Distributed Web Scraping

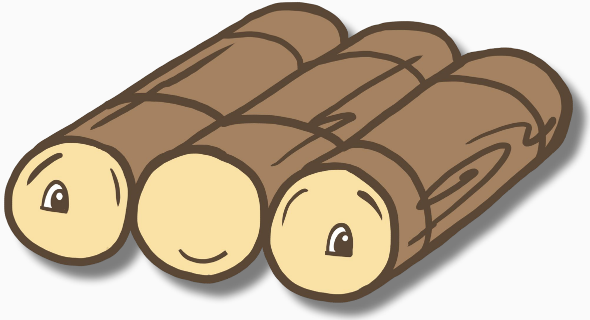
By Michail Roesli, Quinn Gieseke, and Blake Smith

Problem

Web scraping as a service

- Utilize idle phones
- Citizens can support organizations
- Improve performance of systems

Using RAFT and Phones together!



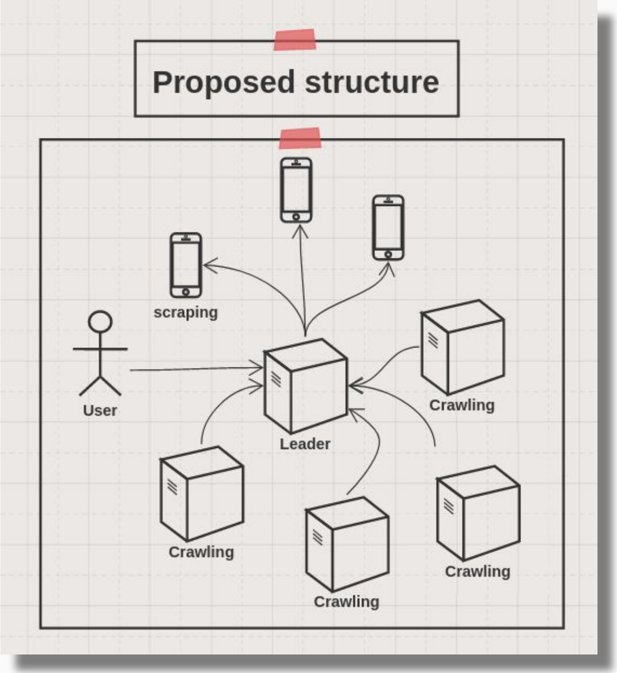
Proposal

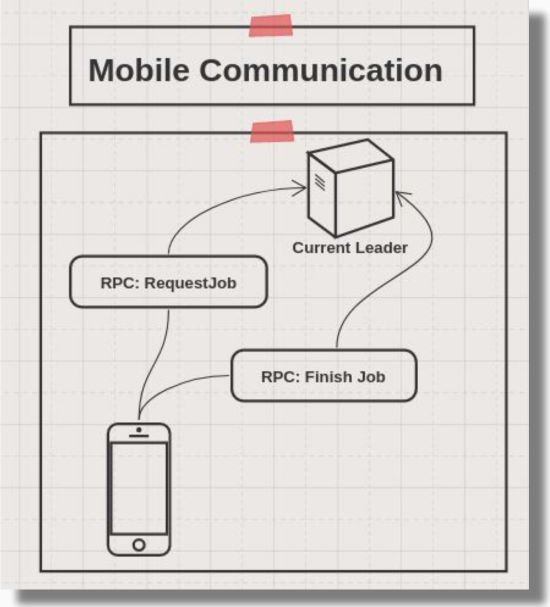
Use RAFT to connect computers and mobile devices into a reliable distributed network

Leader - communicates between phones

Followers - crawl the provided urls

Phones - scrape urls collected





Web Scraping Structure

1. Start Phone application to connect to raft leader
2. Provide function to be executed to leader - hardcoded initially
   - Eventually be able to provide either by phone or some client
   - Runs with interpreter for phones, or some JVM language like Java
3. Leader sends info of connections to crawlers and updated urls to crawl
4. Phones request URLs crawled from the leader
5. Leader provides URLs and function to execute to phones
6. Safe failure from phones going down

Web Scraping Structure Libraries

- Hosting on Google Cloud with $300 credit
- Using GRPC for communication
- RAFT
- JSON
- JVM languages
- Kotlin

Functions

- Word occurrences (first attempt) - A/B testing
- Link frequency
- Images
- Word-relation maps
- Unique sentences
- Tweet maps
- Geo-tagged data for clustering

And all sorts of cool machine learning applications

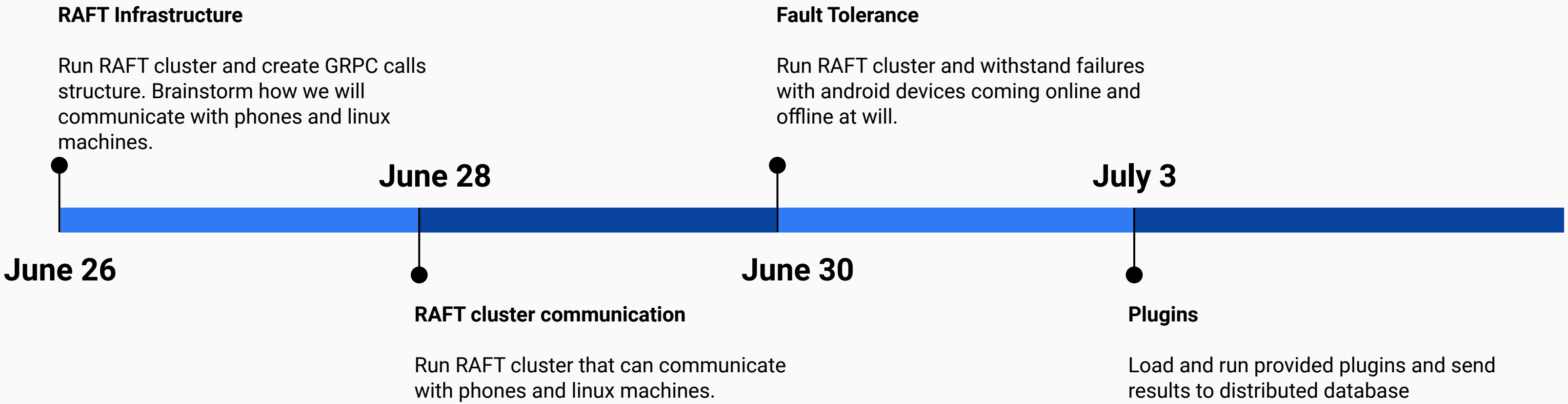
Related work

Folding@home

- distributed computing project for simulating protein dynamics
- helping scientists to better understand biology
- providing new opportunities for developing therapeutics

Distributed Web Scraping

- Distributed computing project for crawling websites and doing data gathering
- Helping organizations with performance for more efficient data acquisition



June 26

RAFT Infrastructure

Run RAFT cluster and create GRPC calls structure. Brainstorm how we will communicate with phones and linux machines.

June 28

RAFT cluster communication

Run RAFT cluster that can communicate with phones and linux machines.

June 30

Fault Tolerance

Run RAFT cluster and withstand failures with android devices coming online and offline at will.

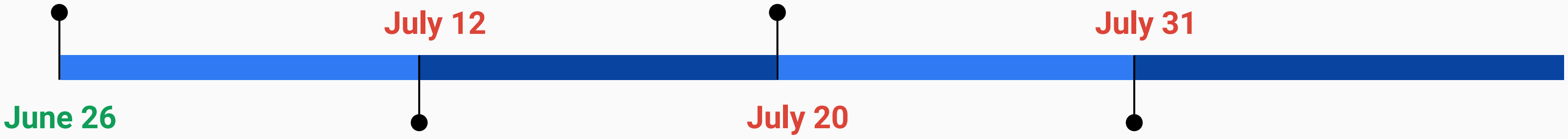
July 3

Plugins

Load and run provided plugins and send results to distributed database

Evaluation - Phase I

Evaluation (Reevaluated) - Phase 2



June 26

RAFT Infrastructure

Run RAFT cluster and create GRPC calls structure. Brainstorm how we will communicate with phones and linux machines.

July 12

RAFT cluster communication

Run RAFT cluster that can communicate with phones and linux machines.

July 20

Fault Tolerance

Run RAFT cluster and withstand failures with android devices coming online and offline at will.

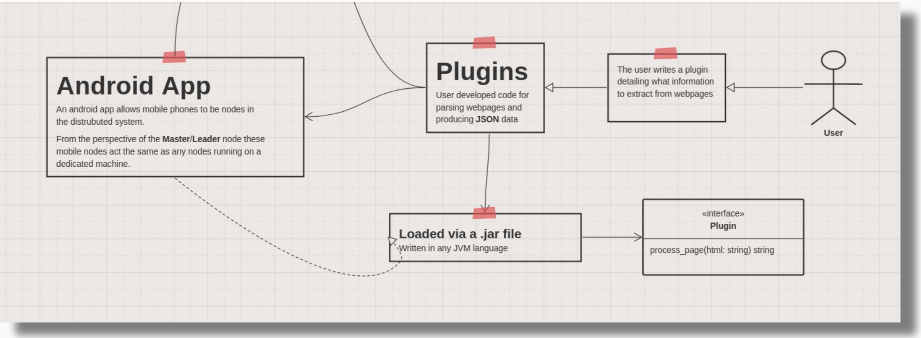
July 31

Plugins

Load and run provided plugins and send results to distributed database

Next Phase

- Dynamic crawling to acquire URLs to scrape instead of using a fixed list of URLs
- Collection of data into a scalable distributed database such as Cassandra
- Multiple client jobs running over the cluster simultaneously



Android App

An android app allows mobile phones to be nodes in the distrubuted system.

From the perspective of the **Master**/**Leader** node these mobile nodes act the same as any nodes running on a dedicated machine.

Plugins

User developed code for parsing webpages and producing **JSON** data

The user writes a plugin detailing what information to extract from webpages

Loaded via a .jar file

Written in any JVM language

«interface»
Plugin

process_page(html: string) string