, Scalable, Fault Tolerant
- FIXME: Larger messages such as Videos
- This is optional, default output on Shared File System (such as HDFS), compatible with Nutch

kafka

Thanks to: Rahul Palamuttam
MS CS @ Stanford University; Intern @ NASA JPL

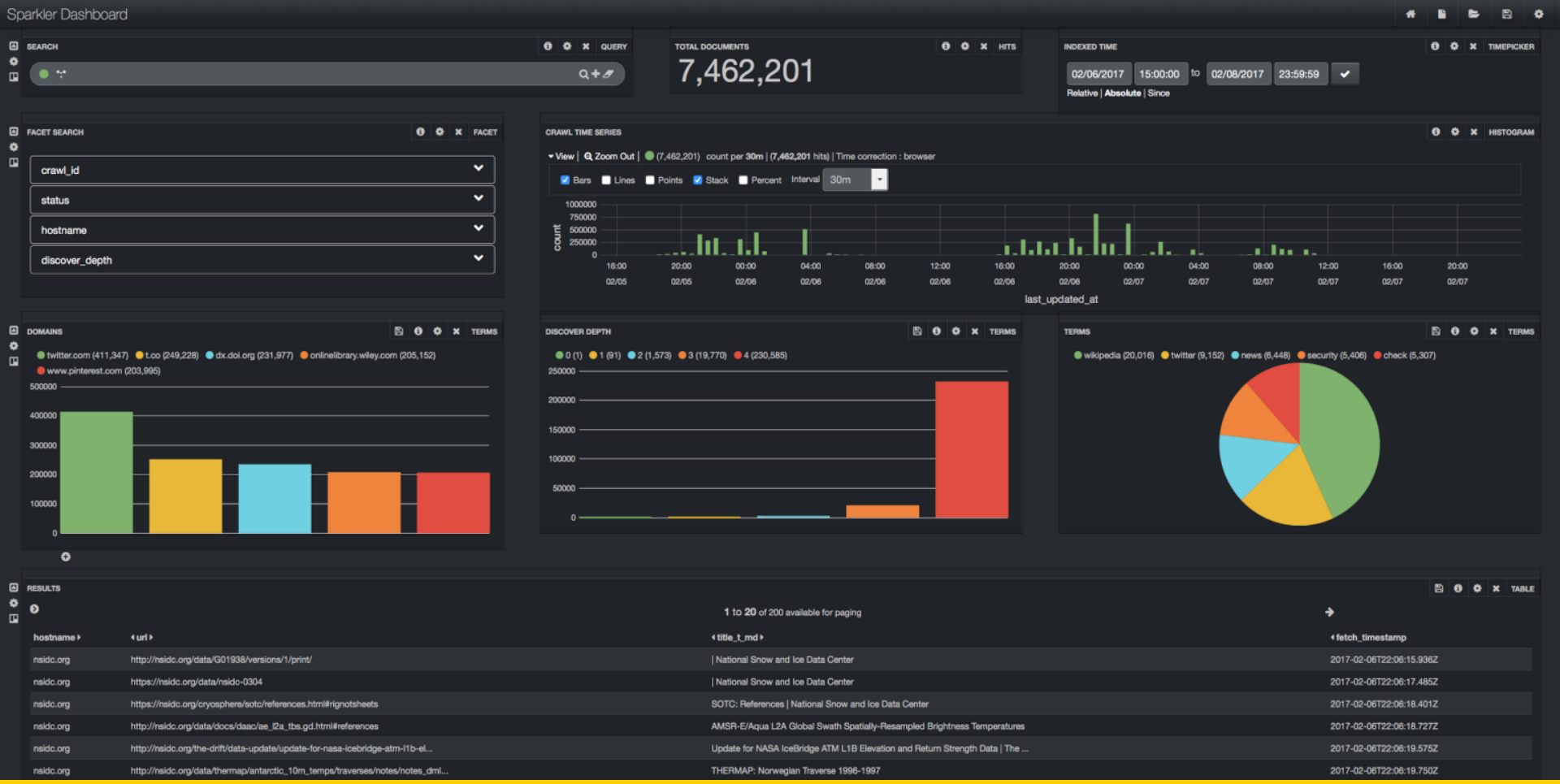
SPARKLER #6: Tika, the universal parser

- Apache Tika
  - •Is a toolkit of parsers
  - •Detects and extracts metadata, text, and URLS
  - •Over a thousand different file types
- Main application is to discover outgoing links
- The default Implementation for our **ParseFunction**

SPARKLER #7: Visual Analytics

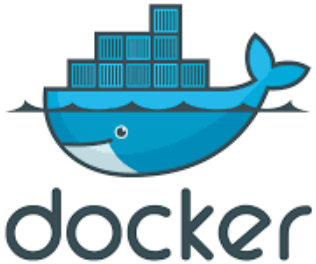
- Charts and Graphs provides nice summary of crawl job
- Real time analytics
- Example:
  - •Distribution of URLS across hosts/domains
  - •Temporal activities
  - •Status reports
- Customizable in real time
- Using Banana Dashboard from Lucidworks
- Sparkler has a sub component named sparkler-ui

Thanks to: Manish Dwibedy
MS CS University of Southern California


Sparkler Dashboard
SEARCH
QUERY
TOTAL DOCUMENTS
HITS
7,462,201
INDEXED TIME
TIMEPICKER
02/06/2017
15:00:00
to
02/08/2017
23:59:59
Relative | Absolute | Since
FACET SEARCH
FACET
crawl_id
status
hostname
discover_depth
CRAWL TIME SERIES
HISTOGRAM
View | Zoom Out | (7,462,201) count per 30m | (7,462,201 hits) | Time correction : browser
Bars
Lines
Points
Stack
Percent
Interval
30m
count
last_updated_at
DOMAINS
TERMS
twitter.com (411,347)
t.co (249,228)
dx.doi.org (231,977)
onlinelibrary.wiley.com (205,152)
www.pinterest.com (203,995)
DISCOVER DEPTH
0 (1)
1 (91)
2 (1,573)
3 (19,770)
4 (230,585)
TERMS
wikipedia (20,016)
twitter (9,152)
news (6,448)
security (5,406)
check (5,307)
RESULTS
TABLE
1 to 20 of 200 available for paging
hostname
url
title_t_md
fetch_timestamp
nsidc.org
http://nsidc.org/data/G01938/versions/1/print/
| National Snow and Ice Data Center
2017-02-06T22:06:15.936Z
nsidc.org
https://nsidc.org/data/nsidc-0304
| National Snow and Ice Data Center
2017-02-06T22:06:17.485Z
nsidc.org
https://nsidc.org/cryosphere/sotc/references.html#rignotsheets
SOTC: References | National Snow and Ice Data Center
2017-02-06T22:06:18.401Z
nsidc.org
http://nsidc.org/data/docs/daac/ae_l2a_tbs.gd.html#references
AMSR-E/Aqua L2A Global Swath Spatially-Resampled Brightness Temperatures
2017-02-06T22:06:18.727Z
nsidc.org
http://nsidc.org/the-drift/data-update/update-for-nasa-icebridge-atm-l1b-el...
Update for NASA IceBridge ATM L1B Elevation and Return Strength Data | The ...
2017-02-06T22:06:19.575Z
nsidc.org
http://nsidc.org/data/thermap/antarctic_10m_temps/traverses/notes/notes_dml...
THERMAP: Norwegian Traverse 1996-1997
2017-02-06T22:06:19.750Z

SPARKLER #8: Deployment

- Docker
- Juju Charms



Thanks to: Tom Barber
Spicule Analytics & NASA-JPL

SPARKLER #Next: What's coming?

Being used for ***Polar Deep Insights*** project

https://www.earthcube.org/group/polar-data-insights-search-analytics-deep-scientific-web

- Scoring Crawled Pages (Work in progress)
- Focused Crawling (Work in progress)
- Domain Discovery (Work in progress)
- Detailed documentation and tutorials on wiki (Work in progress)
- Interactive UI
- Crawl Graph Analysis
- Other useful plugins from Nutch

DEMO

https://github.com/USCDataScience/sparkler

```
$ bin/dockler.sh
```

QUESTIONS?

https://github.com/USCDataScience/sparkler

Sparkler is Hungry! We need more contributors!

THANK YOU

SPARK SUMMIT EAST 2017

https://github.com/USCDataScience/sparkler