

FOSS4G 2017
Boston

The Billion Object Platform (BOP): a system to lower barriers to support big, streaming, spatio-temporal data sources

Devika Kakkar and Ben Lewis
Harvard Center for Geographic Analysis

David Smiley
Independent consultant

Ariel Nunez
Terranodo



Alfred P. Sloan
FOUNDATION



Boston
Area
Research
Initiative



Center for
Geographic Analysis
Harvard University

The
Dataverse
Project 

Goal

Develop a platform that makes it easier for researchers to interactively explore large spatio-temporal datasets.

Initial focus on geo-tweets

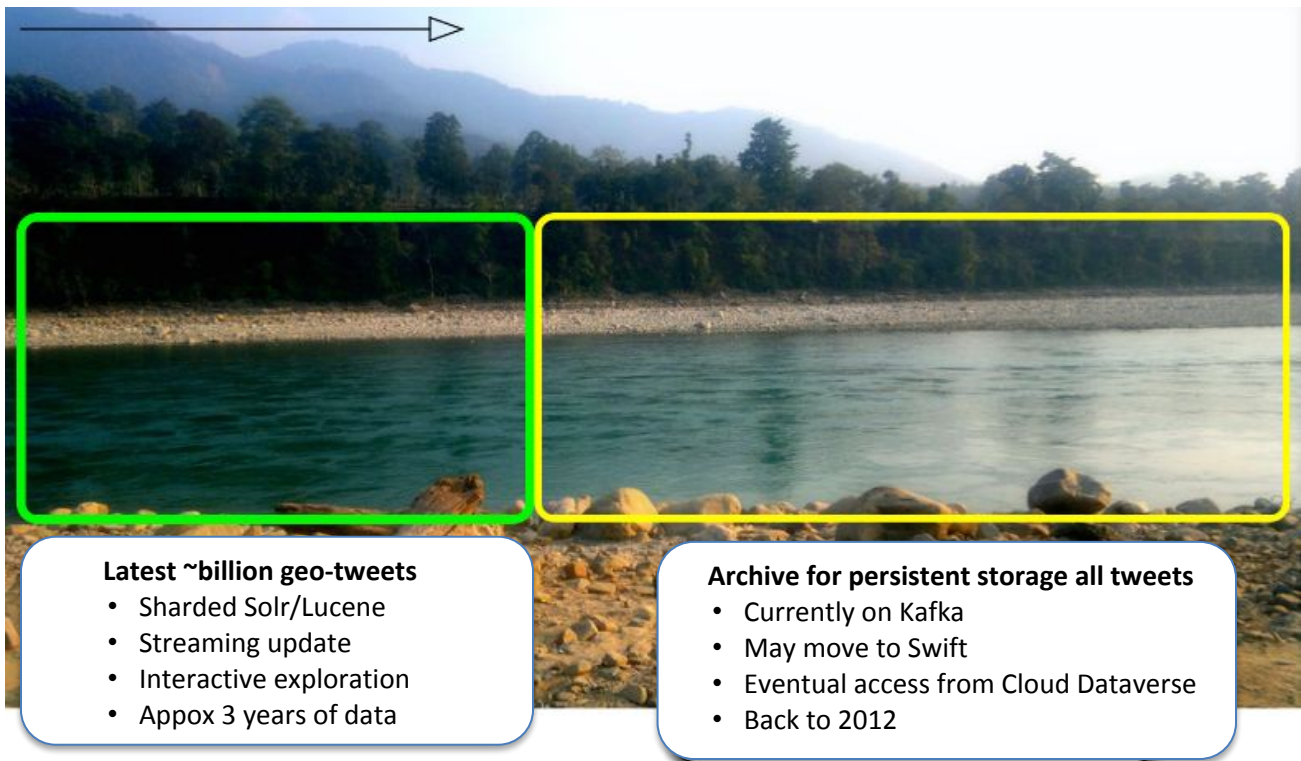
(but could be any streaming dataset)

- 1-2% of tweets have GPS coordinates from the user's device, currently about 1 million per day available via the Twitter API
- The CGA has been harvesting geo-tweets since 2012 and has an informal archive of about 8 billion objects
- Northeastern Professor Ryan Qi Wang also harvested during this period and we plan to eventually merge the two datasets to create a more complete version.

Requirements

- Develop back end and client to support interactive visualization of a billion point features
- Support sub-second queries including heatmaps and temporal histograms
- Expose a general purpose RESTful API
- Run system on low cost commodity hardware or VMs

Latest billion + long term archive



Note:

BOP visualization (2D faceting) originally developed for the HHypermap Registry

The screenshot shows the web interface for worldmap.harvard.edu/maps/new. The search bar contains the text "World Natural Gas Flaring mu...". The search results table is as follows:

Title	Source	Date
IndexLetIdent	gis.icao.int	2016
OSM: 4000 un-tagged probabl...	worldmap.har...	None
Major World Watersheds	water.discom...	2016
Major World Watersheds	water.discom...	2016
Major World Watersheds	water.discom...	2007
WRI Major Watersheds of the ...	worldmap.har...	2016
(ERS_port_nb)	maratias.dis...	2016
00	maratias.dis...	2016
(ERS_port_nb)	maratias.dis...	2016
657 Crude Oil Refineries - Ret...	worldmap.har...	2006
Oil Refineries from IndustryAb...	worldmap.har...	2016
AGEAR	gis.icao.int	2016
AGEAR	gis.icao.int	2016
World Natural Gas Flaring mu...	worldmap.har...	2014

The selected item details are:

- Title:** World Natural Gas Flaring mu206
- Source:** worldmap.harvard.edu
- Abstract:** Worldwide natural gas flaring datasetFor more information and an up to date map visit Skytruth.orgData recorded by the VIIRS instrument aboard NOAA's Suomi NPP satellite between 16 march and 31 october 2014 The flares are visualis
- Date:** Detected

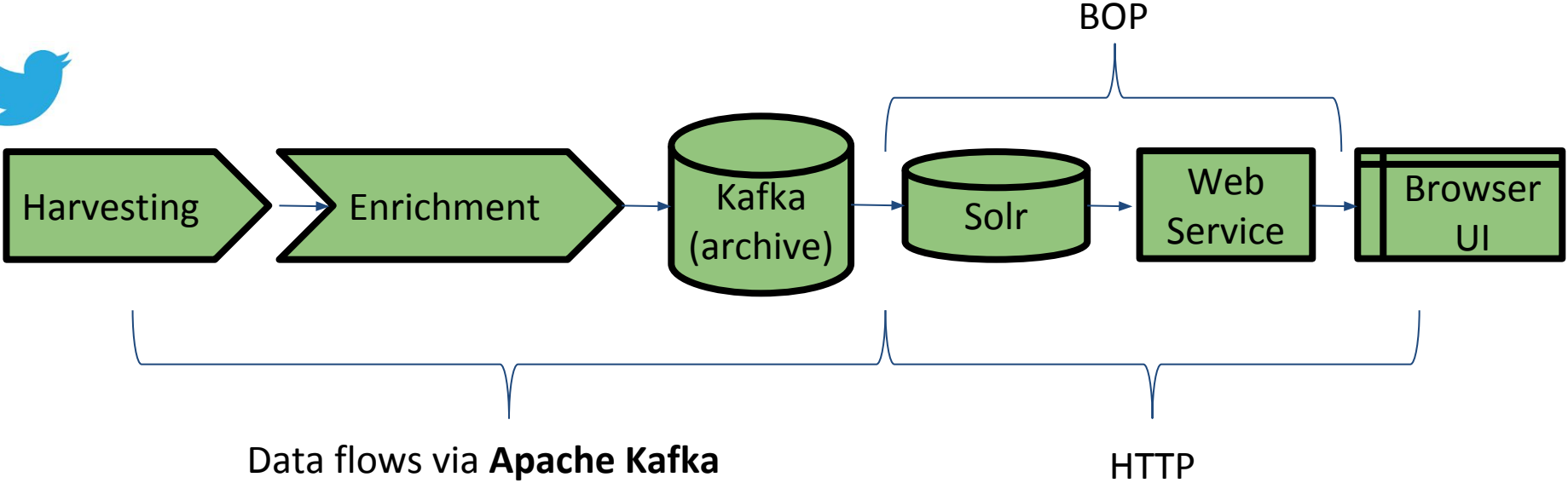
The map shows a world map with a color-coded overlay representing natural gas flaring intensity. The overlay is most prominent in the Middle East and parts of Asia, with colors ranging from yellow to red. The interface includes a search bar, a list of search results, and a detailed view of the selected item.



BOP demo URL

https://youtu.be/tib6M_fgHok

Logical High-Level Architecture



Docker, Kontena, OpenStack Hosting: Mass OpenCloud

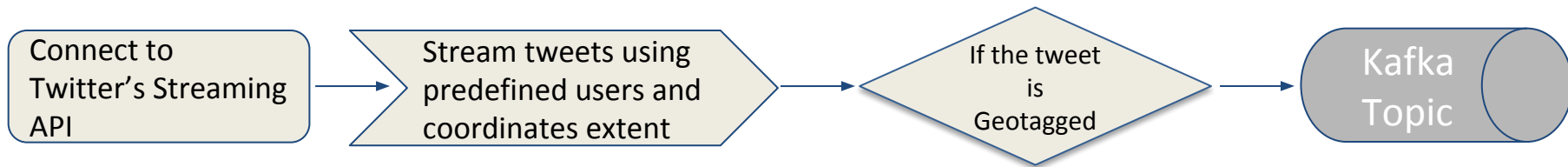
Apache Kafka



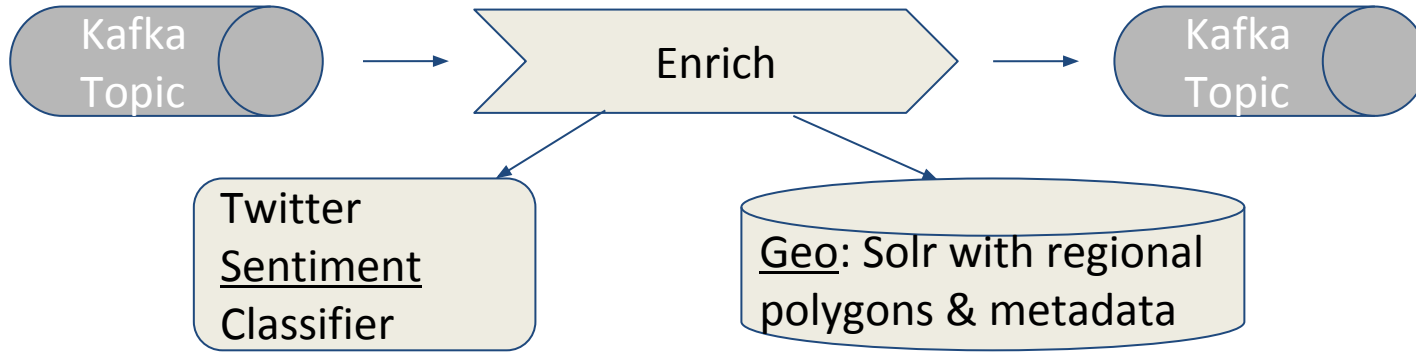
- Kafka: a scalable message/queue platform
- See new Kafka Streams & Kafka Connect APIs
- No back-pressure; can be a challenge
- Non-obvious use:
 - For storage; time partitioning
 - Lots of benefits yet serious limitations



Real-Time Harvesting



Enrichment



Geo: Query Solr via spatial point query; attach related metadata to tweet

Sentiment Analysis

- Classifier: Support Vector Machine (SVM) with Linear Kernel
- Source code in Python
- Uses scikit-learn, numpy, scipy, NLTK
- Two classes of sentiment: Positive (1), Negative (0)
- Training Corpus: Sentiment140, Polarity dataset v2.0, University of Michigan
- Preprocessing: Lower case, URLs, @user, #tags, trimming, repeating characters, emoticons
- Stemming: Porter stemmer
- Precision, Recall, F1 score: 0.82 (82%)
- Processing speed: 20ms/tweet (no emoticon), 5ms/tweet (emoticon)

Sentiment Analysis

Phase 1: Training

Train the classifier



Save as pickle

Phase 2: Prediction

Load the classifier



For each tweet

Parse



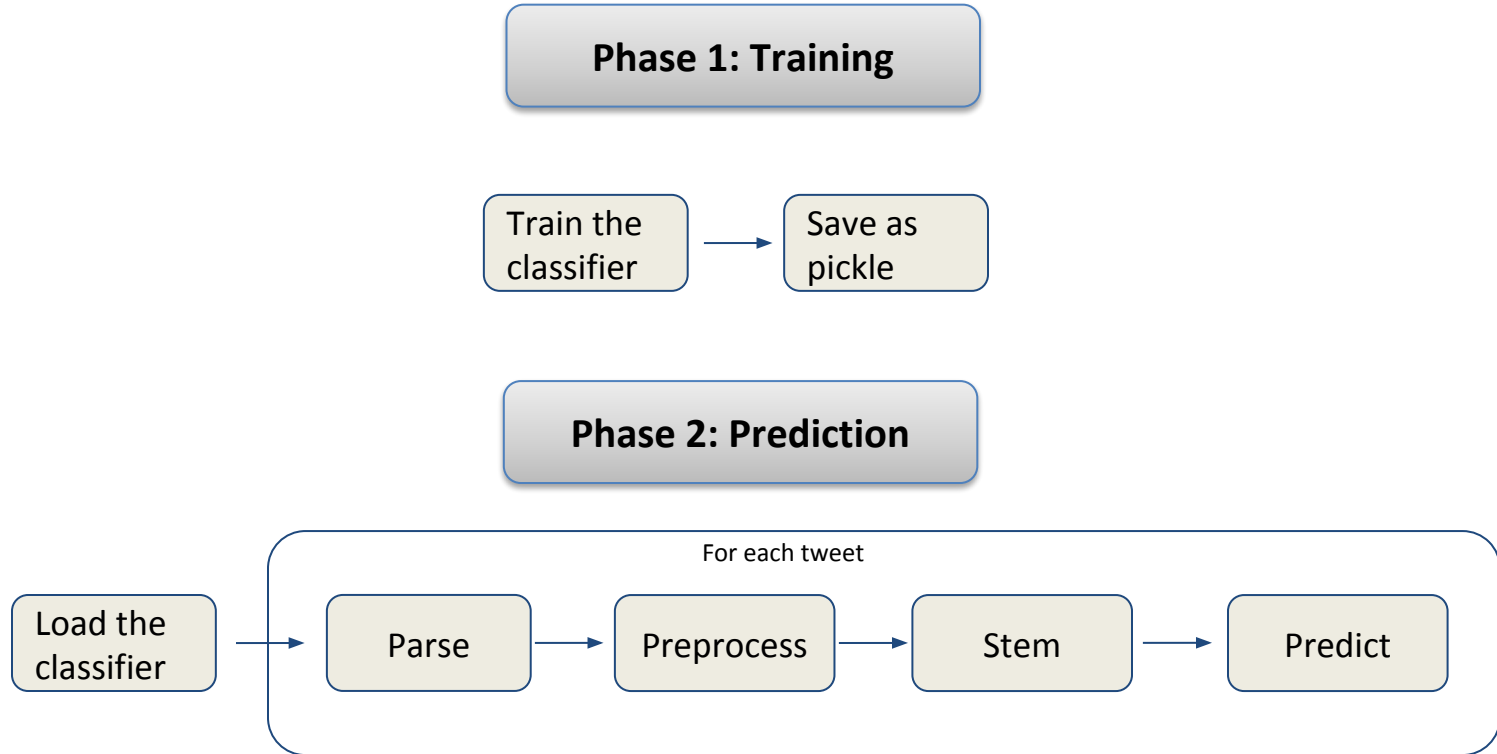
Preprocess



Stem



Predict



Solr for Geo Enrichment

“Reverse Geocoding”

- Tweets (docs) can have a geo lat/lon
- Enrich tweet with Country, State/Province, ...
 - Gazetteer lookup (point-in-polygon)

Data Set	Features	Raw size	Index time	Index size
Admin2	46,311	824 MB	510 min	892 MB
US States	74,002	747 MB	4.9 min	840 MB
Massachusetts Census Blocks	154,621	152 MB	5.9 min	507 MB

Apache Solr



- Search / analytics server, based on Lucene
- Custom add-ons:
 - Time sharded routing (index + query)
 - LatLonPointSpatialField – in Solr 6.5
 - Faster/leaner search & sort for point data
 - HeatmapSpatialField
 - Faster/leaner heatmaps at scale

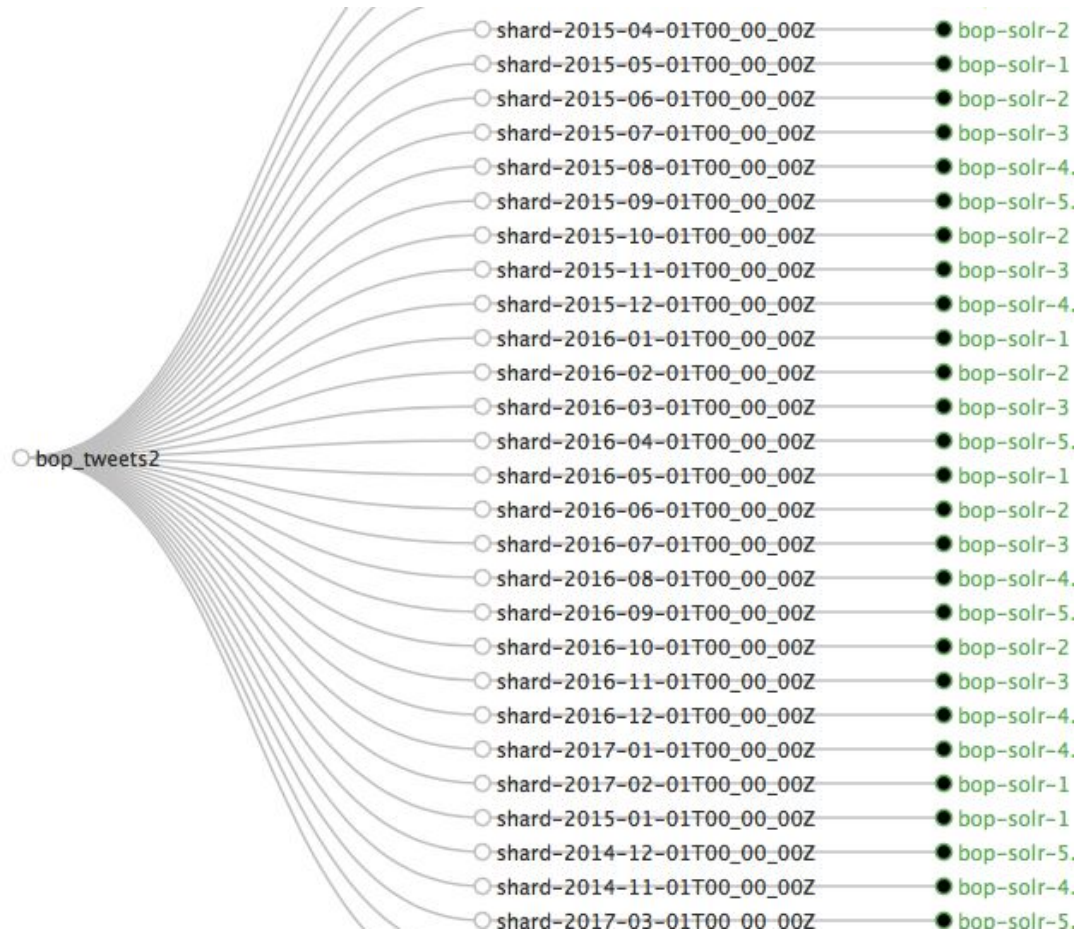
Time “Sharding”

Solr has no built-in time based sharding.

A Solr custom “URP” was developed to route tweets to the right by-month shard. It auto creates and deletes shards.

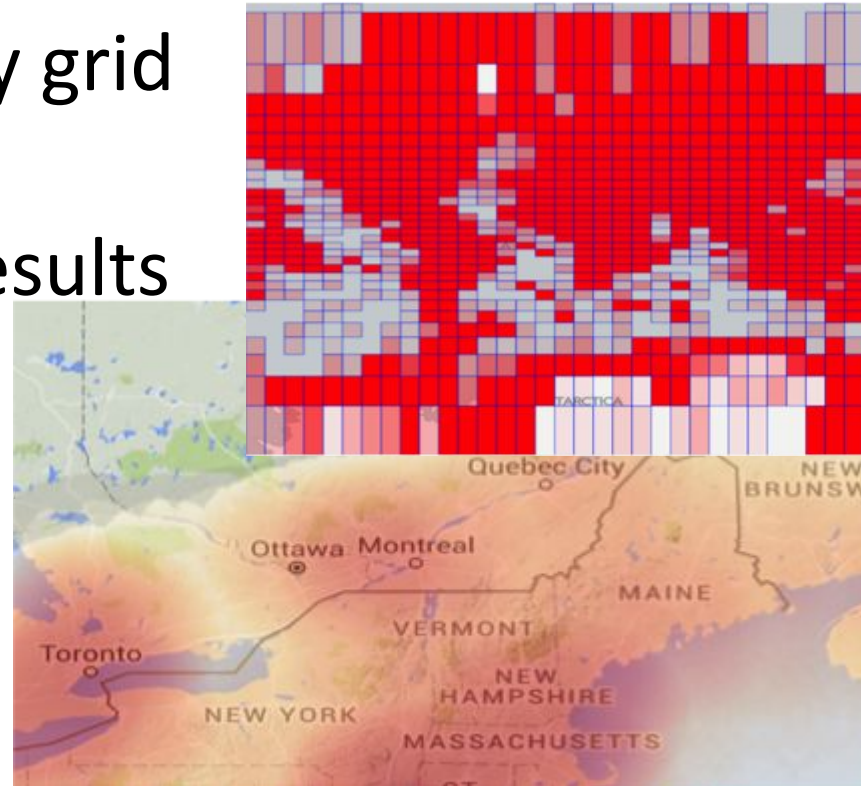
A Solr custom “SearchHandler” was developed to decide which subset of shards to search based on custom parameters sent by the web-service.

Generally useful for others. Need more work for contribution to Solr itself.



Solr Heatmaps: Grid Faceting

- Spatial density summary grid faceting, also useful for point-plotting search results
- Lucene & Solr APIs
- Scalable & fast *usually*..



The BOP Web-Service

- HTTP/REST API
 - Keyword search
 - Faceting
 - Heatmaps
 - CSV export
- Why not Solr direct?
 - Define a supported API
 - Ease of use for clients
 - Security

The screenshot shows the Swagger UI for the BOP web-service. The URL is `http://bop.worldmap.harvard.edu/bopws2/swagger.json`. The API is titled "default". Two endpoints are listed: `GET /tweets/export` and `GET /tweets/search`. The `/tweets/search` endpoint is selected, showing "Implementation Notes" and a "Response Class (Status 200)" for a successful operation. The response class is a JSON object with the following structure:

```
{
  "a.matchDocs": 0,
  "d.docs": [
    {}
  ],
  "a.time": {
    "start": "string",
    "end": "string",
    "gap": "string",
    "counts": [
      ]
    }
  }
```

The response content type is `application/json`. The "Parameters" section shows a query parameter `q.text` with a description: "Constrains docs by keyword search query."

Tech:

- Swagger
- Dropwizard
- Kotlin lang (on JVM)

UI Stack

- BOP's UI uses the following technologies:
 - Angular JS
 - OpenLayers 3
 - npm (dependencies, script minification, development)

Deployment / Operations

- MassOpenCloud “MOC”
 - OpenStack based cloud (mimics Amazon EC2)
 - CoreOS
 - Kontena & Docker
 - Admin/Ops tools:
 - Kafka Manager (Yahoo!)
 - Solr’s admin UI
- Stats:

 - 12 nodes (machines)
 - 5 to Solr
 - 3 to Kafka
 - 3 to enrichment, ...
 - 217 GB RAM
 - 3500 GB disk
 - 17 services (software pieces)
 - 133 containers

Next steps

- Persistent archive in Swift object storage
- Exploring adding analytic capabilities using GeoMesa
- Support faceting on numeric values (in addition to counts) to support other types of visualizations.

Thank you

Ben Lewis

blewis@cga.harvard.edu

David Smiley

david.w.smiley@gmail.com

Devika Kakkar

kakkar@fas.harvard.edu

Backup slides

Enter keyword



Suggestions

@user



Suggestions

@zolaroid2
@_koooooonkon_ おやすみ - 1 <http://t.co/khpv4Dmyd>
Jul 27, 2015 6:39:22 AM

@eitlfy
I'm at Casa d Lets <https://t.co/NG0RQBFGcK>
Jul 3, 2015 11:28:36 AM

@analkin1
THE TWO MOST IMPORTANT PEOPLE IN MY LIFE...@
Amantin, Ashanti, Ghana <https://t.co/EJjedTgSpN>
Jul 18, 2016 6:28:42 AM

@CIPHERHOUSE
See @snoobiii like the \$food the @opengov expired:
today!!! Shaz-zam! Sor we really want @OpenRend
#Blockade4OR #IWANT
Aug 9, 2015 4:54:39 AM

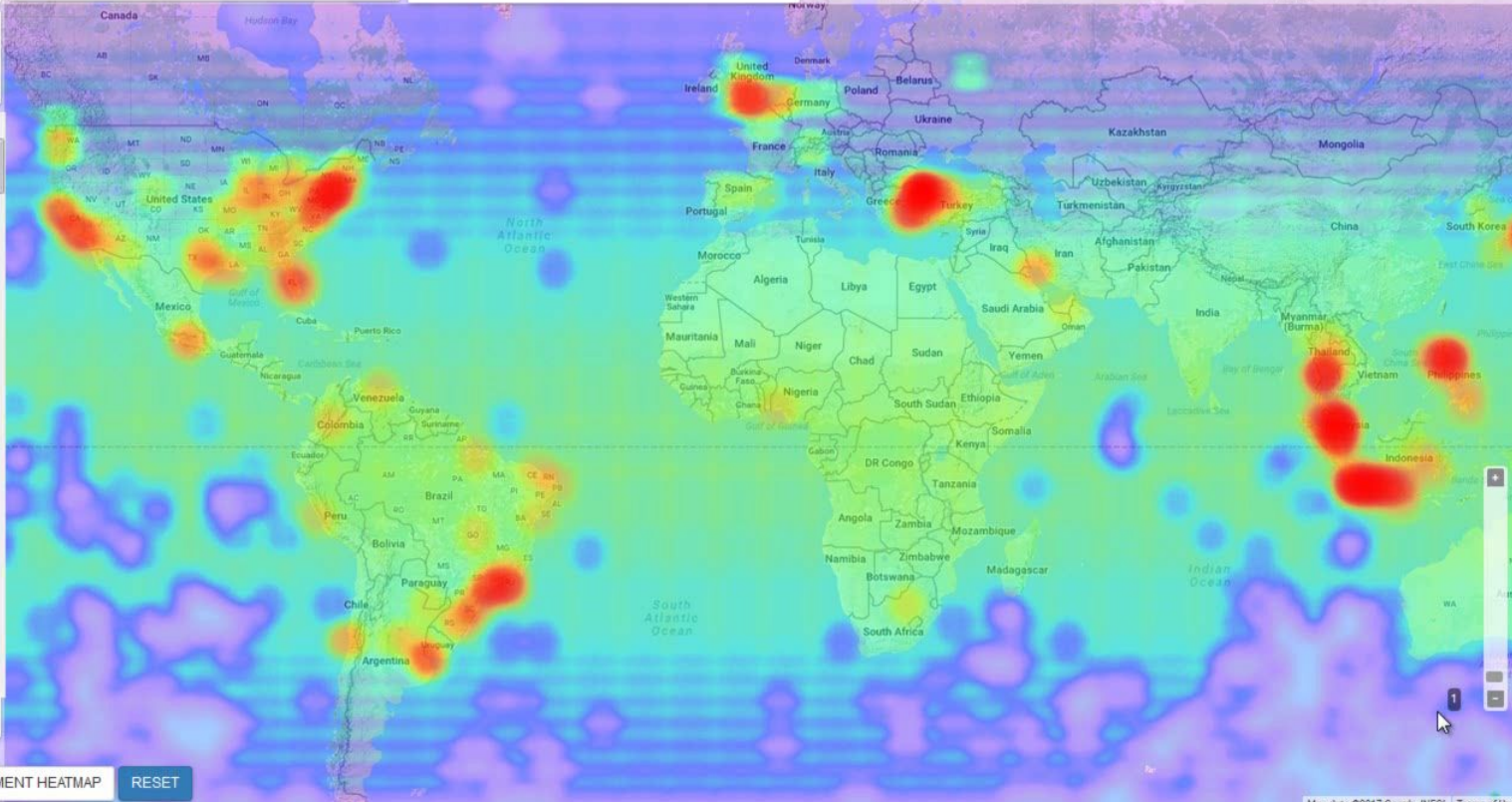
@Alhas sanemman10
@DavidLogan2020 please let have your email
Feb 14, 2016 5:37:22 PM

@OpenRend
@MadridEmpleoTrb topic similar @CIPHERHOUSE this
article <http://t.co/pU1qyCT5yj> entitled: 'Pagefip XOOFS
Module'
Aug 11, 2015 3:11:14 AM

@OforiDwo

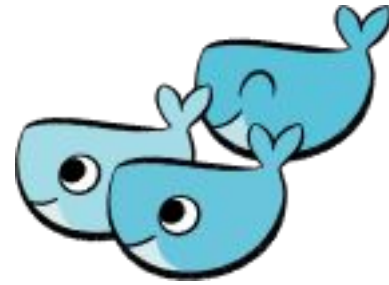
Results for keyword: 312560267

From To



DATAVERSE DOWNLOAD BASEMAPS SENTIMENT HEATMAP **RESET**

Docker



- Easy to find/try/use software
 - No installation
 - Simplified configuration (env variables)
 - Common logging
 - Isolated
- Ideal for:
 - Continuous Int. servers
 - Trying new software
 - Production advantages too
 - but “new”

Docker in Production

- We use “Kontena”
- Common logging, machine/proc stats, security
 - VPN to secure network; access everything as local
- No longer need to care about:
 - Ansible, Chef, Puppet, etc.
 - Security at network or proxy; not service specific
- Challenges: state & big-data

