DISTRIBUTED STREAM PROCESSING

with

KAFKA

overview

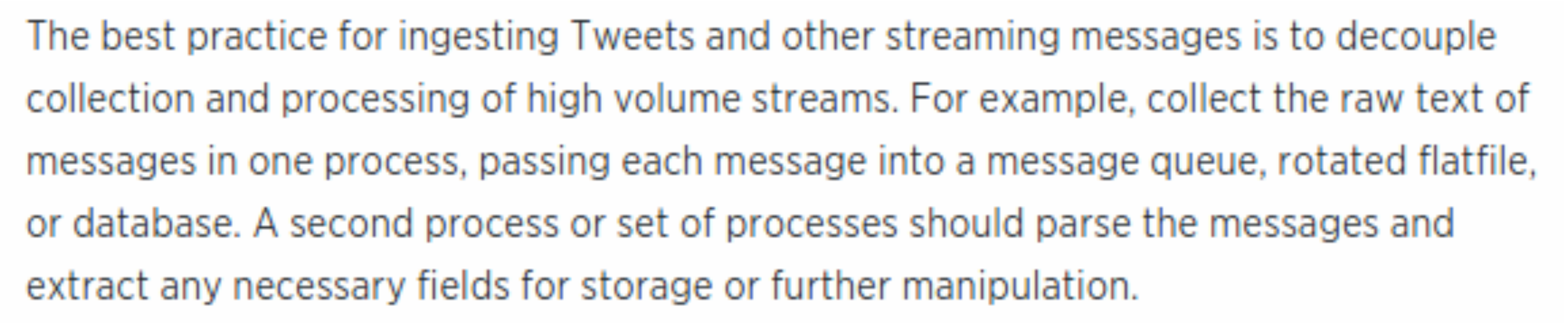
1. Motivation for use of Kafka
2. High-level explanation of Kafka
3. Example of Kafka being used
4. Demo / how to use Kafka.

How would you process data coming over a network in real-time?

How would your run complex NLP on the Twitter Streaming API?

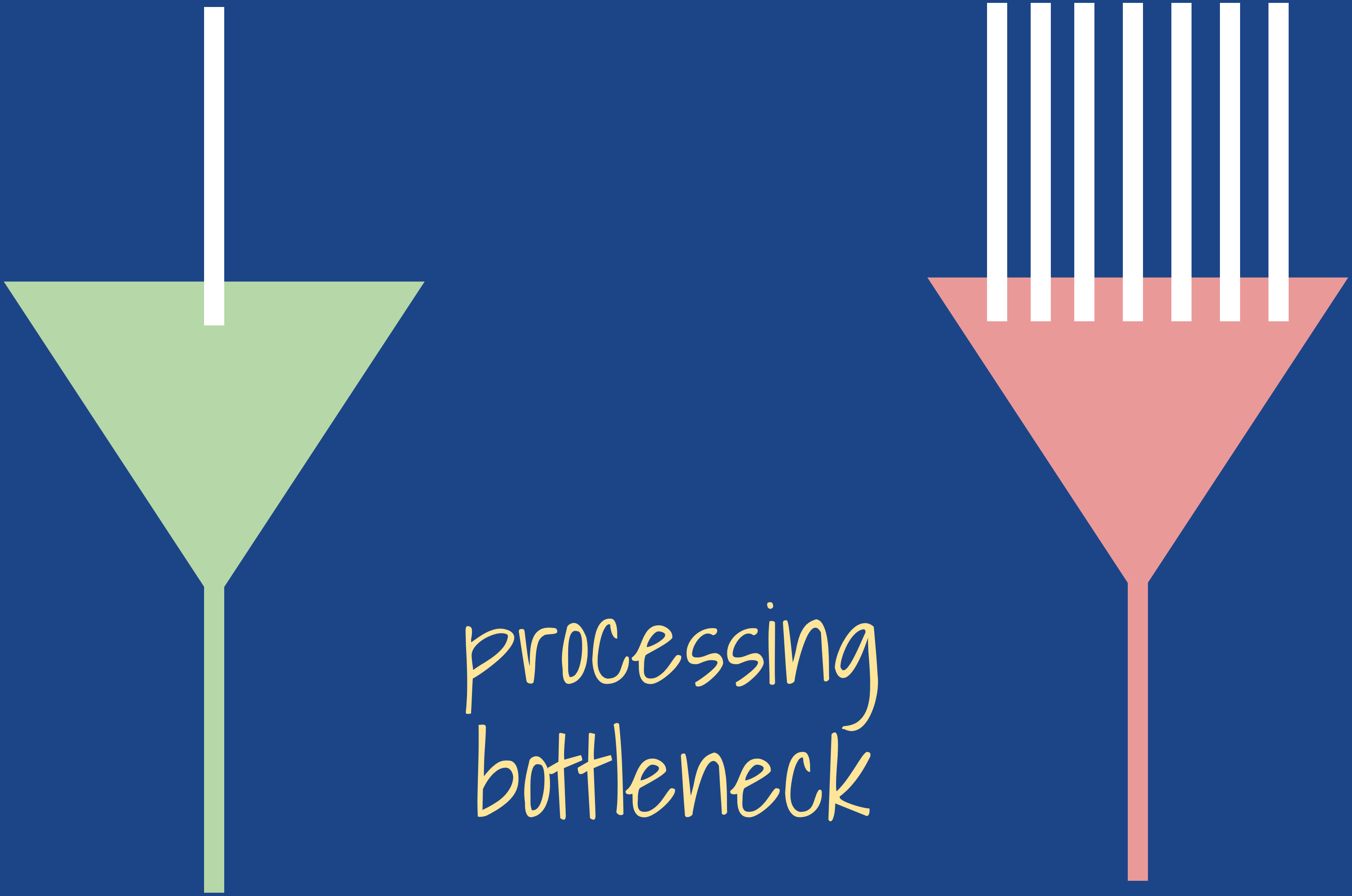
```
var Twitter = require(‘Twitter’);
var client = new Twitter();

// connect to stream
client.stream('statuses/kanye', function(stream) {

  // when a tweet is emmitted, process it
  stream.on('data', function(tweet) {
    complexNLP(tweet);
  });

});
```

https://dev.twitter.com/streaming/overview/processing

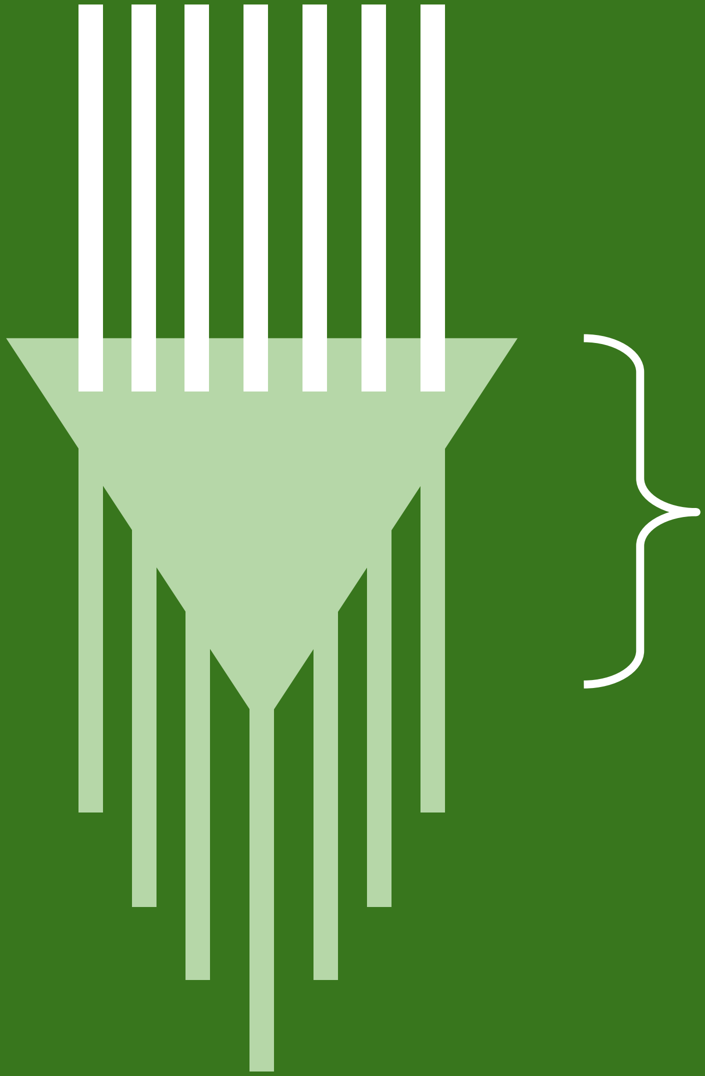


The best practice for ingesting Tweets and other streaming messages is to decouple collection and processing of high volume streams. For example, collect the raw text of messages in one process, passing each message into a message queue, rotated flatfile, or database. A second process or set of processes should parse the messages and extract any necessary fields for storage or further manipulation.

```
var Twitter = require(‘Twitter’);
var client = new Twitter();

var q = [];

Consume = function(q) {
    this.run = function() {
        client.stream(‘statuses/kanye’, function(stream) {
            stream.on(‘data’, function(tweet) {
                q.push(tweet);
            });
        });
    });
};

Process = function(q) {
    this.run = function() {
        while(!q.empty()) { complexNLP(q.shift()); }
    };
};

thread1.exec(new Consume(q).run());
thread2.exec(new Process(q).run());
```

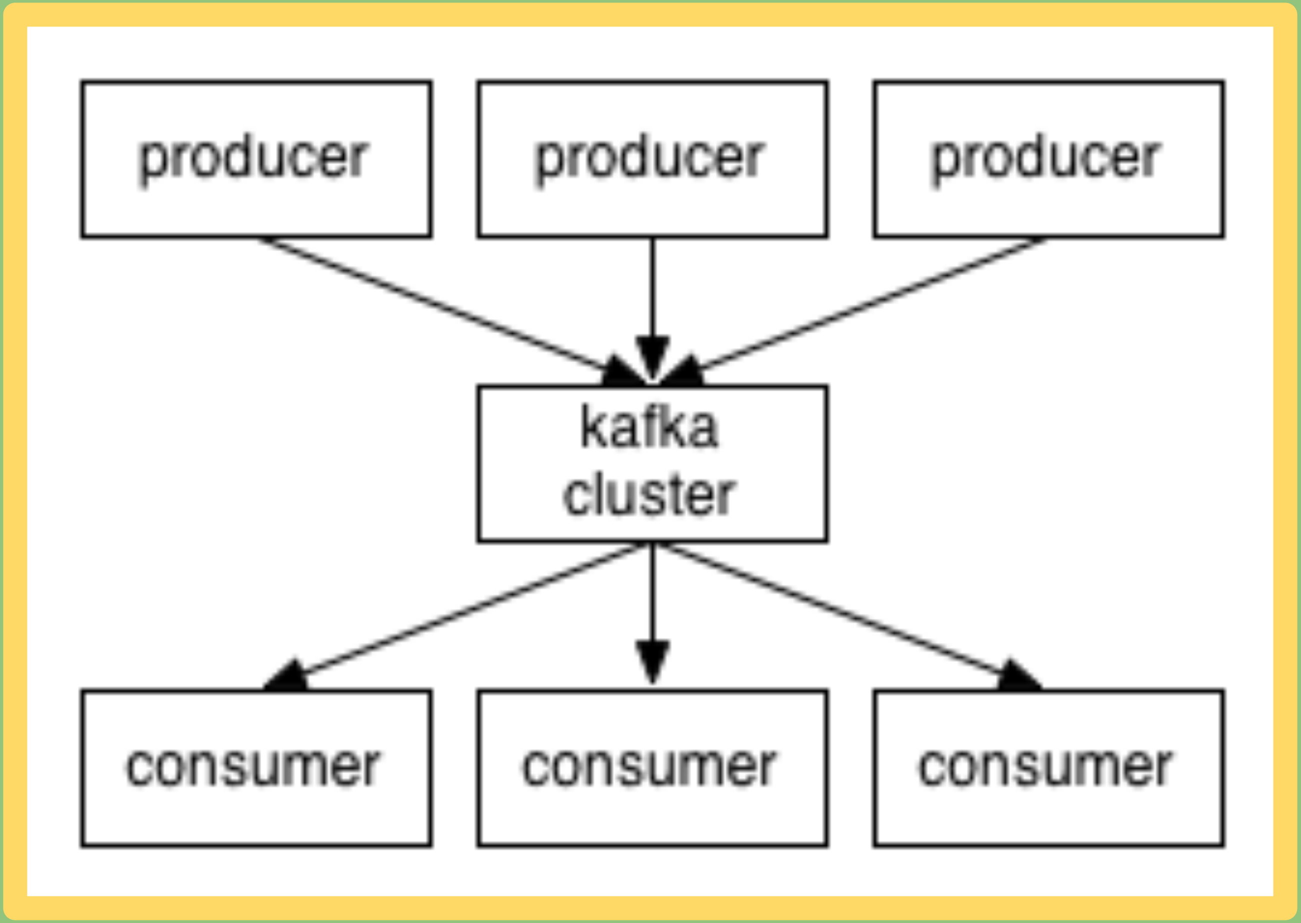


processing bottleneck



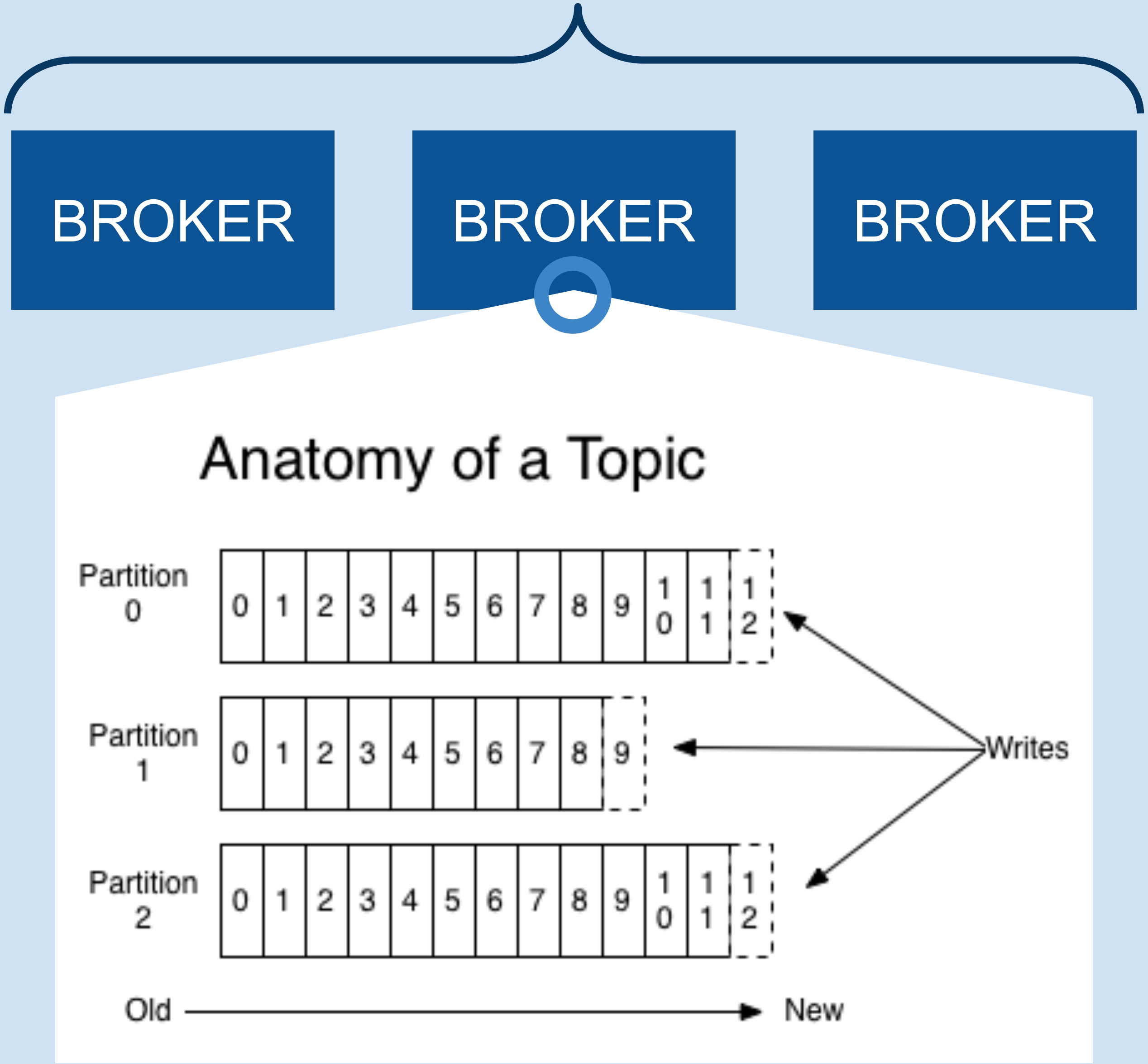
KAFKA

distributed
fault-tolerant
high-throughput
publish-subscribe
messaging system



producers publish, consumers subscribe

KAFKA CLUSTER

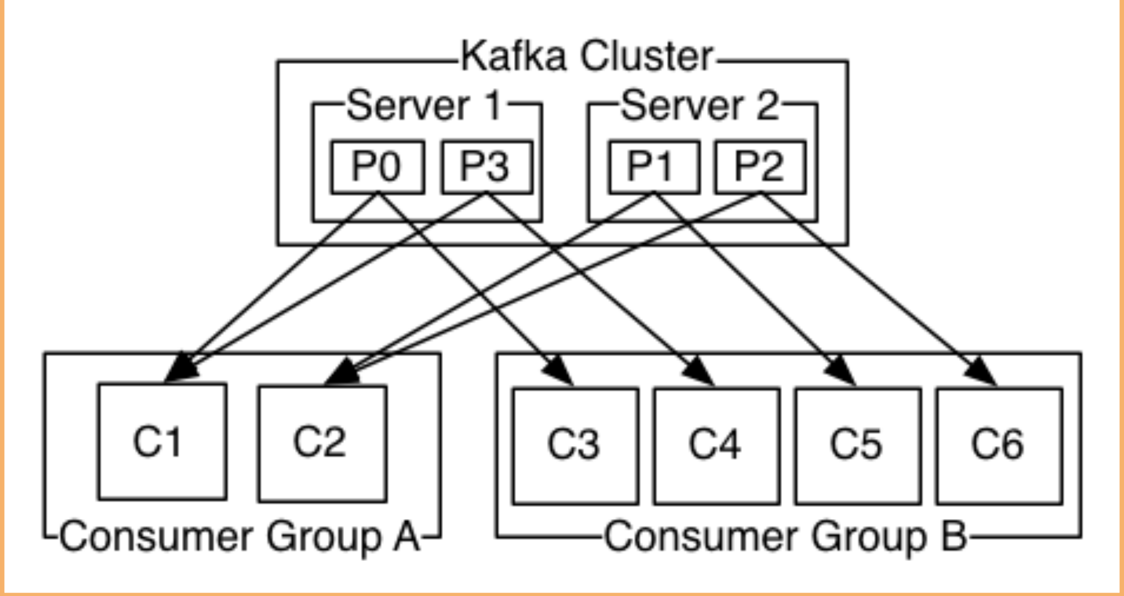


BROKER

BROKER

BROKER

- Consumers can subscribe to any set of partitions of any topic.
- Consumers keep track of where they are in the partition (offset).
- Consumer GROUPS are a provided abstraction to balance the load to different workers implemented in consumers.



Apache ZooKeeper

- Required for Kafka to run
- Another distributed system (fault tolerent)
- Coordinates flow of data between brokers

OTHER FACTS

- Kafka keeps all messages for up to N days
- Kafka is written in Scala
- Kafka is named after the writer, Franz Kafka

ALTERNATIVES

RabbitMQ

Flume

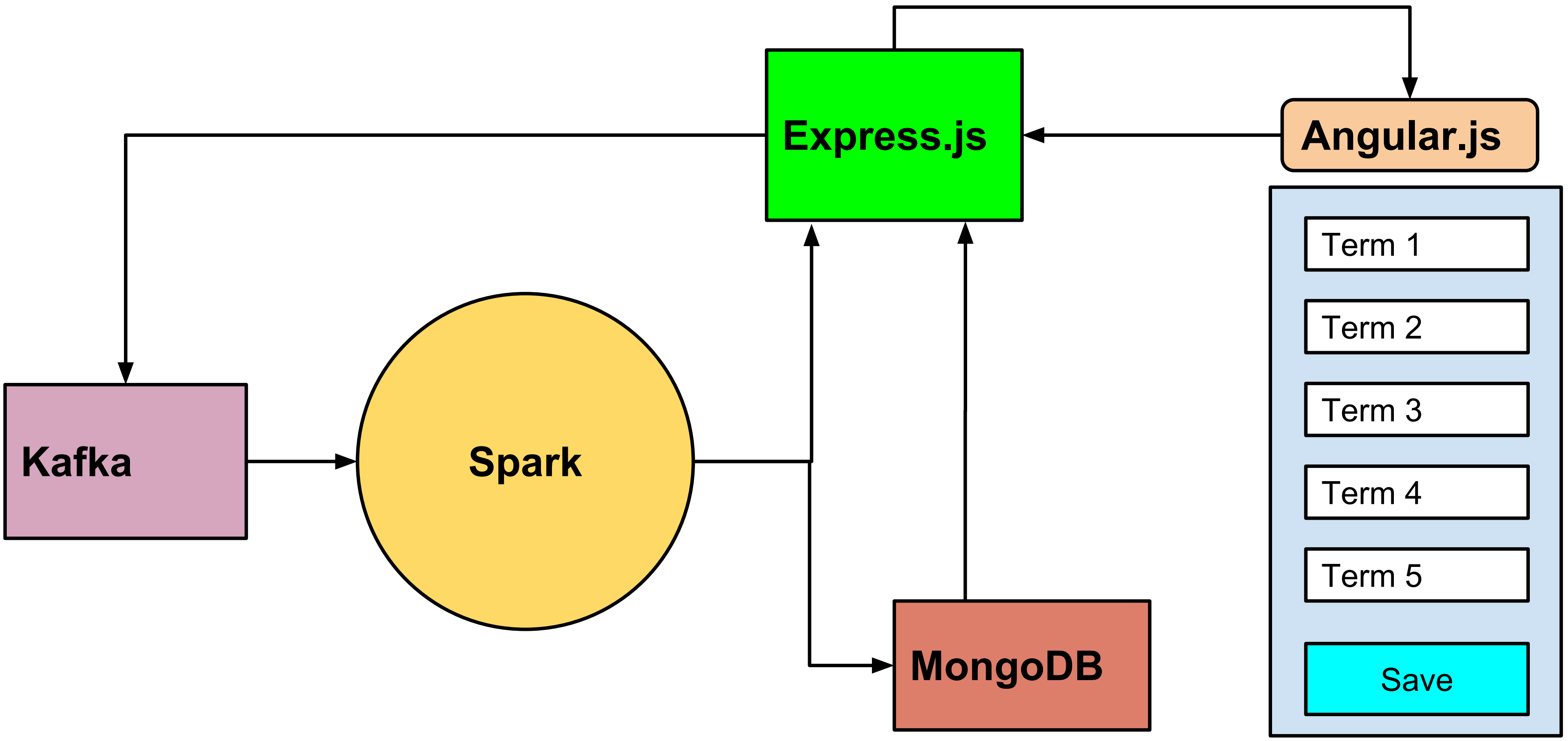
A Database

Redis pub/sub

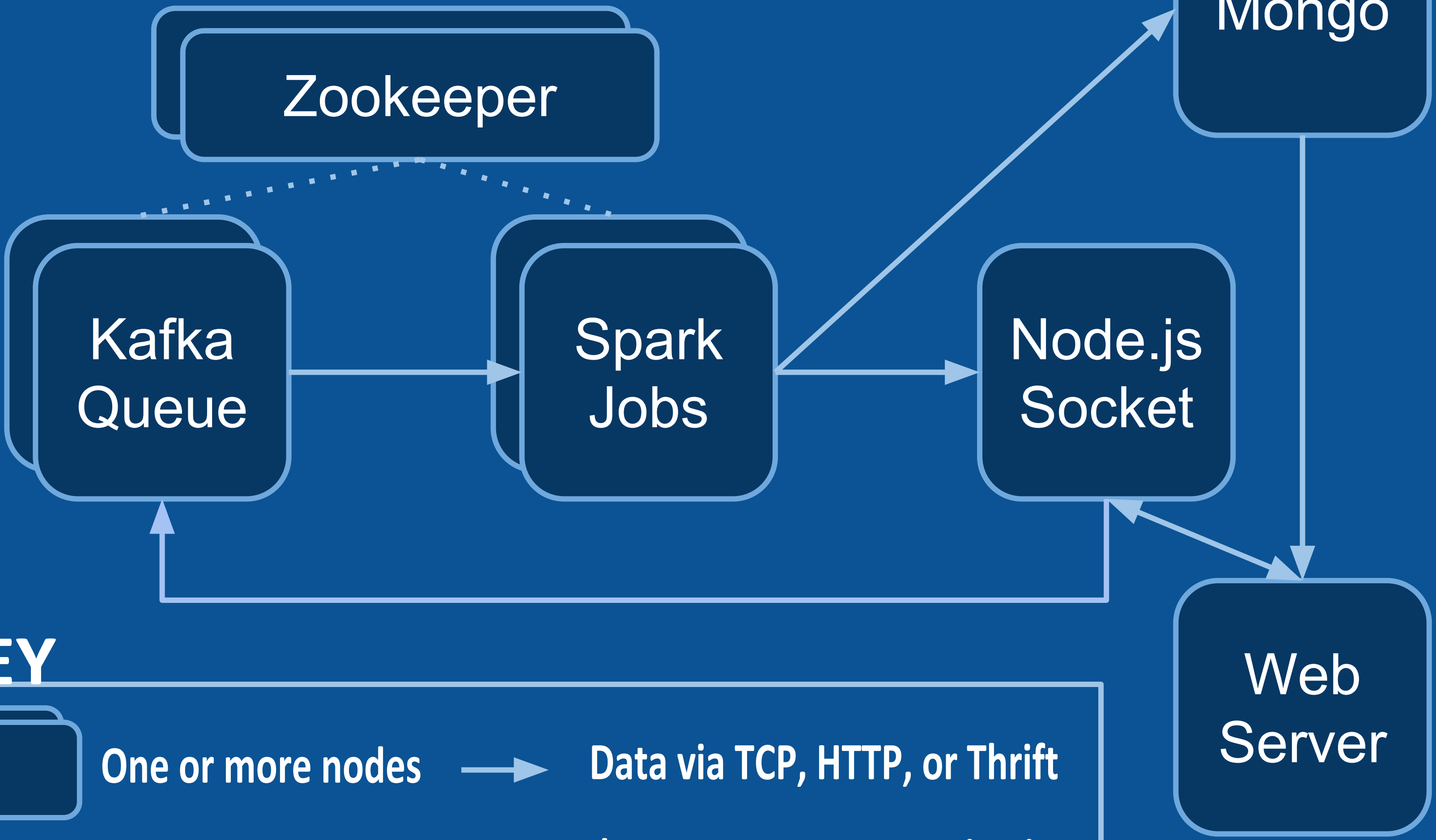
A Supercomputer

Example Architecture

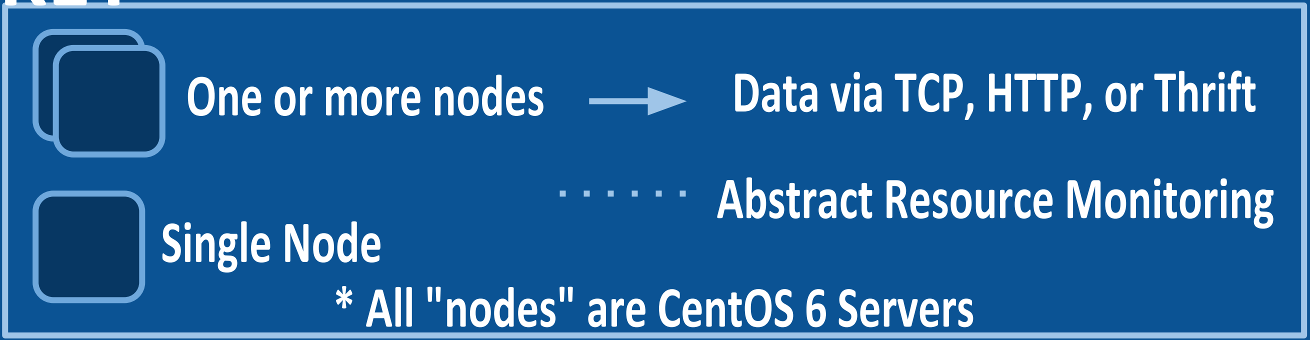
detecting civil unrest



Deployment



KEY



* All "nodes" are CentOS 6 Servers

Resources

Kafka: http://kafka.apache.org

ZooKeeper: http://zookeeper.apache.org

Twitter Streaming Guide: https://dev.twitter.com/streaming/overview/processing