

# Ohm-Wreckers' Solutions

MD5 Hackathon solution

Members: Ali Maredia, Srikar Nalleuri, Samir Hassam

# Problem Statement and Observation

- First responders are placed in situations of rescuing innocents and tackling tough decisions to retain the code of their duty, while infrastructure continues to crumble and there remains no communication, no resources, and no end in sight
- This was not just a dreamt nightmare but a reality that many faced during wake of Hurricane Sandy that made landfall on the east coast in fall 2012
- The storm left lasting impressions on the people of 13 counties in the NY/NJ area and left 346,000 homes destroyed or damaged and around \$4-5 billion dollars in damage

### Twitter vs. Facebook

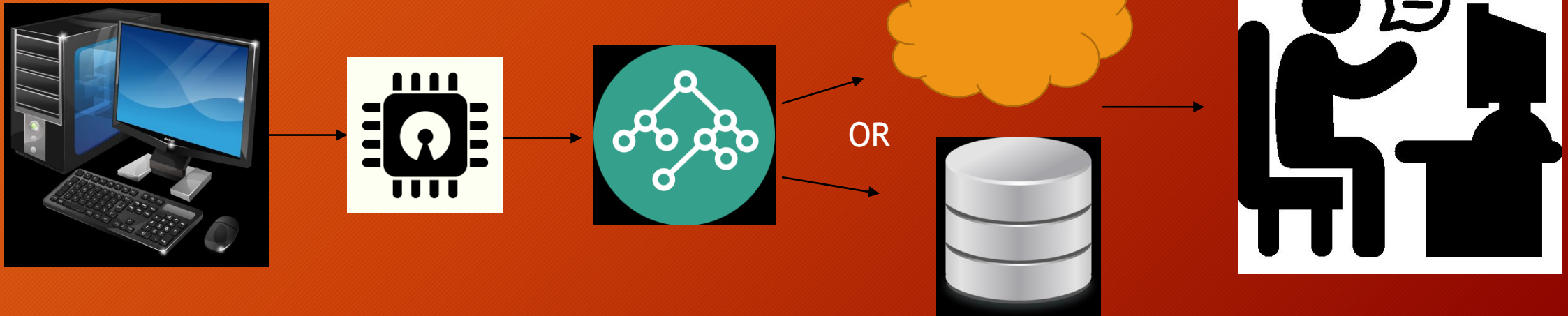
Twitter	Facebook
Tweets <140 characters	16,000+ characters
You can follow someone, but they don't have to follow you back	You friend someone and you are mutually connected
You can look at someone's public profile and see their tweets without following them	You can Like Brand Pages such as Gables, Coke, Oreos and receive their updates
You are shown every update in your Twitter feed	You are not shown every update in your Facebook timeline





# Proposed Solution

- Collecting raw data extracted from the public social media domain whereby posts are sorted by particular key words or phrases
- The issue that we wish to address is gathering the information sourced from social media that act as a crowd-sourced database of dynamic information that can improve ground operations for search, rescue, and recovery



# Solution Methods and Resources

- Use of the APIs allowed for the completion of a model process shown in the simulation ran by our computers using the JSON components



- Google Maps API
- Twitter API
- MongoDB
- Amazon Web Services
- Digital Ocean Servers



# Alternative approach to adjusted situation

- The adapted solution for the case that a reliable network is not available is the use of a technology created by Kobis Mobile.
- Kobis Mobile- company the provides reliable, portable technology for the proper collection and re-evaluation of audio, video, and photo sources

Kopis tactical gear and accessories



Kopis' Network  
Tactical tv module

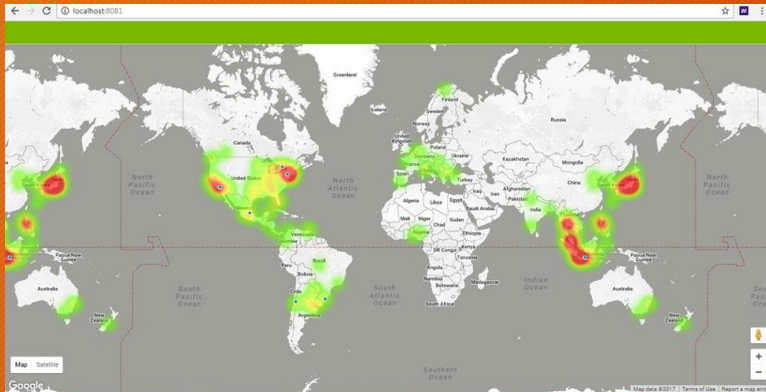


Kopis ESRA product



# Realization of Methods

## Image of Heat-Map Simulation Model of Active User Densities



## Image of Users in and around our concentrated market

Researchers estimated the age and gender distributions of 24 popular online social media communities in the United States (Pingdom, 2012). Regarding age, results revealed slightly more than half of users (51%) fall between the ages 25 and 44. Of the remainder, 21% are age 24 and younger, and 27% are age 45 and older. According to the same research, the average social media user across all 24 sites is almost 37 years in age. Users' average ages are highest on LinkedIn ( $M = 44.2$  years), followed by Facebook ( $M = 40.5$  years), and Twitter ( $M = 37.3$  years). Figure 2 below displays age distributions of users of all examined sites.

**Figure 2: Percentage of social media users by age\*.**



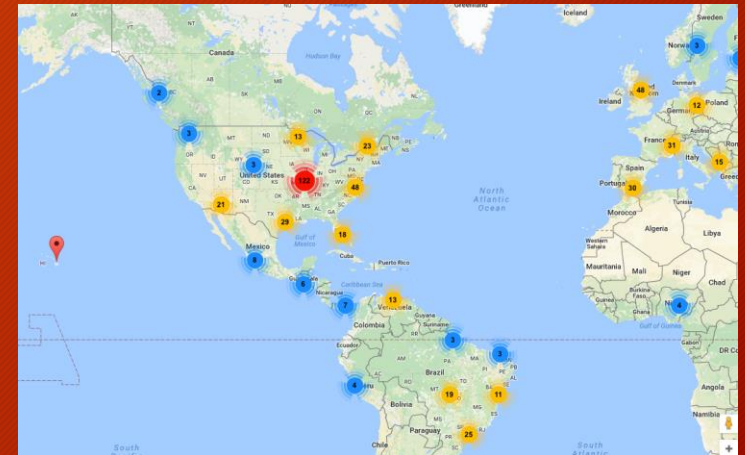
## Citation: Note:

Sites included in analysis were Facebook, Twitter, LinkedIn, Pinterest, Tumblr, Reddit, Hacker News, Slashdot, Github, Stack Overflow, Orkut, Quora, WordPress, Blogger, Flickr, Myspace, Tagged, Hi 5, LiveJournal, Yelp, deviantART, StumbleUpon, Goodreads, and Last.fm.

Source:

Pingdom.com (Pingdom, 2012)

## Image of Markers Simulation Model of Active User Densities





# Cost/Benefit Analysis

- Overview of various expenses that can be incurred if the necessary need and capacity arises
- This current moment expenses are limited to about \$78/year (including all operation costs)

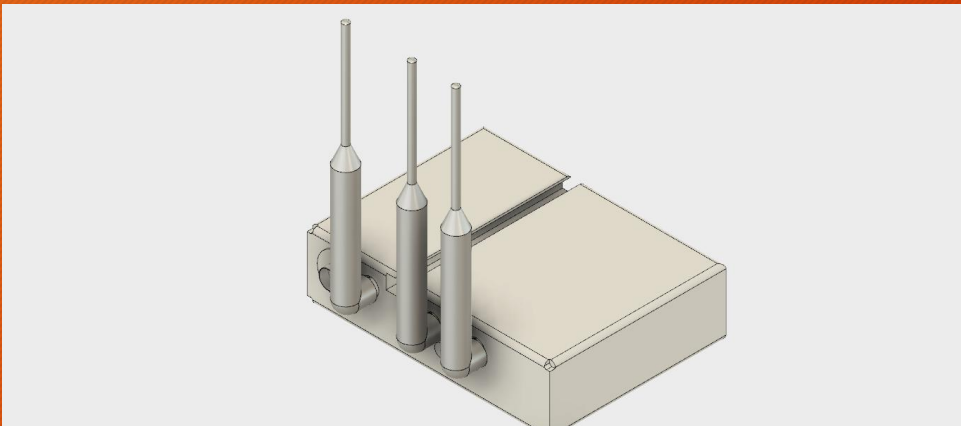
Includes the use of a domain and smaller web-server

Company Names	Amazon Web Services	Twitter API (enterprise package)	Kopis Mobile <u>LLC</u>	IBM Watson Services
Number of products/Services purchased	Lambda or Kinesis server-(1)	Package use-(1)*	Tactical Mobile System + MC package (2)	1 or 2*
Cost of using such things	+\$100 (attained credit)	Provided upon request	\$1413.25	\$20.00/(n-1) classifiers and \$0.0035/(n-1000) calls

\*Services could potentially be used in the future for greater datasets

# Formulated Model/Design

- The following presentation will demonstrate some of the data that we collected and the process by which a simple query can sprawl out and gather an active user base
- The demonstration includes two components: a demonstration of our software application (primary solution) and a overview of our hardware design (secondary solution)





# Strategic Conclusions

- Things that we learned and understood from the process
- Things that we felt were challenging
- Valuable feedback we received from mentors
- Something that could be more improved upon

# Cited Sources and Documentation

1. Google Maps API- Google Maps JavaScript API | Google Developers. (n.d.). Retrieved February 24, 2017, from <https://developers.google.com/maps/documentation/javascript/> Utilized to create map overlay of data using the Google Maps custom layout and tools
2. Twitter API- T. (n.d.). REST APIs — Twitter Developers. Retrieved February 24, 2017, from <https://dev.twitter.com/rest/public> Congregated the information gained from the live twitter feed to report specific trends and events occurring within specific regions
3. Bootstrap- Buytaert, D. (2016, August 29). Bootstrap Library. Retrieved February 25, 2017, from [https://www.drupal.org/project/bootstrap\\_library](https://www.drupal.org/project/bootstrap_library) Sourced this library to use the tools that created a desirable user interface
4. Socket.io- M. (2017, February 22). Socket.io. Retrieved February 24, 2017, from <https://github.com/socketio/socket.io> Used to interface between the JSON configuration and the web-server to source scripts and commit actions
5. jQuery- Jquery.org, J. F. (2017). JQuery API. Retrieved February 26, 2017, from <http://api.jquery.com/> Provided a dynamic library to execute commands that created a more visible and dynamic interface
6. Vex- N. (2017). Vex-js. Retrieved February 26, 2017, from <https://www.npmjs.com/package/vex-js> Included the formatting properties of the Javascript structure to take into account the addition of html elements
7. Twit- Twit-old. (n.d.). Retrieved February 25, 2017, from <https://www.npmjs.com/package/twit-old> Used to structure the api information sourced for our problem solving method with the twitter api