

PDT - MongoDB

https://github.com/MennoCoevoorn/PDT_4.git

Marek Štrba

November 2021

Obsah

1	Navrhnete dátový model	3
1.1	Návrh	3
1.2	Zhodnotenie	4
2	Vytvorenie MongoDB databázy	4
2.1	Tweets	5
2.2	Accounts	6
3	Napíšte query pre Vašu importovanú databázu pre dva hlavné prípady použitia	7
3.1	Vypísať posledných 10 tweetov accountu so screen_name = Marndin12, spolu s údajmi o accounte	7
3.2	Vypísať prvých 10 tweetov - text, meno autora, dátum tweetu a hashtagy, ktoré retweetujú tweet s id = 1243427980199641088 . .	7

1 Navrhnete dátový model

Ako referenciu na to, aké všetky informácie majú o tweete byť zobrazené v rámci jeho zobrazenia vo feede som použil nasledovný tweet:

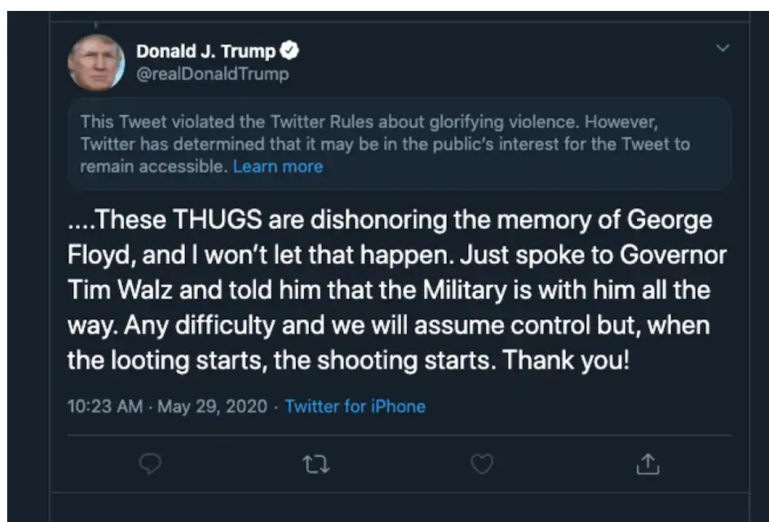


Figure 1: Referencia štruktúry tweetu vo feede

Hodnoty ako počet lajkov som nedoplňal, keďže sa tento údaj nenachádzal v pôvodných dátach.

1.1 Návrh

Povodný návrh nebral do úvahy fakt, že sa budú vyždaovať všetky informácie o účte, preto som imbedol screen name a name autora do Kolekcie tweets a celý návrh sa opieral len o jednu kolekciu. Takto by som zabránil joinom, ktoré sú v mongu nežiadúce. Screen name a name by sa síce museli pri každej zmene updatnut v každom tweete ale z povahy týchto atribútov to nevnímam ako problém. Problémom je aj duplicita údajov ale za cenu vyhnutia sa joinu to je podľa mňa prijateľné.

```
// Prvý návrh
Tweets
{
  "id": "1564980919",
  "content": "lorem ipsum dolor sit amet, consectetur adipiscing elit. #latin #deadlanguage #history",
  "screenName": "yaBoiCicero",
  "name": "Marcus Tullius Cicero",
  "happened_at": "0002-01-24 20:59:19+01",
  "retweet_count": 251,
  "parent_id": "899591098",
  "hashtags": ["latin", "deadlanguage", "history"],
}
```

Figure 2: Pôvodný návrh DB štruktúry

Pri druhom návrhu, pri ktorom som zohľadnil požiadavku zobrazenia všetkých account information, som využil 2 dokumenty. Accounts a Tweets. Accounts obsahoval v sebe pole referencií na Tweets a osobné údaje o účte. Tweets obsahoval údaje o štruktúre a obsahu tweetov.

```
// Druhý návrh
Tweets
{
  "id": "1564980919",
  "content": "lorem ipsum dolor sit amet, consectetur adipiscing elit. #latin #deadlanguage #history",
  "screenName": "yaBoiCicero",
  "name": "Marcus Tullius Cicero",
  "account_id": "2598491951",
  "happened_at": "0002-01-24 20:59:19+01",
  "retweet_count": 251,
  "parent_id": "899591098",
  "hashtags": ["latin", "deadlanguage", "history"],
}

Accounts
{
  "id": "2598491951",
  "screenName": "yaBoiCicero",
  "name": "Marcus Tullius Cicero",
  "description": "Big brain philosopher",
  "followers_count": 98746,
  "friends_count": 505,
  "status_count": 237
}
```

Figure 3: Návrh DB štruktúry

Výhodou tohoto návrhu je, že sa síce duplikujú údaje o používateľoch, ale duplikujú sa iba najmenej meniace sa údaje (screen name a name), ktoré sa dá predpokladať, že budú aj najčastejšie dopytované. Ostatné údaje sú vyčlenené do samostatnej collection, spolu s duplicitou screen name a name a dajú sa jednoducho upraviť na jednom mieste.

1.2 Zhodnotenie

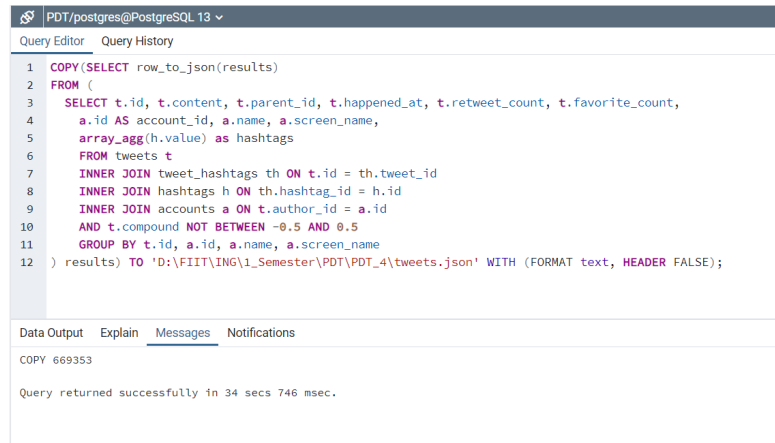
Vybrať návrh nebolo problematické, keďže sa zadanie dalo naplniť iba s agregáciou s údajmi o účte. Embeddovať všetky tieto údaje mi prišlo ako nevýhodné kvôli problému častých updatov hodnôt ako sú napríklad follower count. Zároveň by išlo o extrémnu duplicitu údajov.

2 Vytvorenie MongoDB databázy

Pre potreby MongoDB som si stiahol MongoDB, MongoDB Compass a MongoDB Database tools kvôli mongoimport, ktorý som použil pri importe dát. Musel som niektoré vstupy buď upraviť, alebo vymazať, z dôvodu že ich nevedelo Mongo kvôli problému s formátom importovať. Išlo však radovo o 10tky vstupov, takže to výsledok nemôže výrazne ovplyvniť. Nevedel som tento problém

inak vyriešiť. Skúšal som aj manuálne prekonvertovať JSON z PGSQL na UTF-8, ale to problém tiež nevyriešilo.

2.1 Tweets



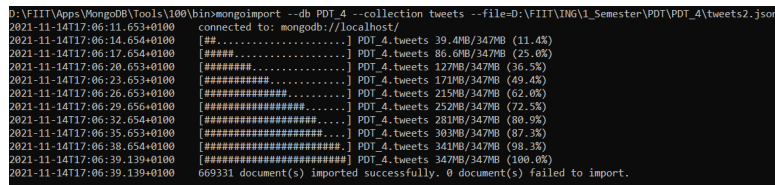
```
1 COPY(SELECT row_to_json(results)
2 FROM (
3 SELECT t.id, t.content, t.parent_id, t.happened_at, t.retweet_count, t.favorite_count,
4 a.id AS account_id, a.name, a.screen_name,
5 array_agg(h.value) as hashtags
6 FROM tweets t
7 INNER JOIN tweet_hashtags th ON t.id = th.tweet_id
8 INNER JOIN hashtags h ON th.hashtag_id = h.id
9 INNER JOIN accounts a ON t.author_id = a.id
10 AND t.compound NOT BETWEEN -0.5 AND 0.5
11 GROUP BY t.id, a.id, a.name, a.screen_name
12 ) results) TO 'D:\FIIT\ING\1_Semester\ PDT\ PDT_4\tweets.json' WITH (FORMAT text, HEADER FALSE);
```

Data Output Explain Messages Notifications

COPY 669353

Query returned successfully in 34 secs 746 msec.

Figure 4: Export extrémnych tweetov z DB



```
D:\FIIT\Apps\MongoDB\Tools\100\bin>mongoimport --db PDT_4 --collection tweets --file=D:\FIIT\ING\1_Semester\ PDT\ PDT_4\tweets2.json
connected to: mongod://localhost/
2021-11-14T17:06:11.653+0100 [## .....] PDT_4.tweets 39.4MB/347MB (11.4%)
2021-11-14T17:06:17.654+0100 [#####] PDT_4.tweets 86.4MB/347MB (25.0%)
2021-11-14T17:06:20.653+0100 [#####] PDT_4.tweets 127MB/347MB (36.5%)
2021-11-14T17:06:23.653+0100 [#####] PDT_4.tweets 171MB/347MB (49.4%)
2021-11-14T17:06:26.653+0100 [#####] PDT_4.tweets 215MB/347MB (62.0%)
2021-11-14T17:06:29.656+0100 [#####] PDT_4.tweets 252MB/347MB (72.5%)
2021-11-14T17:06:32.654+0100 [#####] PDT_4.tweets 281MB/347MB (80.9%)
2021-11-14T17:06:35.653+0100 [#####] PDT_4.tweets 303MB/347MB (87.3%)
2021-11-14T17:06:38.654+0100 [#####] PDT_4.tweets 341MB/347MB (98.3%)
2021-11-14T17:06:39.139+0100 [#####] PDT_4.tweets 347MB/347MB (100.0%)
2021-11-14T17:06:39.139+0100 669331 document(s) imported successfully. 0 document(s) failed to import.
```

Figure 5: Import tweetov do Monga

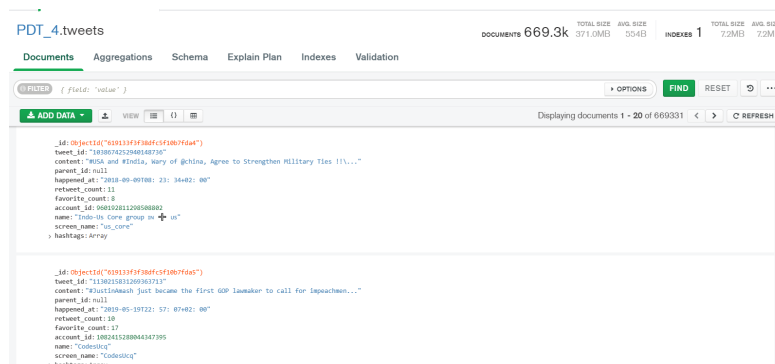


Figure 6: Tweets collection v Compass

2.2 Accounts



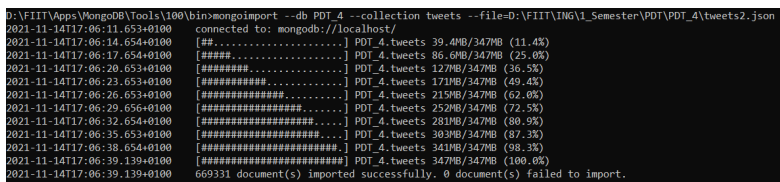
```
1 COPY(SELECT row_to_json(results)
2 FROM (
3   SELECT DISTINCT(a.id), a.screen_name, a.name, a.description, a.followers_count, a.friends_count,
4     a.statuses_count
5   FROM accounts a INNER JOIN tweets t
6   ON a.id = t.author_id
7   AND t.compound NOT BETWEEN -0.5 AND 0.5
8 ) results) TO 'D:\FIIT\ING\1_Semester\PDT\PDT_4\accounts.json' WITH (FORMAT text, HEADER FALSE);
```

Data Output Explain Messages Notifications

COPY 132993

Query returned successfully in 7 secs 785 msec.

Figure 7: Export accountov z PGSQL. Kvôli vyššie spomenutým problémom s formátovaním som sa rozhodol importovať iba accounts, ktoré sú naviazané na extrémne tweety



```
D:\FIIT\Apps\MongoDB\Tools\100\bin>mongoimport --db PDT_4 --collection tweets --file=D:\FIIT\ING\1_Semester\PDT\PDT_4\tweets2.json
connected to: mongodb://localhost/
2021-11-14T17:06:11.653+0100 [##.....] PDT_4.tweets 39.4MB/347MB (11.4%)
2021-11-14T17:06:17.654+0100 [#####] PDT_4.tweets 86.6MB/347MB (25.0%)
2021-11-14T17:06:20.653+0100 [#####] PDT_4.tweets 127MB/347MB (36.5%)
2021-11-14T17:06:23.653+0100 [#####] PDT_4.tweets 171MB/347MB (49.4%)
2021-11-14T17:06:26.653+0100 [#####] PDT_4.tweets 215MB/347MB (62.0%)
2021-11-14T17:06:29.656+0100 [#####] PDT_4.tweets 252MB/347MB (72.5%)
2021-11-14T17:06:32.654+0100 [#####] PDT_4.tweets 281MB/347MB (80.9%)
2021-11-14T17:06:35.653+0100 [#####] PDT_4.tweets 303MB/347MB (87.3%)
2021-11-14T17:06:38.654+0100 [#####] PDT_4.tweets 341MB/347MB (98.3%)
2021-11-14T17:06:39.139+0100 [#####] PDT_4.tweets 347MB/347MB (100.0%)
2021-11-14T17:06:39.139+0100 669331 document(s) imported successfully. 0 document(s) failed to import.
```

Figure 8: Import accounts do Monga

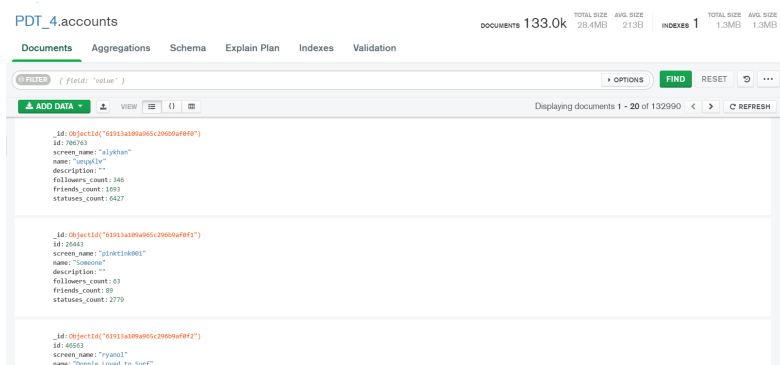


Figure 9: Accounts collection v Compass

3 Napíšte query pre Vašu importovanú databázu pre dva hlavné prípady použitia

3.1 Vypísať posledných 10 tweetov accountu so screen_name = Marndin12, spolu s údajmi o accounte

Pre tento dopyt som potreboval údaje z oboch collections, preto som použil metódu aggregate, v ktorej vyfiltroval tweety obsahujúce screen_name "Marndin12", zoradil ich podľa _id od DESC, limitom vybral iba 10 a pomocou lookup metódy som získal dáta z kolekcie accounts.

```
> db.tweets.aggregate([
  { $match : { screen_name : "Marndin12" } },
  { $sort : { _id : -1 } },
  { $limit : 10 },
  {
    $lookup:
    {
      from: "accounts",
      localField: "account_id",
      foreignField: "id",
      as: "account_info"
    }
  }
])
```

Figure 10: Celková query v MongoDB Compass

3.2 Vypísať prvých 10 tweetov - text, meno autora, dátum tweetu a hashtagy, ktoré retweetujú tweet s id = 1243427980199641088

Pri tomto dopyte som použil metódu find, keďže vďaka duplicite údajov som mal všetky dáta uložené v kolekcii tweets. Keďže bolo potrebné vybrať prvých 10 záznamov tak na to postačila jednoduchá metóda .limit. Definícia atribútov z kolekcie, ktoré sa majú vybrať sa dala spraviť cez objekt a podobne aj podmienka s parent_id.

```
db.tweets.find({"parent_id": "1243427980199641088"}, {_id:0, name:1, screen_name:1, content:1, hashtags:1}).limit(10)
```

Figure 11: Celková query v MongoDB Compass

Táto query nevráti však žiadne výsledky. To je ale v poriadku, keďže ani

v pôvodnom datasete nebol žiadny extrémny tweet, ktorý by bol retweetom daného tweetu.

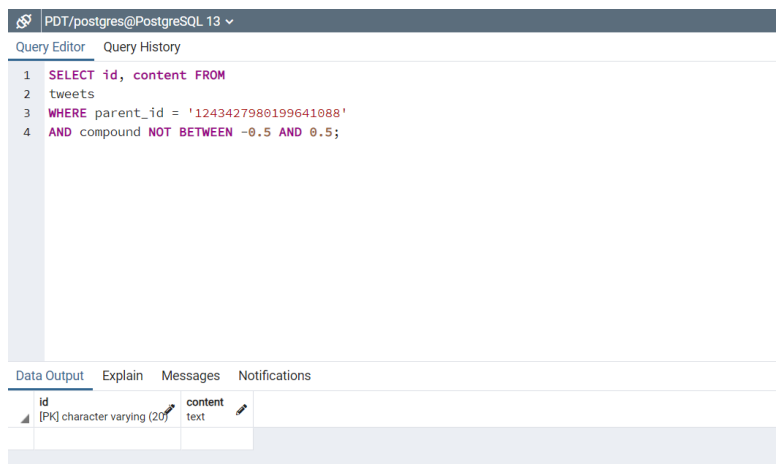


Figure 12: Kontrola v pôvodnom datasete