

Kraków, 25.05.2023r.



Porównanie wydajności złączy i zagnieżdżeń dla schematów znormalizowanych i zdenormalizowanych

Rafał Żegleń

Numer albumu: 411632

1. Cel pracy

Celem pracy jest analiza i porównanie zapytań bazujących na złączeniach i zagnieżdżeniach dla tabeli stratygraficznej. Badanie przeprowadzono dla MySQL i PostgreSQL.

2. Konfiguracja sprzętowa i programowa

Wszystkie testy omówione w niniejszym artykule wykonano na komputerze o następujących parametrach:

CPU: Intel(R) Core(TM) i3-7100U CPU @ 2.40GHz 2.40 GHz

RAM: 8,00 GB,

SSD: SSDPR-CX400-256-G2, 256 GB

S.O.: Windows 2011 Pro.

Jako systemy zarządzania bazami danych wybrano oprogramowanie wolno dostępne:

MySQL, wersja Community Server 8.0.33.0,

PostgreSQL, wersja 15.3-1.

IDE:

MySQL Workbench 8.0 CE

pgAdmin4

3. Przebieg analizy

W celu przeprowadzenia testów utworzono tabele zawierające nazwy eonów, er, epok, okresów i pięter zgodnie z tabelą stratygraficzną. Formę zdenormalizowaną tabeli geochronologicznej osiągnięto tworząc jedną tabelę za pomocą złączenia naturalnego danych z wszystkich tabel.

4. Testy wydajności

Testy przeprowadzono poprzez wykonanie 4 zapytań (Rys.1.), które miały na celu sprawdzić wydajność złączeń oraz zagnieżdżeń z utworzoną wcześniej tabelą stratygraficzną zarówno w wersji znormalizowanej jak i zdenormalizowanej. Najpierw zapytania wykonano bez nałożenia indeksów na kolumny danych, a następnie nałożono indeksy na wszystkie kolumny i powtórzono całą procedurę.

```
-- Zapytanie 1 złączenie syntetycznej tablicy miliona wyników z tabelą geochronologiczną w postaci zdenormalizowanej
SELECT COUNT(*) FROM Milion INNER JOIN GeoTabela ON (mod(Milion.liczba,68)=(GeoTabela.id_pietro));

-- Zapytanie 2 złączenie syntetycznej tablicy miliona wyników z tabelą geochronologiczną w postaci znormalizowanej, reprezentowaną przez złączenia pięciu tabel
SELECT COUNT(*) FROM Milion INNER JOIN GeoPietro ON (mod(Milion.liczba,68)=GeoPietro.id_pietro) NATURAL JOIN GeoEpoka
NATURAL JOIN GeoOkres NATURAL JOIN GeoEra NATURAL JOIN GeoEon;

-- Zapytanie 3 złączenie syntetycznej tablicy miliona wyników z tabelą geochronologiczną w postaci zdenormalizowanej,
-- przy czym złączenie jest wykonywane poprzez zagnieżdżenie skorelowane

SELECT COUNT(*) FROM Milion WHERE mod(Milion.liczba,68)= (SELECT id_pietro FROM GeoTabela WHERE mod(Milion.liczba,68)=(id_pietro));

-- Zapytanie 4

SELECT COUNT(*) FROM milion WHERE mod(milion.liczba,68)
IN (SELECT geopietro.id_pietro FROM geopietro
NATURAL JOIN geoepoka NATURAL JOIN
geookres NATURAL JOIN geoera NATURAL JOIN geoeon);
```

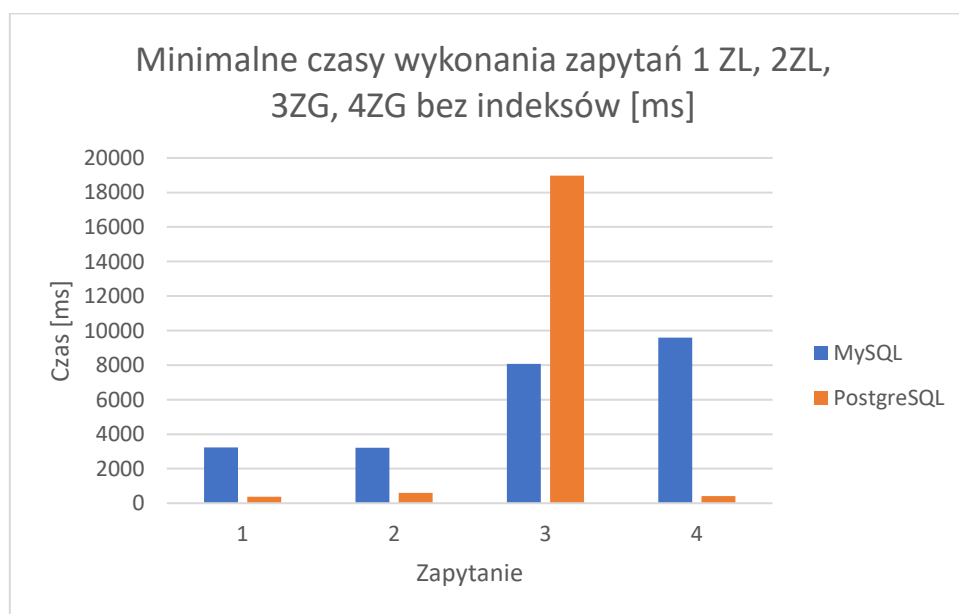
Rysunek 1 Fragment skryptu w MySQL zawierający zapytania wykorzystywane do testów

5. Wyniki testów

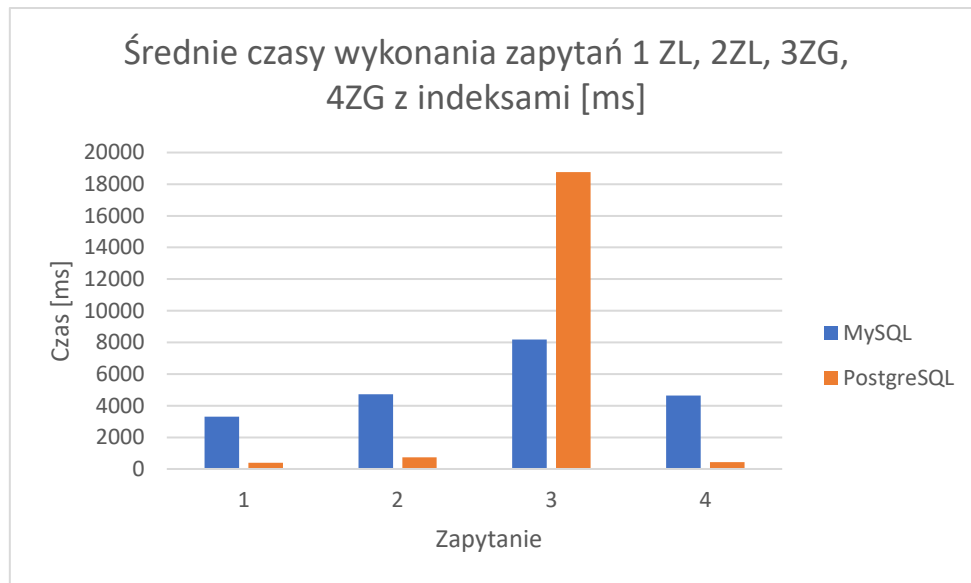
Każdy z testów został przeprowadzony wielokrotnie, wartości skrajne zostały pominięte. Wyniki przedstawiono w Tabeli 1. W celu porównania zostały utworzone wykresy (Wykres 1., Wykres 2.), które graficznie przedstawiają dane z tabeli.

Tabela 1.

Czasy wykonania zapytań 1 ZL, 2 ZL, 3 ZG i 4 ZG [ms]								
	1 ZL		2 ZL		3 ZL		4 ZL	
Bez indeksów	MIN	ŚR	MIN	ŚR	MIN	ŚR	MIN	ŚR
MySQL	3228,1657	3476,7195	3213,2525	3301,4049	8065,4155	8259,5306	9593,5075	9895,68125
PostgreSQL	383	441,75	603	650,5	18984	19531,5	411	557,75
Z indeksami								
MySQL	3302,8195	3301,48436	4724,874	4742,7926	8173,97925	8745,58658	4636,212	4814,81669
PostgreSQL	387	576,5	728	769,75	18756	19485,5	442	565,25



Wykres 1



Wykres 2

6. Wnioski

Otrzymane wyniki pozwalają wyciągnąć następujące wnioski:

- Zagnieżdżenia skorelowane są dużo wolniejsze w wykonaniu niż złączenia
- Użycie indeksów nie wpłynęło znacznie na czas wykonywania się zapytań
- W systemie PostgreSQL czasy wykonywania zapytań ZL1, ZL2, ZG4 są znacznie niższe niż w systemie MySQL. Wyjątek jednak stanowi zapytanie ZG3, którego celem jest złączenie tablicy miliona wyników z tabelą stratygraficzną w postaci zdenormalizowanej poprzez zagnieżdżenie skorelowane. Czas wykonywania tego zapytania w systemie PostgreSQL jest około dwa razy większy niż w MySQL