

Analiza prędkości wykonywania zapytań w wybranych bazach nie SQL-owych

Wojciech Bolesta*

Politechnika Lubelska, Instytut Informatyki, Nadbystrzycka 36B, 20-618 Lublin, Polska

Streszczenie. Artykuł naukowy dotyczy porównania prędkości zapytań dwóch baz danych NoSQL. Opisywanymi bazami danych będą MongoDB oraz CouchDB. W pracy przedstawione zostaną porównania prędkości takich zapytań jak dodawanie danych do bazy, edycja danych bazy danych, usuwanie danych z bazy, a także wyszukiwanie danych w bazie danych. Przedstawione zostanie również ogólne porównanie baz, z odpowiedzią na pytanie, która z badanych baz NoSQL jest szybsza.

Słowa kluczowe: NoSQL; MongoDB; CouchDB

*Autor do korespondencji.

Adres e-mail: wojtek.bolesta@wp.pl

Analysis of query execution speed in the selected NoSQL databases

Wojciech Bolesta*

Institute of Computer Science, Lublin University of Technology, Nadbystrzycka 36B, 20-618 Lublin, Poland

Abstract. The scientific article deals with a comparison of the query speed of two NoSQL databases. Described databases will be MongoDB and CouchDB. The work presents speed comparisons of such queries as adding data to the database, editing database data, deleting data from the database, and searching data in the database. Also a general comparison of bases will be presented, with the answer to the question of which of the tested NoSQL databases is faster.

Keywords: NoSQL; MongoDB; CouchDB

*Corresponding author.

E-mail address: wojtek.bolesta@wp.pl

1. Wstęp

W poniższym artykule naukowym przedstawione zostaną wyniki badań dotyczących porównania prędkości wykonywania zapytań uzyskanych z dwóch nierelacyjnych bazach MongoDB oraz CouchDB. Do badania baz danych użyta została stworzona na potrzeby badań baza danych, do której następnie dodane zostały w procesie dodatkowe pozycje w ustalonym schemacie tak w MongoDB jak i CouchDB. Następnie w celu porównania, użyte zostały logi obu programów, dzięki którym uzyskane zostały wyniki prędkości wykonania operacji dodania do baz tych samych informacji za pomocą języka NoSQL[8]. W trakcie wykonywania badań wykonane zostały takie zapytania jak dodawanie do bazy danych, usuwanie danych z bazy, edycja wstawionych wcześniej danych, wyszukanie kilku zestawów danych, a także porównanie prędkości pobierania danych z różnych baz.[10,6,5]

Informacje otrzymane z badań umożliwiają określenie bazy danych, której czas wykonywania zapytań jest krótszy. Różnice w pojedynczych przypadkach, jak można założyć, mogą się okazać nieistotne dla użytkowników indywidualnych, lecz w przypadku dużych firm, które przeprowadzają operacje na dużej ilości danych, takie różnice mogą jednak okazywać się o wiele większe, przez co wybieranie określonego środowiska jest bardzo istotnym krokiem podczas prowadzenia pracy na danych tekstowych. Bazy NoSQL posiadają o wiele mniejszy możliwy zakres użytkowania, ale w porównaniu do baz SQL

można się spodziewać iż w tych wąskich dziedzinach, w których operują, bazy NoSQL okazują niebywałą wydajność[1,10,2]. Wszystkie badania, tak w MongoDB jak i CouchDB przeprowadzone zostały na systemie operacyjnym Windows 7 Home Premium, na tym samym stanowisku testowym.

Cel

Celem poniższego artykułu naukowego jest przedstawienie uzyskanych wyników z analizy prędkości wykonywanych zapytań w wybranych bazach NoSQL, MongoDB oraz CouchDB. W oparciu o posiadane wyniki pomiaru prędkości zapytań o tym samym działaniu w wybranych bazach, dokonana zostanie następnie analiza porównawcza pomiędzy badanymi bazami MongoDB oraz CouchDB.

Teza

Za główne założenie tezy, przygotowanej na potrzeby artykułu naukowego przyjęto fakt iż MongoDB jest bazą wydajniejszą względem czasu od drugiej badanej bazy NoSQL CouchDB.

2. MongoDB

MongoDB jest jednym z najpopularniejszych nierelacyjnych systemów zarządzania bazami danych, napisanym w języku C++. MongoDB cechuje się dużą skalowalnością, wydajnością oraz nie posiada zdefiniowanej struktury obsługi danych. W MongoDB dane przechowywane

są jako dokument JSON, przez co tworzonym aplikacjom umożliwiające jest naturalniejsze przetwarzanie danych, przy zachowaniu możliwości tworzenia hierarchii oraz indeksowania.[1,2,3,5]

Aktualnie MongoDB posiada możliwość przeprowadzania operacji na dokumentach, a dzięki korzystaniu z MongoDB Atlas, można również obserwować wydajność bazy danych, a sam program w sposób automatyczny zabezpiecza bazę danych, na której użytkownik przeprowadza prace, a także w porównaniu do desktopowej wersji MongoDB, posiada możliwość odzyskiwania danych dzięki możliwości tworzenia backupów.[8,4]

3. CouchDB

CouchDB jest nowym rodzajem baz danych. Podobnie jak MongoDB nie jest to baza relacyjna, ani obiektowa. Przechowuje ona dokumenty. W CouchDB dokument jest czymś w rodzaju występującej w Java kolekcji map, czyli posiada klucze i wartości[7]. Kluczem w CouchDB zawsze jest String, który jest unikalny w ramach pojedynczego dokumentu. Ciekawą i wartą wspomnienia cechą CouchDB jest fakt, iż dokumenty te są dostępne przez RESTowy interfejs oraz protokół http, zapisane w JSONie, znanym z JavaScripta formacie zapisu złożonych danych, który w przypadku badań przeprowadzanych na potrzeby artykułu naukowego, jest formatem zapisu bazy danych, na której przeprowadzane są badania. Językiem jaki został użyty do napisania CouchDB jest język Erlang. Autor wybrał ten język z powodu jego kompatybilności z pisaniem aplikacji wielowątkowych, co w efekcie sprawiło iż CouchDB jest dobrze skalowalny na wiele procesów/rdzeni.[8,10,9,6]

4. Przykłady

Poniższy Przykład obrazuje schemat jednego przygotowanego zestawu danych z bazy użytej na potrzeby wykonania badań.

Przykład 1. Przykładowy zestaw danych

```
{
  "fields": {
    "edited": "2014-12-20T21:17:56.891Z",
    "name": "Luke Skywalker",
    "created": "2014-12-09T13:50:51.644Z",
    "gender": "male",
    "skin_color": "fair",
    "hair_color": "blond",
    "height": "172",
    "eye_color": "blue",
    "mass": "77",
    "homeworld": 1,
    "birth_year": "19BBY"
  },
  "model": "resources.people",
  "pk": 1
}
```

Przykład 2. Wprowadzenie danych – MongoDB

```
db.collection.insertOne(
  {
    "fields": {
      "edited": "2014-12-20T21:17:56.891Z",
```

```
"name": "Luke Skywalker",
"created": "2014-12-09T13:50:51.644Z",
"gender": "male",
"skin_color": "fair",
"hair_color": "blond",
"height": "172",
"eye_color": "blue",
"mass": "77",
"homeworld": 1,
"birth_year": "19BBY"
},
"model": "resources.people",
"pk": 1
},
}
```

Przykład 3. Przykład 3. Wprowadzenie danych – CouchDB

```
{
  "fields": {
    "_id": "2014-12-20T21:17:56.891Z",
    "name": "Luke Skywalker",
    "created": "2014-12-09T13:50:51.644Z",
    "gender": "male",
    "skin_color": "fair",
    "hair_color": "blond",
    "height": "172",
    "eye_color": "blue",
    "mass": "77",
    "homeworld": 1,
    "birth_year": "19BBY"
  },
  "model": "resources.people",
  "pk": 1
},
}
```

5. Analiza porównawcza

Poniższy rozdział zawiera wyniki badań uzyskanych podczas przeprowadzania badań magisterskich, dla obu badanych baz danych, MongoDB oraz CouchDB. Wszystkie wyniki badań przedstawione w poniższym rozdziale reprezentując wyniki badań dla określonej liczby powtórzeń danej operacji. Poza tabelami 1 oraz 2, wszystkie tabele zawierają operacje dotyczące tysiąca zestawów danych. Tabela 1 przedstawia wynik czasu uzyskanego dla przeprowadzenia operacji dziesięciokrotnie, a tabela 2 reprezentuje wprowadzanie kolejno dla jednego, dziesięciu, stu i tysiąca zestawów danych. Tabela 1 przedstawiona została uwzględniając wymienione ilości wprowadzeń zestawów, gdyż stanowiła przykład największych różnic uzyskanych podczas wprowadzania testowych zestawów danych widocznych w Przykładzie 1 do obu testowanych baz danych. Operacje usuwania, wyszukiwania, oraz aktualizacji danych najlepiej obrazują różnice gdy wykonane zostały na tysiącu testowych zestawów danych.

Tabela 1. Średni czas wykonanie z użyciem klauzuli WHERE

Operacja	CouchDB (ms)	MongoDB (ms)
WHERE field = ?	0,032	0,016
WHERE field zLIKE 'x%'	0,072	0,040

Wszystkie wyniki widoczne w powyższej tabeli 1 przedstawiają średni wynik wykonania klauzuli dziesięć razy, a wartości przedstawione są w milisekundach. Zauważyć można iż w obu badanych przypadkach czas wykonania MongoDB jest znacznie krótszy.

Tabela 2. Operacja INSERT

Operacja	CouchDB (ms)	MongoDB (ms)
INSERT 1 ROW	1,201	0,187
INSERT 10 ROW	3,057	2,068
INSERT 100 ROW	13,246	8,842
INSERT 1000 ROW	402,352	106,492

Podobnie jak w przypadku klauzuli, podczas operacji wprowadzania danych można zauważyć iż MongoDB jest wydajniejszy, niezależnie od tego, jak wiele zestawów danych zostaje dodanych do bazy danych w pliku JSON. Na podstawie powyższych wyników można zaobserwować iż im więcej informacji wprowadzamy, tym większa jest różnica zauważalna pomiędzy CouchDB, a MongoDB.

Tabela 3. Operacja DELETE

Operacja	CouchDB (ms)	MongoDB (ms)
DELETE	0,822	0,624

Tabela 3 przedstawia czas w jakim badane bazy wykonują operacje usunięcia wprowadzonych zestawów danych. Wynik ten dotyczy ostatniego dodania z tabeli 2 dotyczącego tysiąca zestawów danych. Podobnie jak w powyższych dwóch przypadkach czas wykonywania operacji w MongoDB jest krótszy od uzyskanego w CouchDB.

Tabela 4. Operacja UPDATE

Operacja	CouchDB (ms)	MongoDB (ms)
UPDATE	1,302	1,331

Podobnie jak w tabeli 3 i w tabeli 4 można zauważyć czas wykonania dla tysiąca testowych zestawów danych. W przeciwieństwie do wszystkich poprzednich wyników, tym razem operacja, która zajęła mniej czasu odbyła się w CouchDB. Podczas badania obu baz, jedynie w uaktualnianiu informacji bazy, CouchDB wykazał się lepszym czasem wykonania od MongoDB.

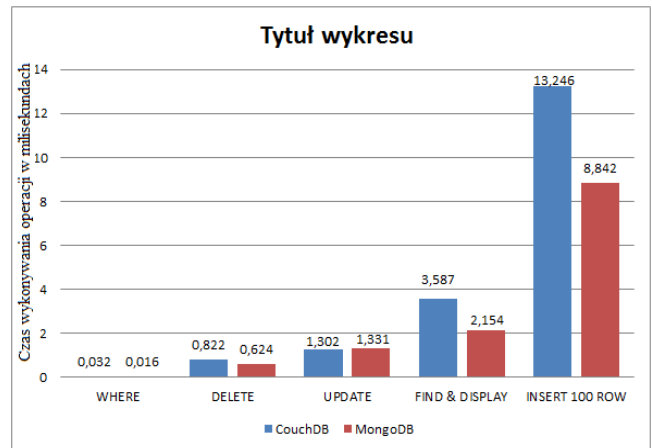
Tabela 5. Operacja FIND & DISPLAY

Operacja	CouchDB (ms)	MongoDB (ms)
FIND & DISPLAY	3,587	2,154

W obu systemach baz danych operacje wyszukania oraz wyświetlenia wykonywane dla tysiąca zestawów danych wprowadzanych do bazy danych, wykonują się w prędkości uznawanej za zadowalającą jak na tak dużą ilość danych, ale także jak w przypadku większości badanych zapytań, tak i to pokazuje iż MongoDB jest wydajniejszy od CouchDB. Rysunek 1 przedstawia obrazowe porównanie umieszczonych w artykule badań. Podczas tego kroku widoczne jest, iż MongoDB poza aktualizacją danych jest wydajniejszy względem czasu, niż druga badana baza NoSQL CouchDB.

6. Wnioski

Podstawowym wnioskiem uzyskanym podczas badań poświęconych porównaniu prędkości zapytań pomiędzy bazami NoSQL CouchDB oraz MongoDB jest potwierdzenie tezy, iż baza MongoDB jest bazą wydajniejszą względem czasu od bazy CouchDB.



Rys.1. Porównanie uzyskanych czasów dla wykonania zapytania.

Poza jednym przypadkiem, aktualizacji danych, baza CouchDB jest bazą zdecydowanie mniej wydajną od bazy MongoDB, która w przypadkach, operacji dodawania, usuwania, wyszukiwania danych oraz prędkości działania pętli jest bazą zdecydowanie wydajniejszą od drugiej badanej bazy.

Największe różnice zaobserwować można w przypadku operacji dodawania tysiąca rekordów do bazy. Wynik uzyskany przez CouchDB jest prawie czterokrotnie wyższy od wyniku uzyskanego przez MongoDB.

Najniższą różnicę natomiast można zaobserwować w przypadku badania operacji aktualizacji danych, w której to MongoDB wypadło nieznacznie gorzej od CouchDB, a różnica pomiędzy badanymi bazami wynosiła zaledwie 0,029 ms

Literatura

- [1] R. Henricsson, Document Oriented NoSQL Databases: A comparison of performance in MongoDB and CouchDB using a Python interface, PublicationsElectronic Research Archive Blekinge Techniska Hogskola; 2011.
- [2] P.Noiumkar, T. Chomsiri. A Comparison the Level of Security on Top 5 Open Source NoSQL Databases. Annual International Conference on Information Technology & Applications, 2014.
- [3] N. Aloisio Dourado, E. Fernandes, M. Holanda, E. Riberio, RCarvalho MongoDB performance analysis: A comparative study between stand-alone and sharded cluster deployments with open data from Brazilian Bolsa familia program, 12th Iberian Conference on Information Systems and Technologies, 2017.
- [4] M. Patil, A. Hanni, C. Tejeshwar, P. Patil, A qualitative analysis of the performance of MongoDB vs MySQL database based on insertion and retrieval operations using a web/android application to explore load balancing — Sharding in MongoDB and its advantages, International Conference on I-SMAC, 2017.
- [5] M. Jung, S. Youn, J. Bae, Y. Choi, A Study on Data Input and Output Performance Comparison of MongoDB and PostgreSQL in the Big Data Environment, 8th International Conference on Database Theory and Application, 2015.
- [6] M. Brown, Getting Started with CouchDB, Sebastopol, 2012.
- [7] S. Gupta, A comparative study of elasticsearch and CouchDB document oriented databases, International Conference on Inventive Computation Technologies, 2016.

- [8] K. Kumar, S. Mohanavalli, A performance comparison of document oriented NoSQL databases, International Conference on Computer, Communication and Signal Processing, 2017.
- [9] N. Surya, E. Ramez, Quantitative Analysis of Scalable NoSQL Databases, San Francisco. 2016.
- [10] P. Pragati, G.Saumya, K. Anil, Analysis of Various NoSql Database, Rajasthan, 2015.