KAFKA + STORM= REALTIME DATA

Vaibhav Puranik
Principal Engineer
vaibhav@gumgum.com

GUMGUM

IN-IMAGE ADVERTISING

Contextually-targeted marketing messages served on images where users are actively engaged.

REALTIME DATA

COMPANY

We help advertisers achieve their brand objectives and premium publishers earn more revenue

First to introduce in-image advertising and hold multiple patents related to the technology

Experienced team of 24 and growing – several open sales and technology positions

Based in Santa Monica and active in local technology community

REALTIME DATA

HADOOP KAFKA AND STORM

Hadoop
Batch processing (using map reduce) system and HDFS

Kafka
High throughput distributed messaging system

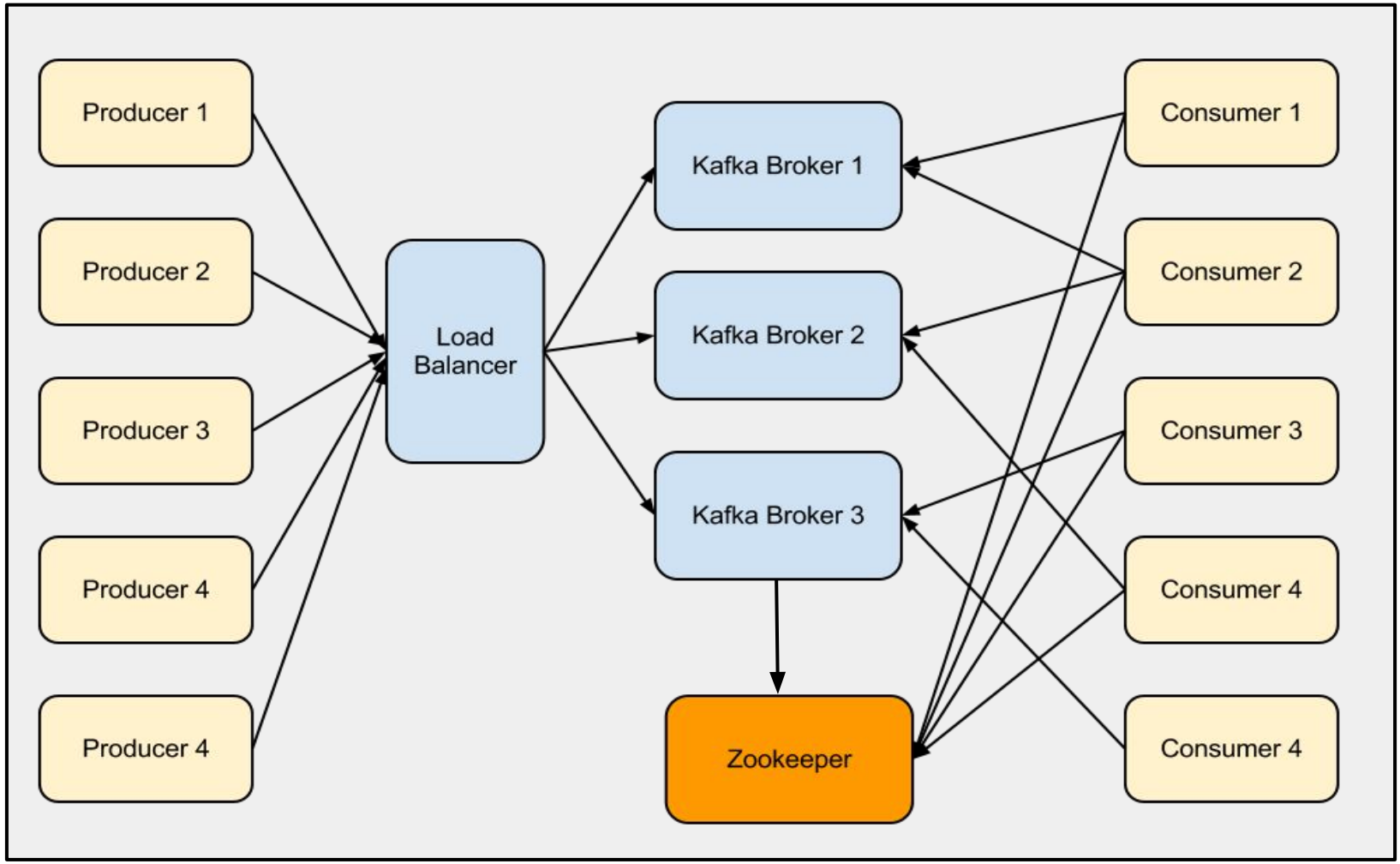
Storm
Distributed realtime computation system

APACHE KAFKA

KAFKA

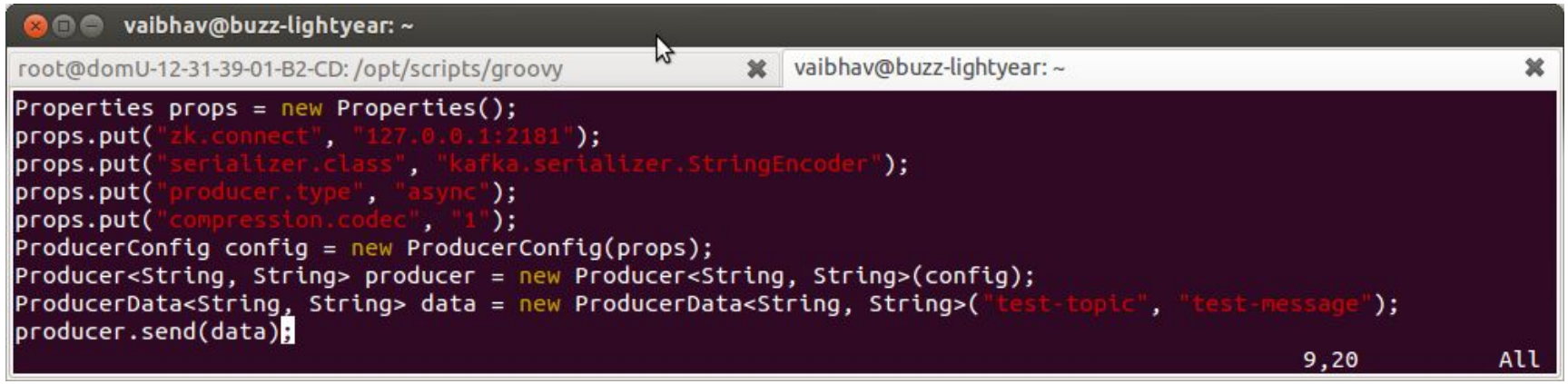
- Originated at LinkedIn
- High throughput distributed messaging system
- Sequential disc access
- Why Kafka?

APACHE KAFKA



APACHE KAFKA

PRODUCER



- Sync Producer
- Async Producer
- Compression
- Log4j Appender

APACHE KAFKA

BROKER AND TOPICS

- Topics
- Can be load balanced
- Zookeeper based coordination
- Partitioning
- Mirroring
- Replication – 0.8

APACHE KAFKA

CONSUMER

- SimpleConsumer
- ConsumerConnector
- Hadoop Consumer
- Consumer Groups
- Queue or Topic Semantics

APACHE KAFKA

CONSUMER

vaibhav@buzz-lightyear: ~

root@domU-12-31-39-01-B2-CD: /opt/scripts/groovy | vaibhav@buzz-lightyear: ~

```
// specify some consumer properties
Properties props = new Properties();
props.put("zk.connect", "localhost:2181");
props.put("zk.connectiontimeout.ms", "1000000");
props.put("groupid", "test_group");

// Create the connection to the cluster
ConsumerConfig consumerConfig = new ConsumerConfig(props);
ConsumerConnector consumerConnector = Consumer.createJavaConsumerConnector(consumerConfig);

// create 4 partitions of the stream for topic “test”, to allow 4 threads to consume
Map<String, List<KafkaStream<Message>>> topicMessageStreams =
    consumerConnector.createMessageStreams(ImmutableMap.of("test", 4));
List<KafkaStream<Message>> streams = topicMessageStreams.get("test");

// create list of 4 threads to consume from each of the partitions
ExecutorService executor = Executors.newFixedThreadPool(4);

// consume the messages in the threads
for(final KafkaStream<Message> stream: streams) {
  executor.submit(new Runnable() {
    public void run() {
      for(MessageAndMetadata msgAndMetadata: stream) {
        // process message (msgAndMetadata.message())
      }
    }
  });
}
                                                                    1,1           All
```

APACHE KAFKA

GUMGUM KAFKA INSTALLATION

- 100 million events per day
- 3 m1.small with ELB
- 1 c1.medium zookeeper
- 72 hours of retention period
- 10 producers, 8 consumers
- gzip compression enabled
- string messages

APACHE KAFKA

PROBLEMS FACED

- ELB problem - 60 seconds
- Zookeeper disc full
- If consumption is lagging, you may lose events

STORM

STORM

Twitter Acquires Social Analytics Platform BackType

LEENA RAO

Tuesday, July 5th, 2011 **9 Comments**

Twitter has just **acquired** social analytics startup **BackType.** Financial terms of the deal were not disclosed.

BackType's analytics dashboard aims to help brands and agencies understand the business impact of social media in order to make more intelligent marketing decisions. For example, the company's product BackTweets helps publishers understand the reach of their Tweets and content, who they are reaching, and how Tweets covert to web traffic, sales and other KPIs. The company assists more than 100 companies with their social media analytics, from The New York Times and Edelman to startups like Bitly, HubSpot, Hunch and SlideShare.

BackType will be joining Twitter's platform team, where they will be developing tools for Twitter's publisher partners. Along with BackType's technology, this also seems to be a pretty big **talent acquisition** for Twitter.

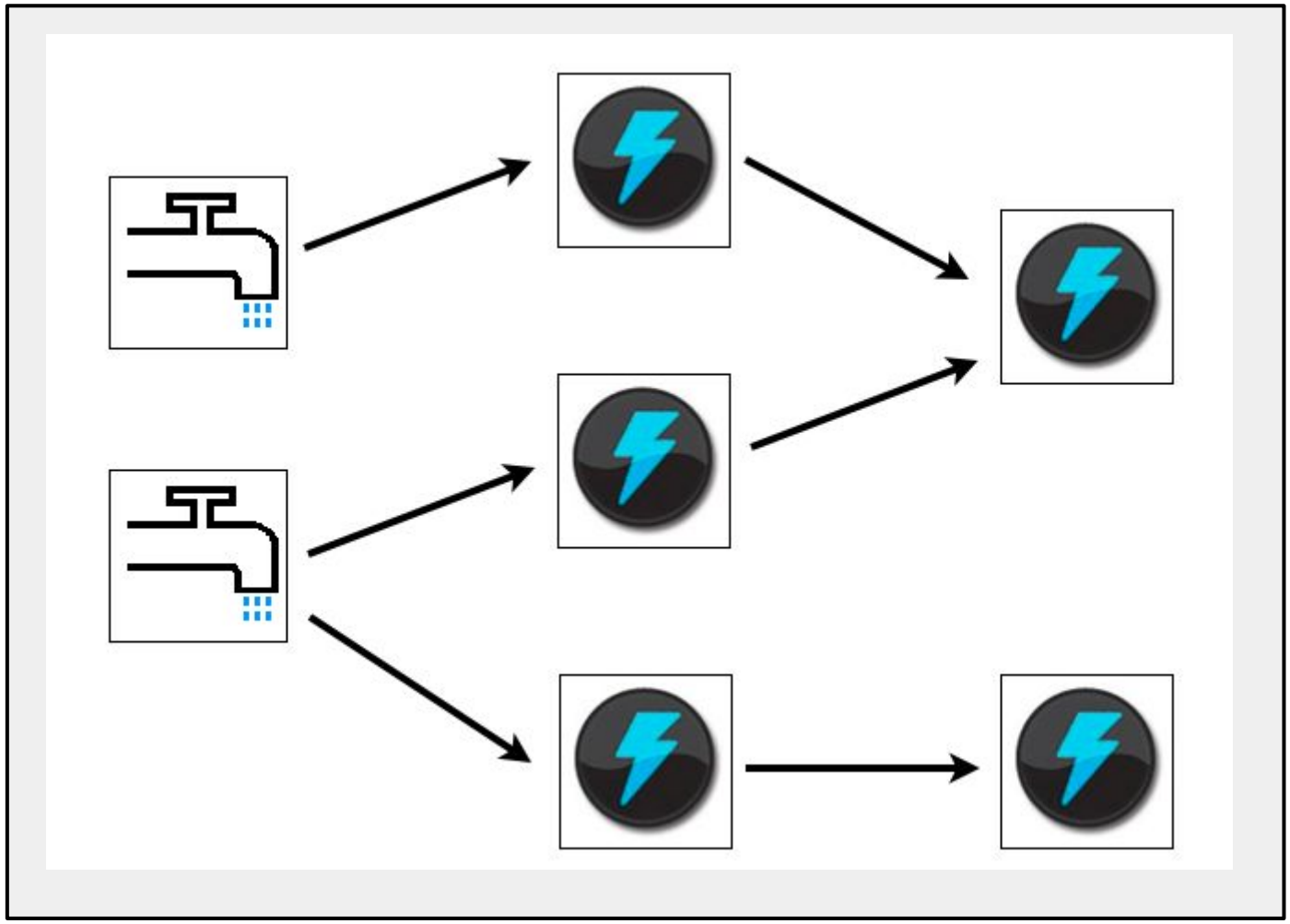
BackType also **offers a WordPress plug-in** that allowed users integrate relevant comments from social media sites like Twitter and Facebook back to the original post on WordPress.

BackType has raised **over $1 million** in funding from Y Combinator, True Ventures, K9 Ventures, Freestyle Capital, Lowercase Capital, 500 Startups, Founder Collective, Raymond Tonsing, and others.

The company says that its BackTweets product will be offered to current users for free. But the company will no longer accept new registrations for BackTweets, and eventually the BackType product and API services will be discontinued.

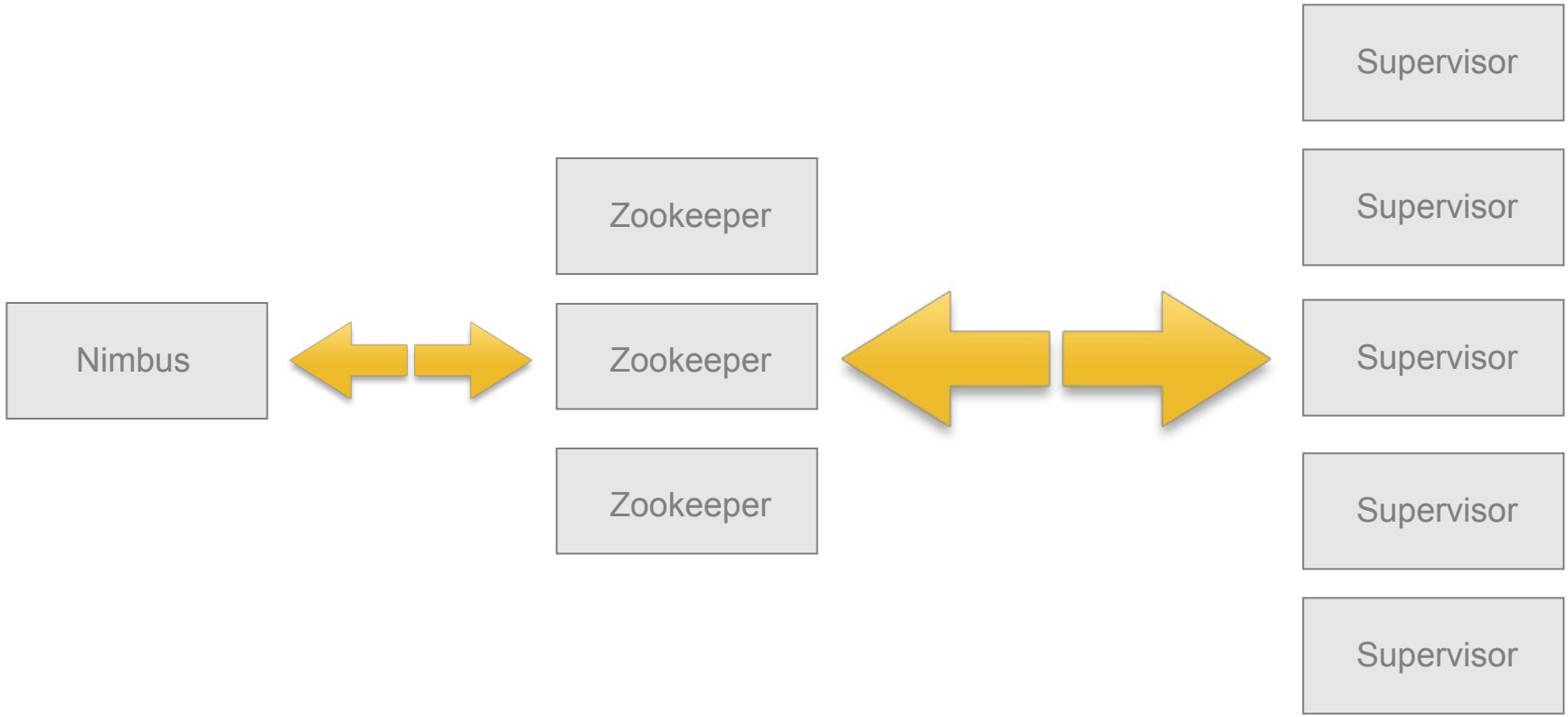
STORM

SPOUTS AND BOLTS



STORM

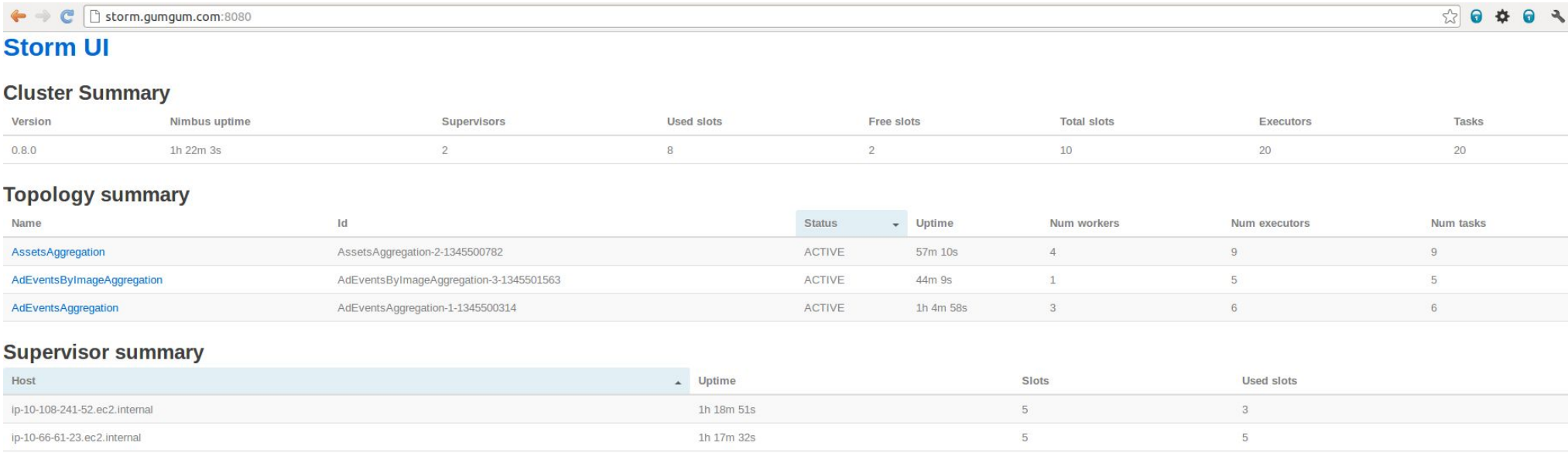
PHYSICAL ARCHITECTURE



- Nimbus
- Supervisors
- Workers
- Executors
- Tasks

STORM

STORM UI



- Fields shown are available through thrift interface

SPOUTS AND BOLTS

- Implement IRichSpout
- Spout - Override open, close, activate, deactivate, nextTuple and declareOutputFields
- Implement IRichBolt
- Override prepare, execute, declareOutputFields, cleanup
- Bolts must call ack

SAMPLE TOPOLOGY

vaibhav@buzz-lightyear: ~

vaibhav@buzz-lightyear: ~ | vaibhav@buzz-lightyear: ~

```
TopologyBuilder builder = new TopologyBuilder();
SimpleKafkaSpout kafkaSpout = new SimpleKafkaSpout(Topic.AD_EVENTS.toString(), 5000);
builder.setSpout("kafka-ad-event-spout", kafkaSpout, 2);
AdEventsAggregationBolt aggregationBolt = new AdEventsAggregationBolt();
builder.setBolt("ad-events-aggregation-bolt", aggregationBolt, 3).shuffleGrouping("kafka-ad-event-spout");

Config conf = new Config();

if (topologyName != null) {
    conf.setNumWorkers(3);
    StormSubmitter.submitTopology(topologyName, conf, builder.createTopology());
} else {
    conf.setDebug(true);
    LocalCluster cluster = new LocalCluster();
    cluster.submitTopology("test", conf, builder.createTopology());
    Utils.sleep(1000000);
    cluster.killTopology("test");
    cluster.shutdown();
}
                                                                    19,1          All
```

```
vaibhav@buzz-lightyear:~/Workspace/storm$ storm jar build/libs/gg-storm-SNAPSHOT.jar com.gumgum.storm.adevents.
AdEventsAggregationTopology AdEventsAggregation
```

STORM

TRIDENT

- Higher level abstraction
- Three types of Spouts
- Trident State
- Grouped Stream
- Functions, Filters
- Aggregators
- Query

STORM

TRIDENT TOPOLOGY

vaibhav@buzz-lightyear: ~

vaibhav@buzz-lightyear: ~ | vaibhav@buzz-lightyear: ~

```
TridentTopology topology = new TridentTopology();

SimpleKafkaSpout kafkaSpout = new SimpleKafkaSpout(Topic.AD_EVENTS.toString(), 5000);
topology.newStream("ad-events-by-image-kafka-spout", kafkaSpout)
    .aggregate(new Fields(DAY.name(), HOUR.name(), AD_ID.name(), VIEWS.name(), CLICKS.name()),
        new AdEventsByImageAggregator(), new Fields(DAY.name(), AD_ID.name(), VIEWS.name(), CLICKS.name()));

Config conf = new Config();

if (topologyName != null) {
    StormSubmitter.submitTopology(topologyName, conf, topology.build());
} else {
    conf.setDebug(true);
    LocalCluster cluster = new LocalCluster();
    cluster.submitTopology("test", conf, topology.build());
    Thread.sleep(2000000);
    cluster.shutdown();
}
                                                                    18,1          All
```

STORM

TRIDENT DRPC

- Distributed RPC
- Works on existing topology
- might require custom coding
- json output

vaibhav@buzz-lightyear: ~

vaibhav@buzz-lightyear: ~ | vaibhav@buzz-lightyear: ~

```
topology.newDRPCStream("words")
        .each(new Fields("args"), new Split(), new Fields("word"))
        .groupBy(new Fields("word"))
        .stateQuery(wordCounts, new Fields("word"), new MapGet(), new Fields("count"))
        .each(new Fields("count"), new FilterNull())
        .aggregate(new Fields("count"), new Sum(), new Fields("sum"));
~
:
```

```
DRPCClient client = new DRPCClient("drpc.server.location", 3772);
System.out.println(client.execute("words", "cat dog the man");
// prints the JSON-encoded result, e.g.: "[[5078]]"
```

STORM

OUR STORM CLUSTER

- 60 million events crunched per day
- 2 c1.xlarge Supervisors
- m1.large Nimbus
- 1 c1.medium zookeeper
- spring framework

STORM

PROBLEMS FACED

- OutofMemoryError - No Acks
- Problems with Transactional Kafka Spouts and Opaque Transactional Kafka Spouts
- Minor mistakes in documentation

RELATED PROJECTS

- Kafka Template and Spout on Git Hub
  https://github.com/vpuranik/kafka-utils
  https://github.com/vpuranik/storm-utils

- Write your bolts in R
  https://github.com/allenday/R-storm

GUMGUM

IMPORTANT EMAIL

careers@gumgum.com

Vaibhav Puranik
Principal Engineer
vaibhav@gumgum.com
@vpuranik