Szukanie danych GPS w Firebase

GDG Rzeszów #4 - Firebase Day

Ja!

Nazywam się Andrew Torski

Fullstackuję @ yawn.wtf

Cloud & Firebase Specialist @ GDG Warszawa

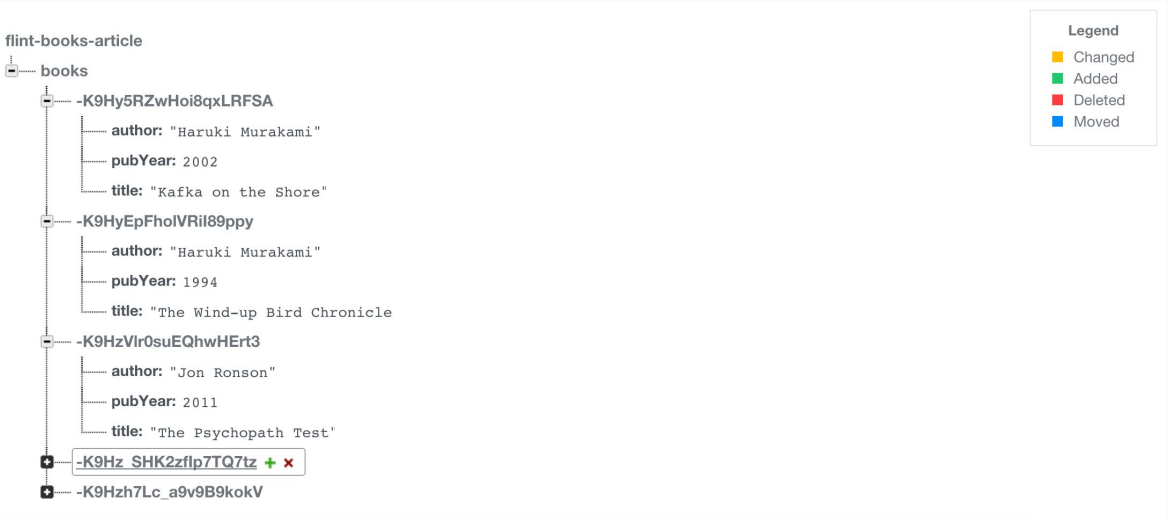
A dzisiaj także speaker @ GDG Rzeszów #4 ;)

Dlaczego  ?

Wiele narzędzi do monetyzacji, konwersji, hostingu…

Ale też bardzo fajna baza danych!

… która na bieżąco nas informuje o zmianach.

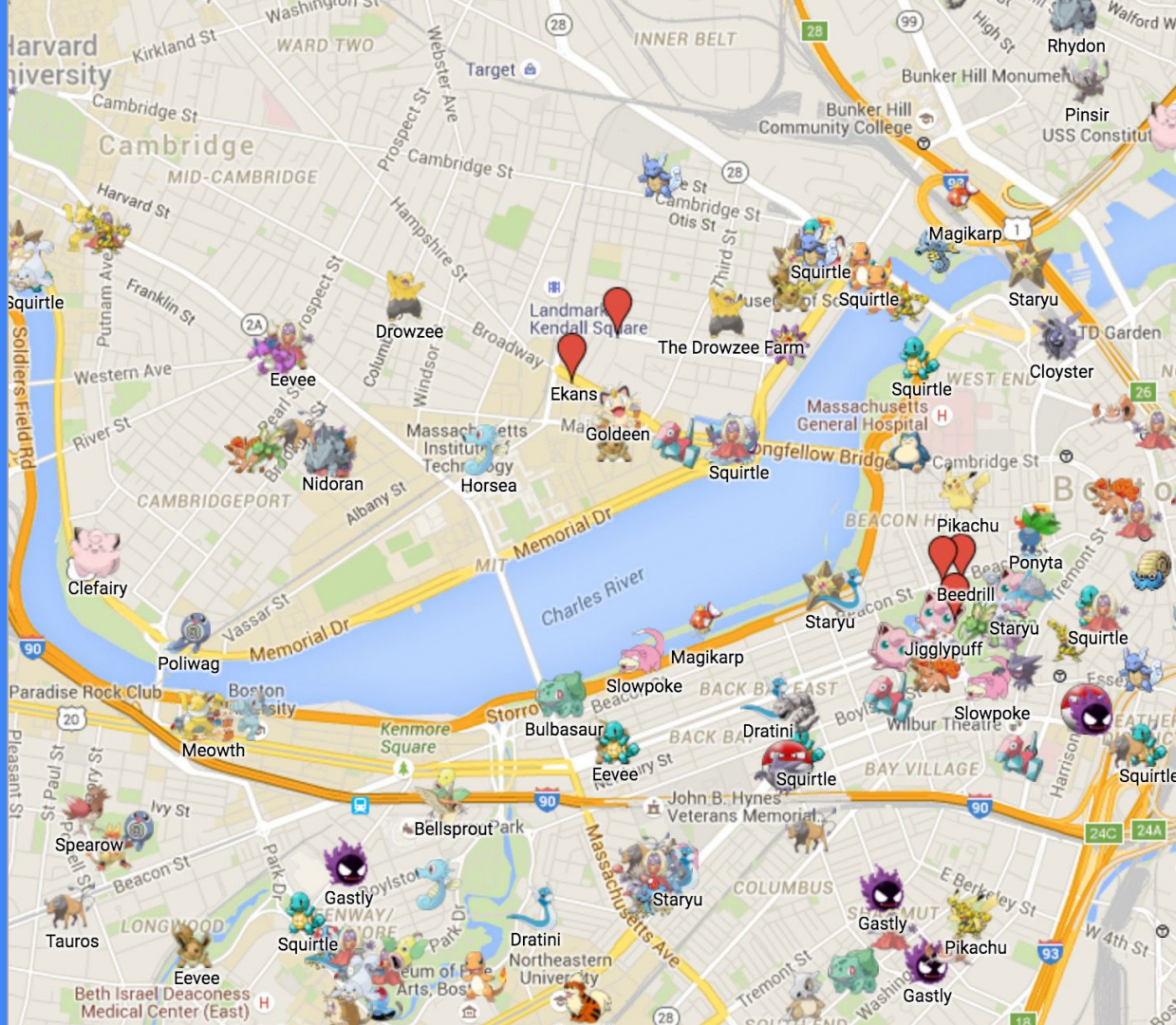


Jeden epicko duży JSON!

Appka na

Na mobilkach zaznaczamy gdzie zobaczyliśmy pokemona.

Inni użytkownicy mogą na bieżąco obserwować gdzie się pojawiają lub się pojawiły Poksy.



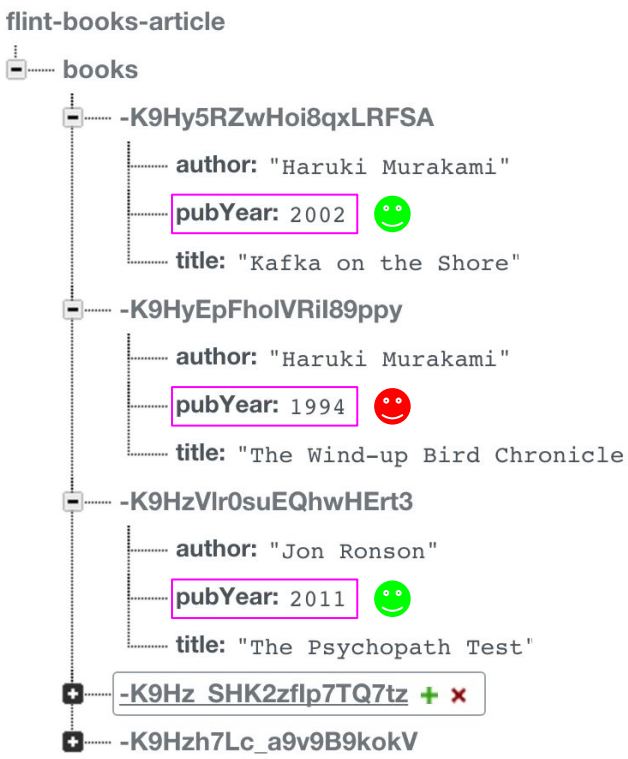
Specyfika bazy danych

Jest to quid pro quo:

Wszystko jest *real time* i możemy sortować i filtrować ale…

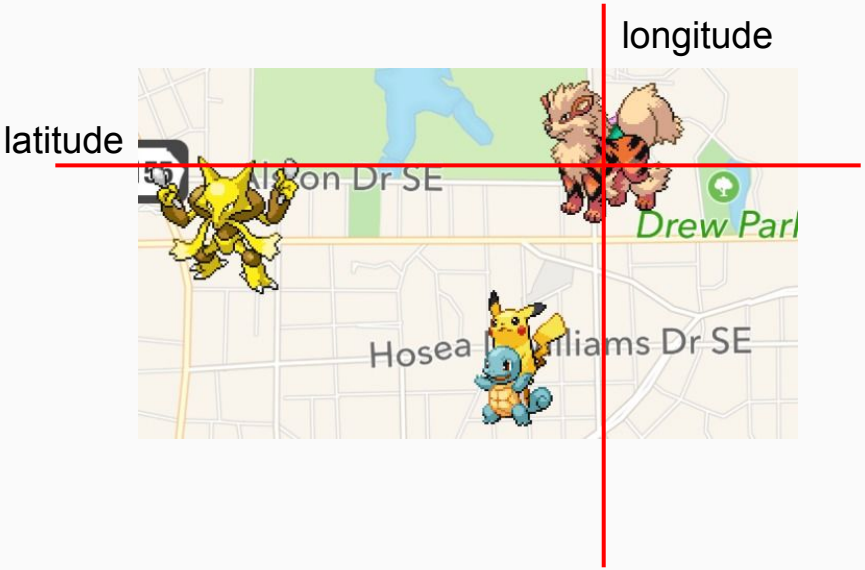
...filtrujemy tylko po jednym kluczu...

```
pubYear > 2000 && title.beginsWith("The")
```



Jak się to ma do naszych Poksów?

Struktura widzianego Poksa



-{klucz}

-pokemon: Arkanite,

-lat: 52.23,

-lng: 20.34,

-time: 1231543564 // unix timestamp

Co chcemy zrobić?

Na mobilku pobrać wszystkie poksy z obszaru mapy, którą akurat widzimy z ostatniego dnia.

Mamy do dyspozycji:

- Długości i szerokości rogów widocznego skrawka mapy,
- Czas,
- Firebase Realtime Database,
- Chęć i determinację by to zrobić.


```
northeast
  lat = 52.10
  lng = 21.30
```

```
southwest
  lat = 50.10
  lng = 20.30
```

Konwencjonalny SQL

```
SELECT *
FROM pokemon_sightings
WHERE lat >= southwest.lat
   && lng >= southwest.lng
   && lat <= northeast.lat
   && lng <= northeast.lng
   && time > 24h temu
```

Brak real time!

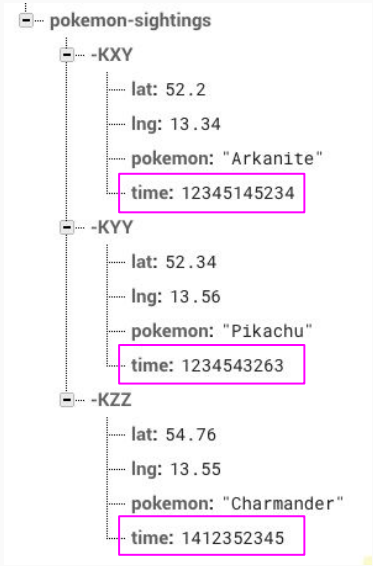
northeast
lat = 52.10
lng = 21.30

southwest
lat = 50.10
lng = 20.30

A jak użyjemy Firebase’a?

Rozwiazanie 1 - dwa razy filtrujemy

Wybieramy **jedno** pole po którym sama baza danych wyfiltruje.



Tym polem będzie timestamp!

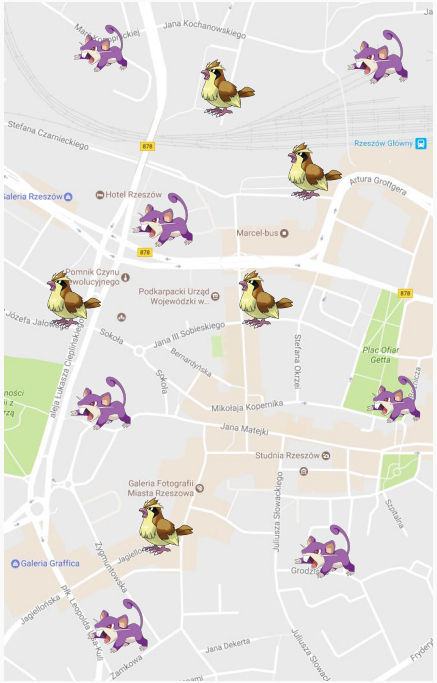
```
poke-ref
      .orderByChild(“time”)
      .startAt(dayAgo)
      .addChildValueListener(...)
```

Następnie ręcznie filtrujemy te poksy, które nie są widoczne.

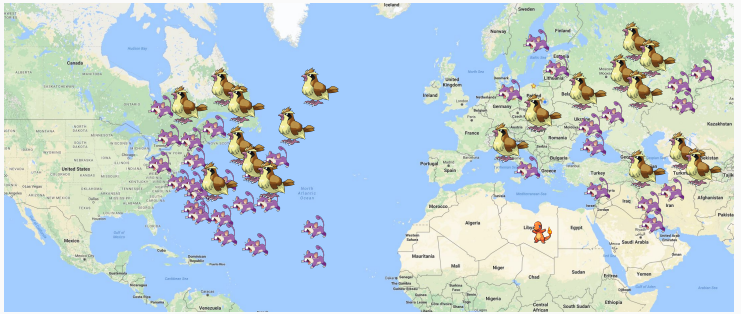
+ i -

+ Realtime!
+ Prosty zapis
+ Prosty odczyt

- Zaciągamy wszystkie poksy z całego globu z ostaniego dnia!



Chcę Rattaty i Pidgeye tylko z Rzeszowa!



… a dostaję je z całego świata!

Potencjalnie pobieramy tysiące obiektów, na które użytkownik nigdy nie spojrzy…

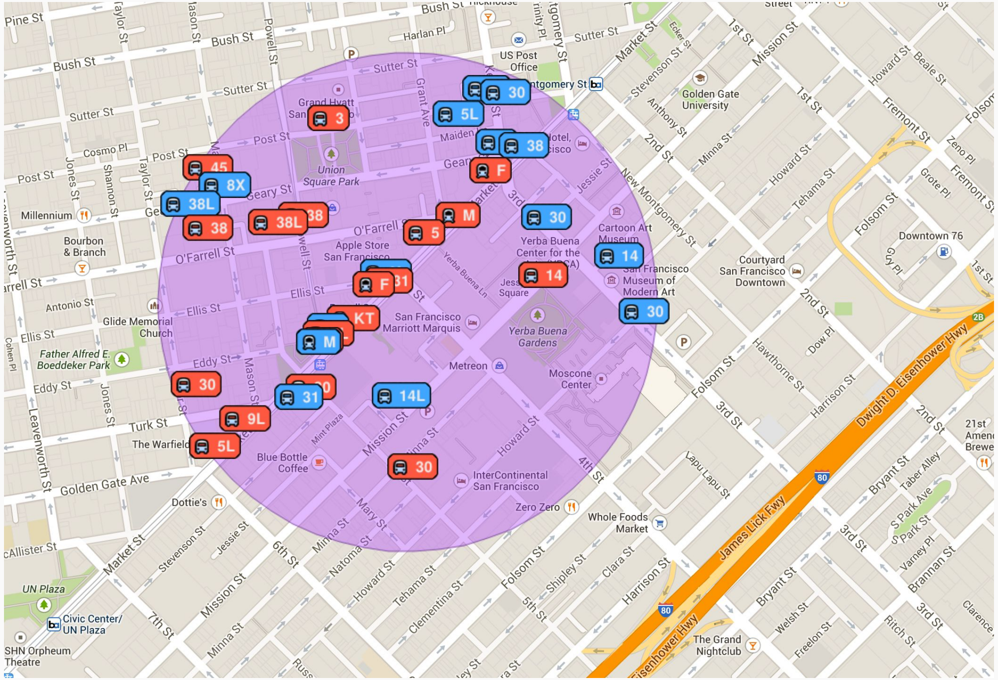
Rozwiązanie 2
GeoFire

https://github.com/firebase/geofire

Biblioteka na web, Android i iOS.

Pozwala na zapis danych GPS i tworzenie zapytań w stylu:

>“Poproszę wszystkie dane w odlegości Z od punktu X,Y…”

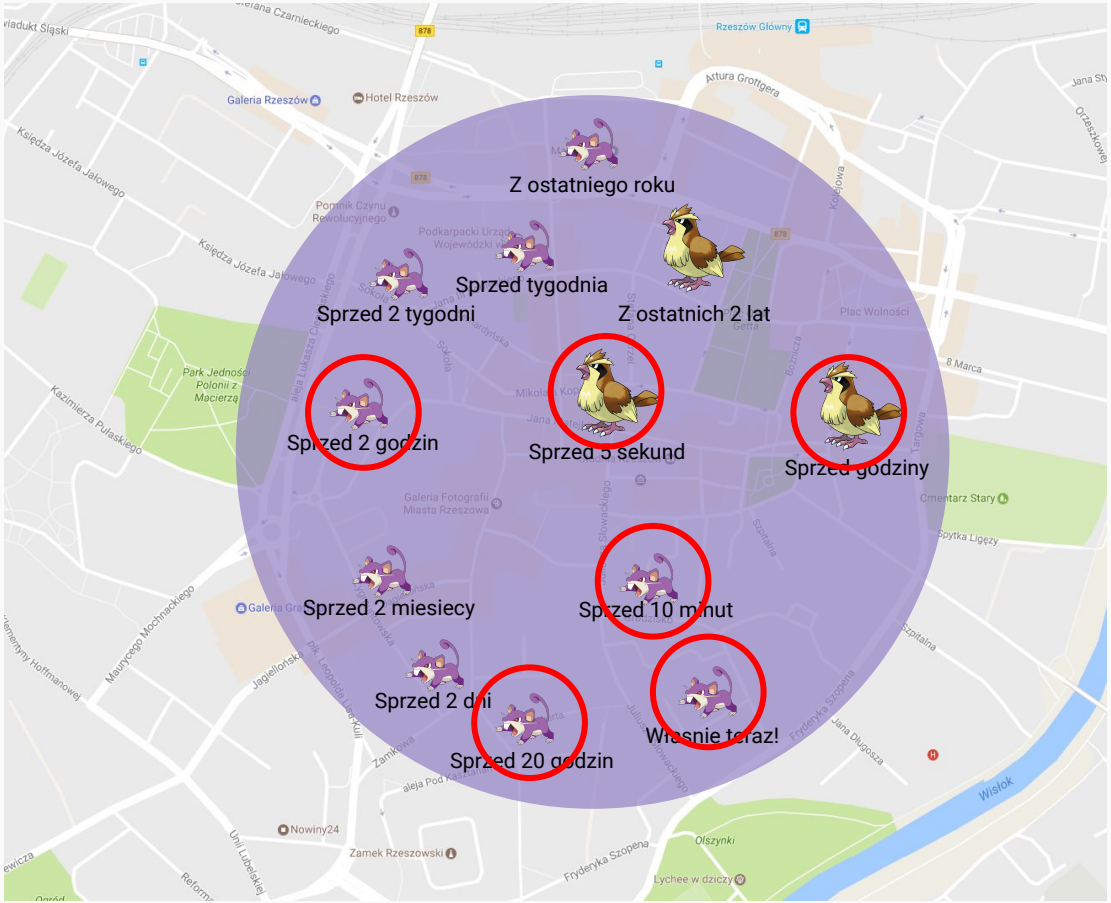


https://firebase.googleblog.com/2014/06/geofire-20.html

GeoFire

1. Chcę Poksy w odległości 500m od punktu 50.03, 22.00

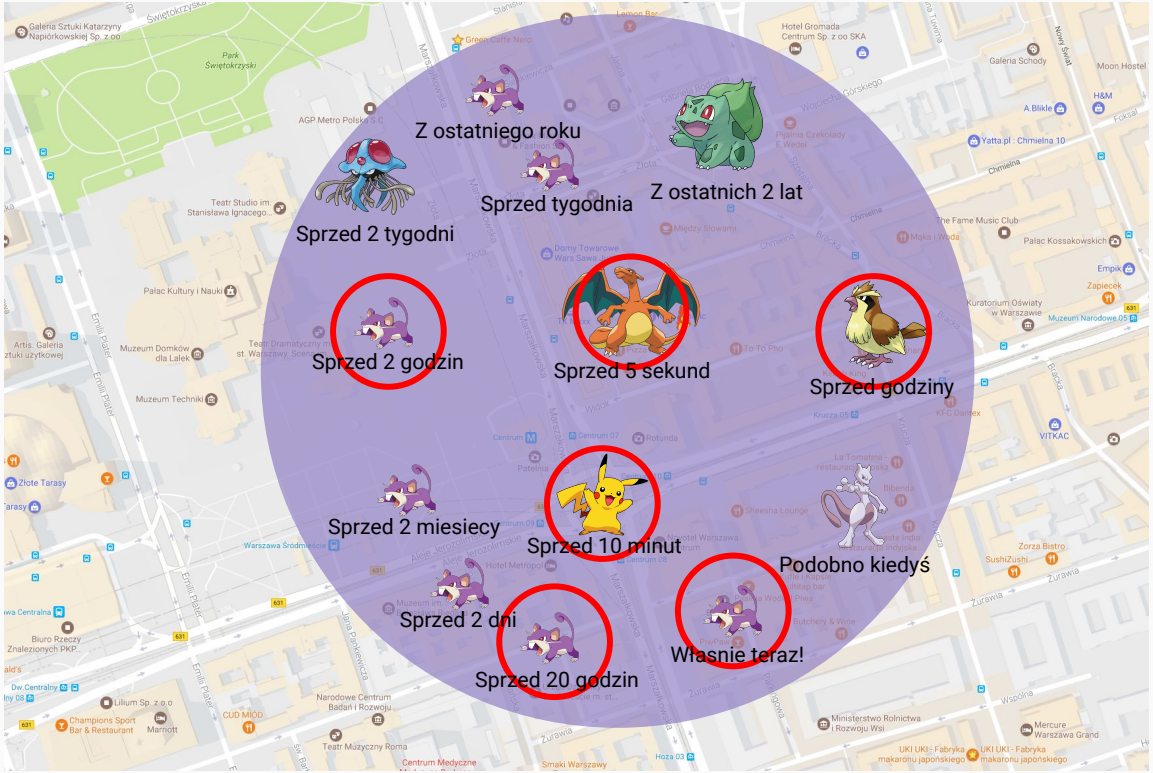
2. Ręcznie filtrujemy Pokemony sprzed 24h



+ i -

+ Realtime!
+ Pokemony tylko z widocznego obszaru
+ Zapis i odczyt wspomagany przez libkę

- Dostajemy wszystkie pokemony od początku i musimy filtrować



Jest lepiej niż wcześniej, ale nadal możemy pobierać nie potrzebne dane…

Geohashe

Co to?

{50.0375926, 22.0040021} → [u2zsstnf]

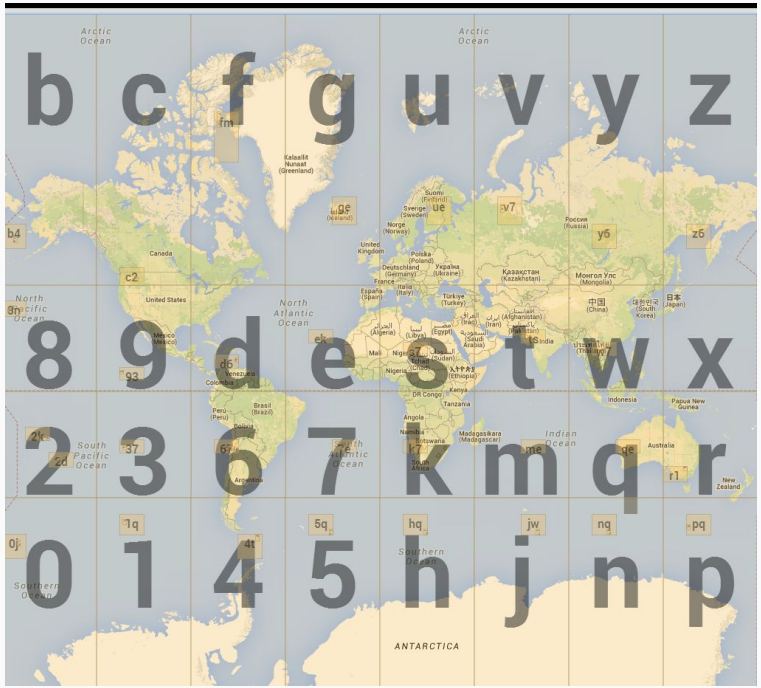
Estrada Cafe

Sami ustalamy długość

Odwrotnie też można!

[u2zsstnf] → {50.0375926, 22.0040021}

https://en.wikipedia.org/wiki/Geohash, http://www.movable-type.co.uk/scripts/geohash.html, xkcd.com/426/



Geohash - jak to wyglada!

W praktyce?

Cały świat jest podzielony na 32 prostokąty.

A z kolei te prostokąty na jeszcze 32 prostokąty…

I tak dalej…

https://chrysohous.files.wordpress.com/2012/09/300px-maidenhead_grid_over_europe.png

Im dłuższy tym większa dokładność

u
u2
u2z
u2zs
u2zss
u2zsst
u2zsstn
u2zsstnf

http://geohash.gofreerange.com

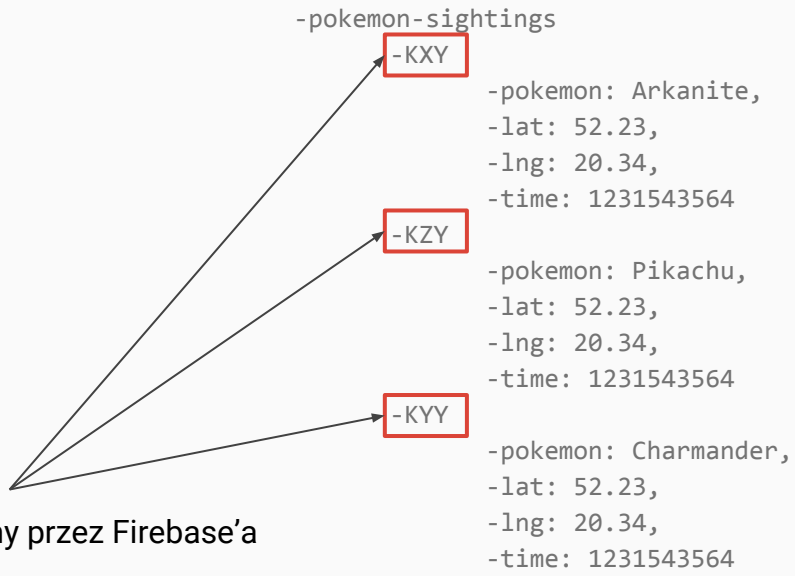
Jak tego użyjemy?

Struktura

Sam Pokemon się nie zmieni...

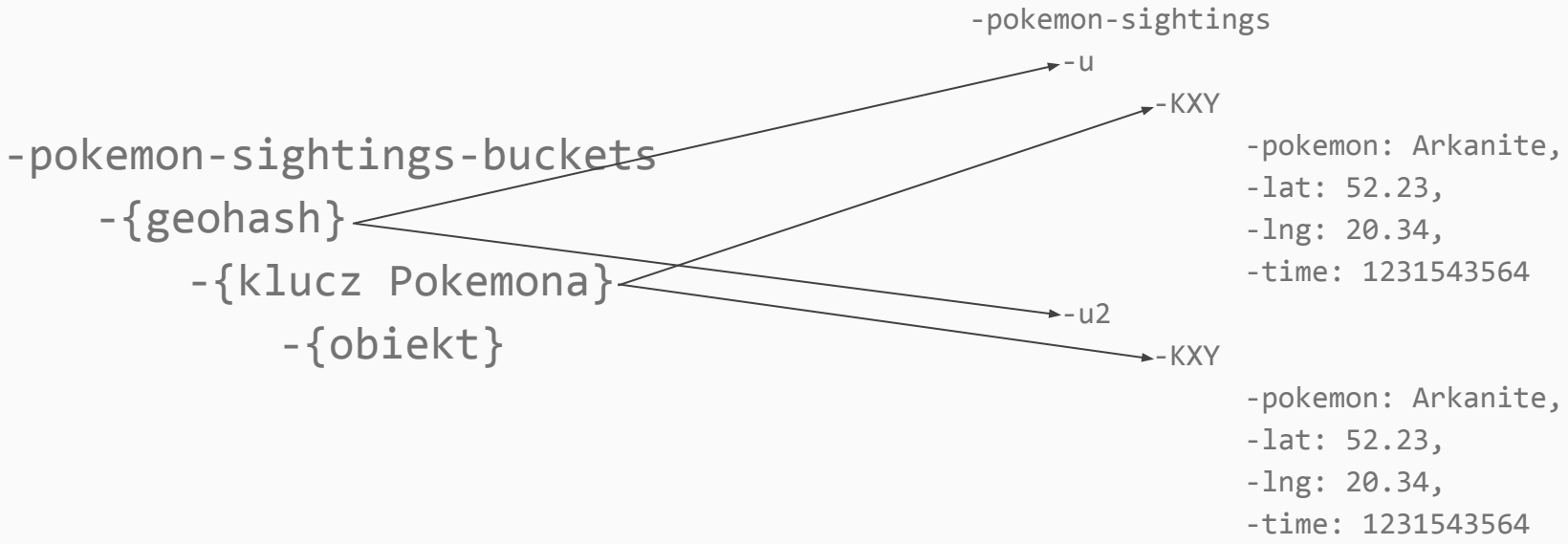
```
-{klucz}
    -pokemon: Arkanite,
    -lat: 52.23,
    -lng: 20.34,
    -time: 1231543564
```

Ale struktura, w której jest trzymany już tak



Klucz generowany przez Firebase'a

Nowa struktura



Zapis - Geohashe

1. Pojawia się Pokemon:

```
-pokemon: Arkanite,
-lat: 50.0375926,
-lng: 22.0040021,
-time: 1231543564
```

2. Generujemy geohash - 6 znaków

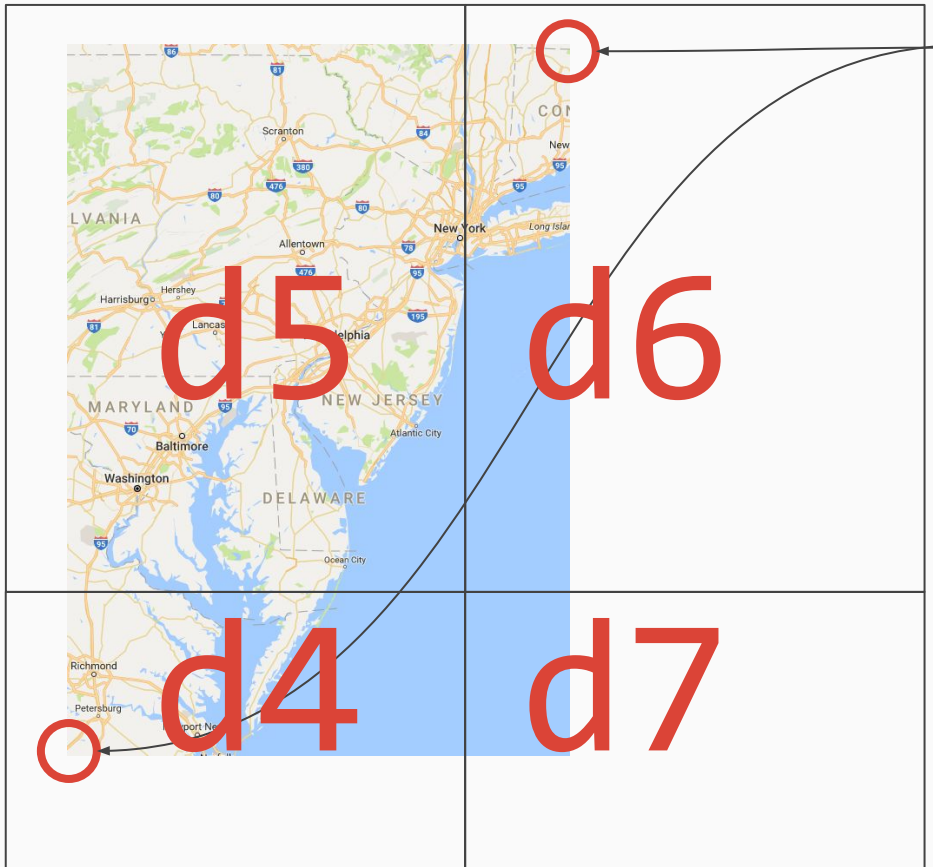
```
u2zsst
```

3. Zapisujemy go do kolejnych dłuższych geohashów

```
-pokemon-sightings-buckets
      -u
            -KXY - {arkanite}
      -u2
            -KXY - {arkanite}
      -u2z
            -KXY - {arkanite}
      -u2zs
            -KXY - {arkanite}
      -u2zss
            -KXY - {arkanite}
      -u2zsst
            -KXY - {arkanite}
```

Odczyt - Geohashe

1. Mamy ekran mobilki



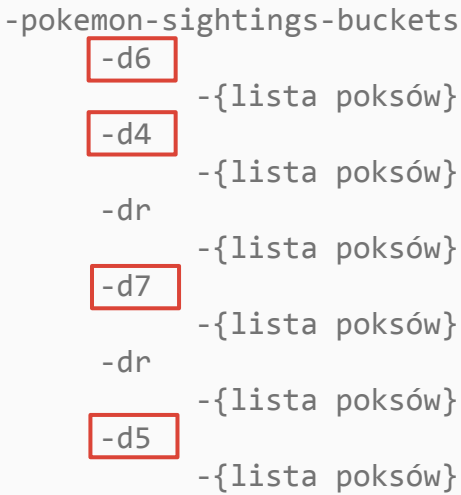
Mamy rogi ekranu!

Możemy rozpoznać jakie geohashe mieszczą się na ekranie

Mam na to nawet algorytm!
https://gist.github.com/ndrwtrsk/401e2e28d5275e3e525038cc9d6fed7f

Odczyt - Geohashe

2. Mając d4, d5, d6, d7 robimy zapis na zmiany w nich.



#magic

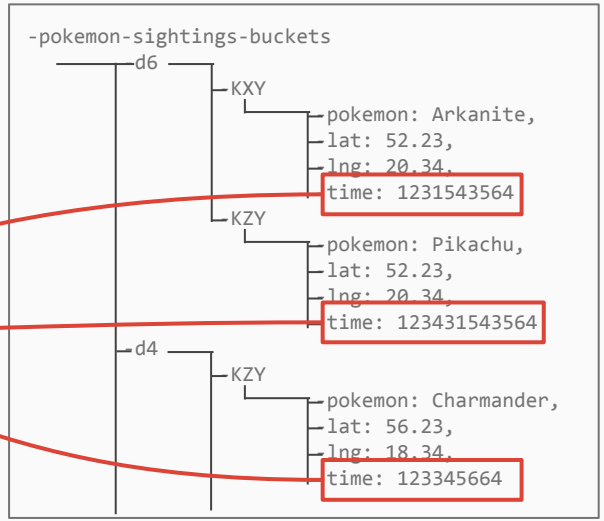
Niejawnie filtrujemy po 4 polach...

```
lat >= southwest.lat &&
lng >= southwest.lng &&
lat <= northeast.lat &&
lng <= northeast.lng
```

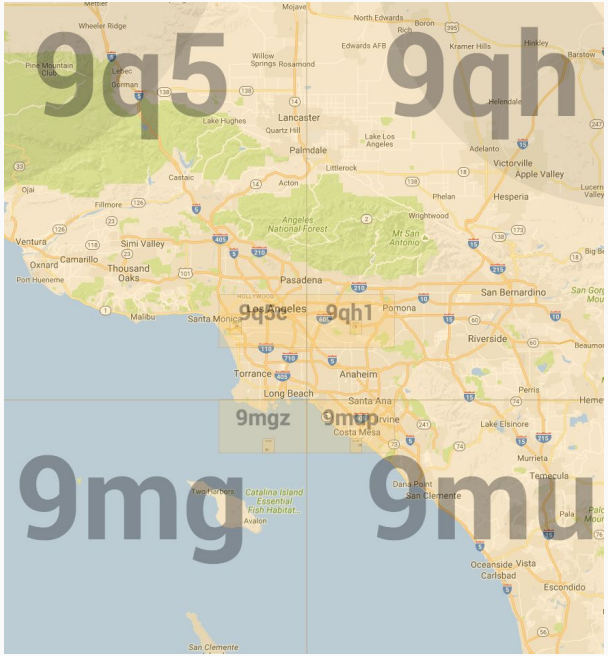
Jawnie możemy filtrować po czym tylko chcemy!

```
Czas, nazwa, whatever
```

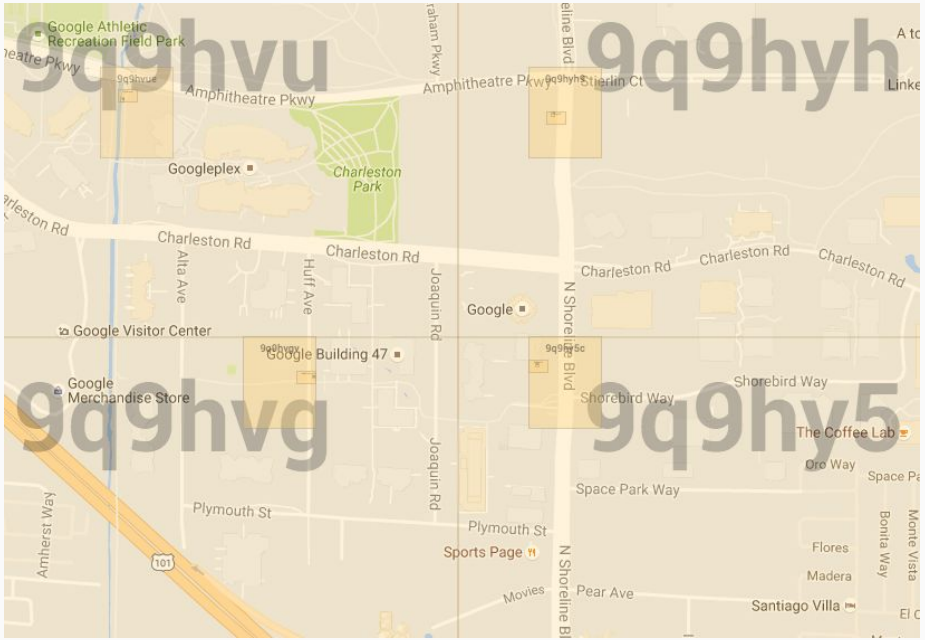
Geohashe od razu mają w sobie zawarty przedział lat i lng



Odczyt - dlaczego tyle geohashy?



Jak mały zoom, to krótsze hashe



Jak większy zoom, to dłuższe hashe

Zawsze robimy zapis na mniejwiecej tyle samo hashy

+ i -

+ Real Time!
+ Dostajemy tylko te dane, które chcemy dostać
+ Jawnie filtrujemy po czym chcemy

- Baaaaardzo dużo powielonych danych.
- Duuuuży rozmiar obiektu z geohashami
  32^6 - 1 073 741 824 obiektów

Dzięki wielkie!

Pytania?
Teraz jest dobry czas!

andrew.torski@gmail.com