



Project Fortis

Using Spark Streaming and ML for real-time insights into humanitarian crisis

Erik Schlegel [@erikschlegel1](#)
Microsoft - Senior Engineer

Erik Schlegel

- Senior Engineer – Microsoft / PCT
- Focus on emerging technology projects with innovative partners



Project Fortis: A collaboration with the UN

Challenges

- Humanitarian aid plans are manually composed
- Resulting in imprecise aid relief plans
- Aid planners monitor 400+ data sources daily
- Impacted areas often have limited accessibility
- Slow turnaround



Project Fortis: Goals

- Accelerate the construction of aid planning
- Improve its data accuracy
- Provide deeper insights and trends
- Real-time analytics
- More intelligence and insight to enable better forecasting

With those goals in mind, we built Fortis, which at its core, is a data ingestion, analysis and visualization pipeline.

Project Fortis - Showcase

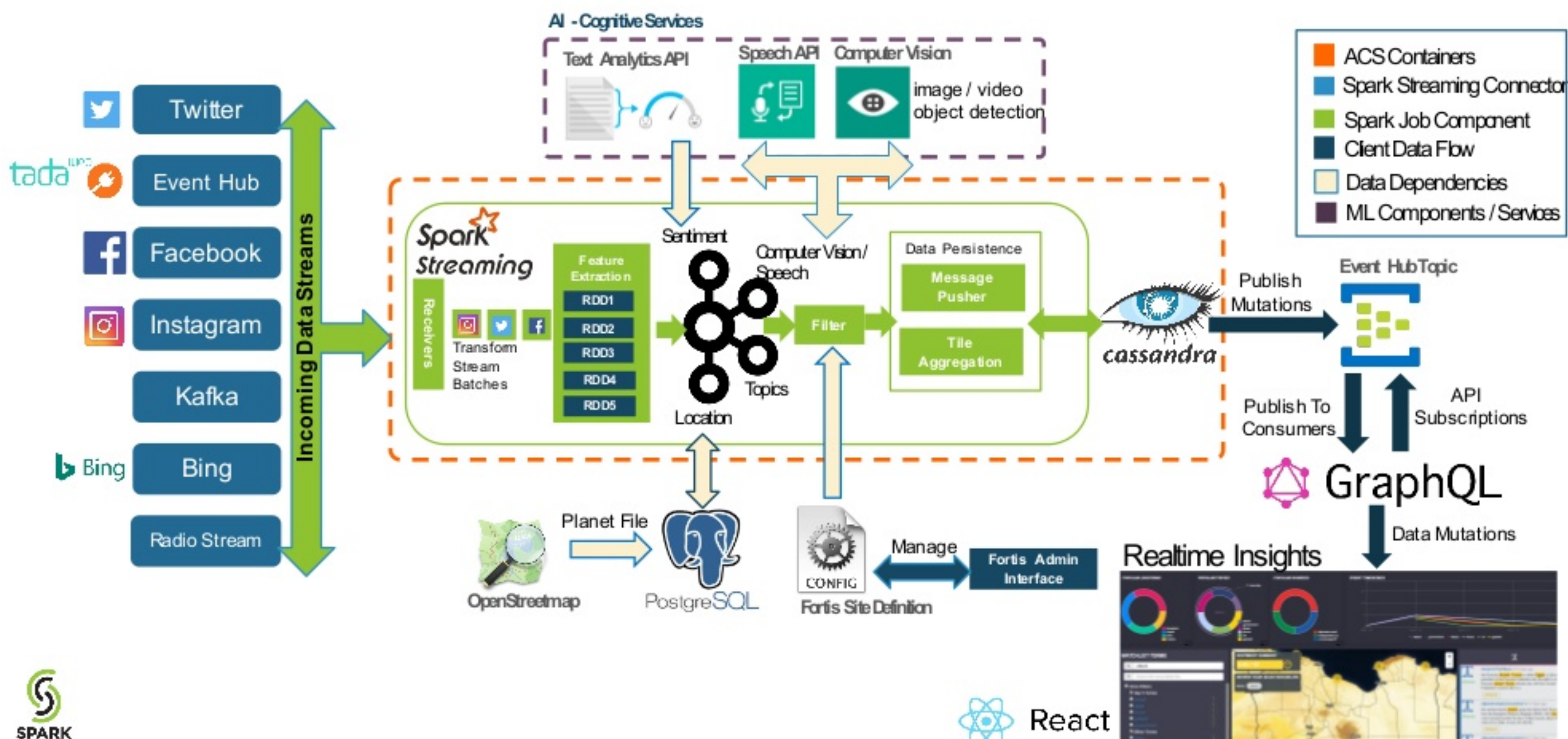


Spark Streaming: Realtime Data Pipeline

Kubernetes ACS Cluster

- Using spark streaming connector extensions
- Social and public DStreams are collected in RT
- Events are processed and merged through Spark
- Cassandra acts as the state engine for aggregation
- Results are published to Event Hub topics
- Consumers are notified through GraphQL Subscriptions

Data Pipeline: Functional Architecture



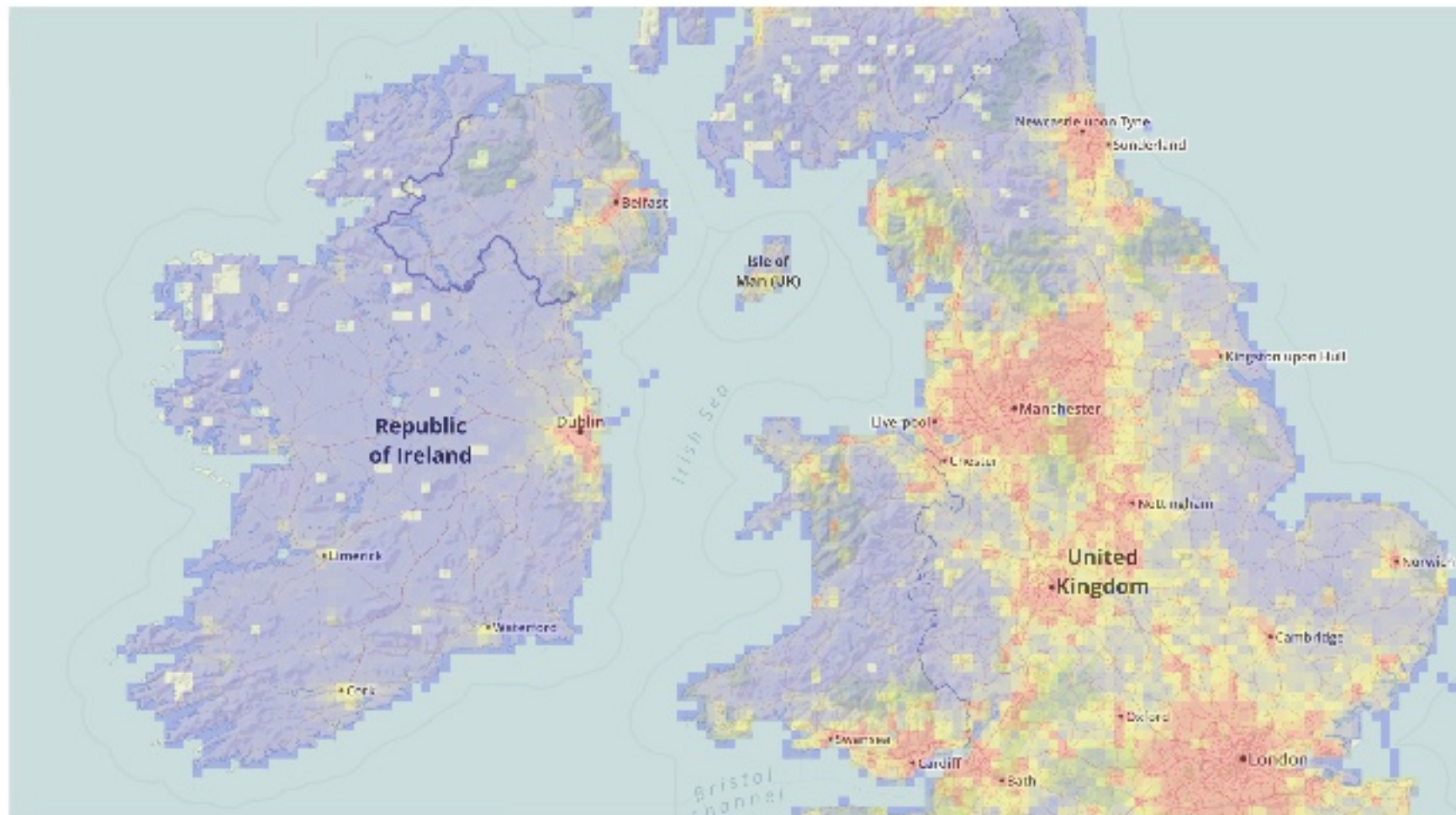
Geospatial Requirements



- Geotag mentioned places from conversations
- To query activities within a bounding box
- Filter activities based on topic(s)

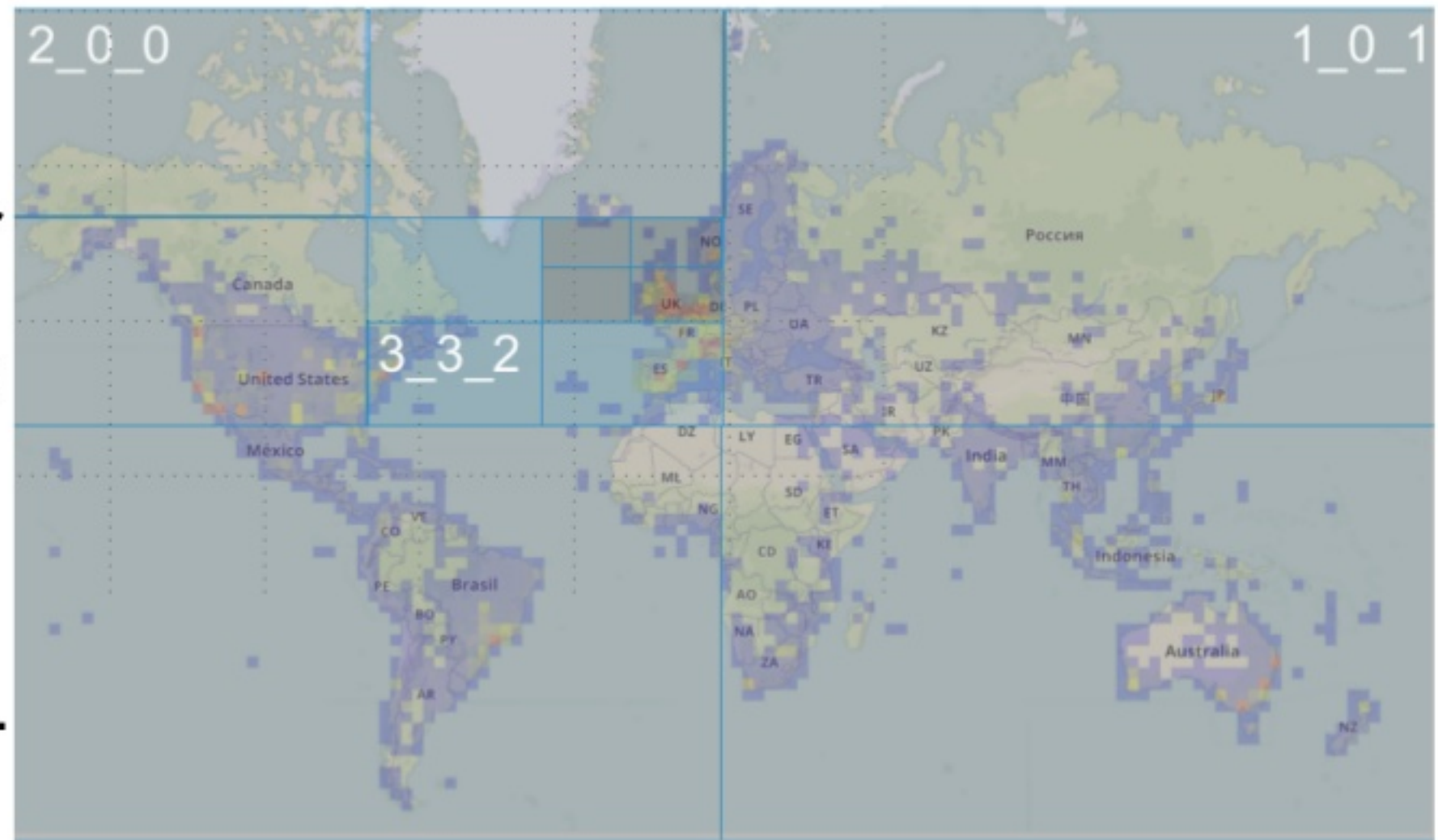
Heatmap Generation

Aggregate detected places across geographic regions



XYZ Tiles for summarization

- Divides world up into tiles.
- Each tile has four children at the next higher zoom level.
- Maps multiple layers to a single space dimension.



Heatmap Spark Mapper

For each location, map to tiles at every zoom level:

(36.9741, -122.0308) → [
 (10_398_164, 1), (11_797_329, 1)
 (12_1594_659, 1), (13_3189_1319, 1),
 (14_6378_2638, 1), (15_12757_5276,1),
 (16_25514_10552, 1), (17_51028_21105, 1),
 (18_102057_42211, 1)
]

Heatmap Spark Mapper

Reduce all mappings with the same key into an aggregate value

```
(10_398_164, 151) ← [  
  (10_398_164, 1), (10_398_164, 1), ...  
  (10_398_164, 1), (10_398_164, 1), ...  
  (10_398_164, 1), (10_398_164, 1), ...  
  (10_398_164, 1), (10_398_164, 1), ...  
  (10_398_164, 1)  
]
```

Heatmap Spark Algorithm

For each location, compute a tile id key/value pair for all zoom levels

```
final val MAX_ZOOM_LEVEL = 16
final val MIN_ZOOM_LEVEL = 5

def tile_id_mapper(location: (Double, Double)): List[(String, Int)] = {
  (for (zoom <- MIN_ZOOM_LEVEL to MAX_ZOOM_LEVEL)
    yield (TileUtils().tile_id_from_lat_long(location._1, location._2, zoom).tileId, 1)).toList
}
```

Heatmap Spark

Build the heatmap then save the reduced dataset to Cassandra

```
val locationListSamples = List[(Double, Double)]((30.294221,-97.7760937),  
(30.294221,-97.7760937), (30.4007241,-97.7368647),(30.4007241,-97.7368647),....)  
val locationRdd = sc.parallelize(locationListSamples)  
  
locationRdd.flatMap(tile_id_mapper)  
  .reduceByKey(_+_)  
  .saveToCassandra("fortis", "tilesExample", SomeColumns("tileId", "count" append))
```

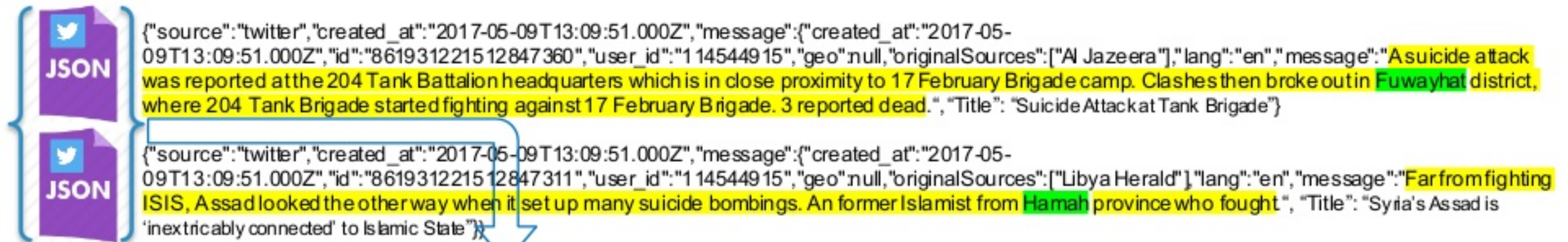

Aggregated Tile Schema

All inbound events are aggregated by
tile_x, tile_y, tile_z, period, source, publisher, topic, lang

Aggregated Tile Data - Cassandra: fortis_tiles_tbl

tile_x	tile_y	tile_z	period	Source	publisher	Topic	Lang	feature_collection
15	13346	15	month-2017-06	Twitter	Al Jazeera	isis	en	{ "mention_count": 1213, "sentiments_avg": ["neg": .78, "pos": .02], "entities": [{ "name": "bashar assad", "mention_count": 627, "ref_id": "3256" }] }
231	3345	14	month-2017-06	Facebook	Times of Libya	Isis	en	{ "mention_count": 453, "sentiments_avg": ["neg": .78, "pos": .02], "entities": [{ "name": "bashar assad", "mention_count": 124, "ref_id": "3256" }] }
76	98242	13	month-2017-06	Facebook	Times of Libya	isis	en	{ "mention_count": 453, "sentiments_avg": ["neg": .78, "pos": .02], "entities": [{ "name": "bashar assad", "mention_count": 124, "ref_id": "3256" }] }

NLP Feature Extraction



Event Details – Cassandra: fortis_events_tbl

event_id	source	Title	src_url	detected_features	Publisher	Topics	lang	message_body	Feature_collection
851144530439090180	twitter	Syria's Assad is 'inextricably connected' to Islamic State	https://twitter.com/AJEnglish/status/851144530439090180	["wof85678363"]	Al Jazeera	["isis", "bombings"]	en	Far from fighting ISIS, Assad looked the other way when it set up many suicide bombings. An former Islamist from Hamah province who fought	{ "sentiments_avg": [{"neg": ".78", "pos": ".02"}, "entities": [{"name": "bashar assad", "ref_id": "3256"}]}
861931221512847360	twitter	Suicide Attack at Tank Brigade	https://twitter.com/AJEnglish/status/851144530439090180	["wof85678323"]	Libya Herald	["suicide", "attack", "clashes"]	en	A suicide attack was reported at the 204 Tank Battalion headquarters which is in close proximity to 17 February Brigade camp. Clashes then broke out in Fuwayhat	{ "sentiments_avg": [{"neg": ".78", "pos": ".02"}, "ref_id": "3256"}]}

Realtime Audio Streaming with Spark

- Speech Recognition APIs through radio broadcasts



Kubernetes Helm Contributions

Available on github.com/CatalystCode/charts

- Spark 2.1 / Zeppelin
- High availability chart for Kafka
- High availability chart for Cassandra
- ELK chart for logging

Streaming Connector Contributions

- Instagram / Computer Vision Integration

Maven [com.github.catalystcode:streaming-instagram_2.11](#)

- Bing streaming with customized search ranking

Maven [com.github.catalystcode:streaming-bing_2.11](#)

- Facebook

Maven [com.github.catalystcode:streaming-facebook_2.11](#)

Spark Streaming

Accelerating our ability to use data for good



Thank You.



Erik Schlegel  @erikschlegel1

Do more open source!

github.com/catalystcode/project-fortis