

*Wiktor Salwa 409775*

# **PORÓWNYWANIE WYDAJNOŚCI ZŁĄCZEŃ I ZAGNIEŹDZEŃ DLA SCHEMATÓW ZNORMALIZOWANYCH I ZDENORMALIZOWANYCH**

## **Wprowadzenie**

Celem analizy jest porównanie wydajności kwerend bazujących na złączeniach oraz zagnieżdżeniach. Kwerendy zostały wykonane na tabeli stratygraficznej. Następnie zbadano czas wykonywania poleceń bazujących na złączeniach oraz zagnieżdżeniach. Wyniki zostały przedstawione w tabeli.

## **Konfiguracja komputerowa oraz programistyczna**

Wszystkie testy, opisane w niniejszym materiale zostały wykonane na komputerze o poniższych parametrach.

- CPU: Intel(R) Core(TM) i3-9100F 3.60GHz
- HDD: TOSHIBA HDWD120 7200 rpm
- RAM: 16 GB
- OS: Windows 10 Pro

Jako system zarządzania bazami danych wybrano oprogramowanie:

- PostgreSQL 11.20
- MySQL community-8.0.33.0

Jako zintegrowane środowisko programistyczne (IDE) wybrano:

- DataGrip 2023.1.2 Build 231.9011.35

## Konstrukcja baz danych

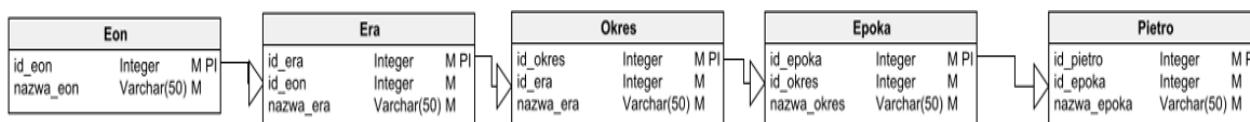
Do analizy została wykorzystana tabela stratygraficzna. Przechowuje ona jednostki geochronologiczne mające wymiar czasowy tj.: eon, era, okres, epoka oraz wiek. Tabela stratygraficzna wykorzystywana do testów została przedstawiona w załączniku nr.1 W opracowaniu skupiono się na konstrukcji tabeli stratygraficznej dla dwóch przypadków

- Schemacie znormalizowanym (rys 1.)
- Schemacie zdenormalizowanym (rys 2.)

Formę zdemoralizowaną tabeli osiągnięto przy wykorzystaniu złączenia, obejmujące wszystkie tabele tworzące hierarchię:

```
CREATE TABLE GeoTabela AS (SELECT * FROM geo.Pietro NATURAL JOIN geo.Epoka  
NATURAL JOIN geo.Okres NATURAL JOIN geo.Era NATURAL JOIN geo.Eon );
```

Tworząc tabelę zdenormalizowaną *GeoTabela* uzyskaliśmy łatwy dostęp do danych w jednym miejscu, co nie jest możliwe w przypadku schematu znormalizowanego.



Rysunek 1 Postać znormalizowana

GeoTabela	
id_pietro	Integer M PI
nazwa_pietro	Integer M
id_epoka	Integer M
nazwa_epoka	Integer M
id_okres	Integer M
nazwa_okres	Integer M
id_era	Integer M
nazwa_era	Integer M
id_eon	Integer M
nazwa_eon	Integer M

Rysunek 2 Postać zdenormalizowana

## Pytania testowe:

W testach wykorzystano cztery zapytania:

- Zapytanie 1 (1 ZL), którego celem jest złączenie syntetycznej tablicy miliona wyników z tabelą geochronologiczną w postaci zdenormalizowanej, przy czym do warunku złączenia dodano operację modulo, dopasowującą zakresy wartości złączanych kolumn:

```
SELECT COUNT(*) FROM geo.Milion INNER JOIN GeoTabela ON
```

`(mod(geo.Milion.liczba,102)=(GeoTabela.id_pietro));`

- Zapytanie 2 (2 ZL), którego celem jest złączenie syntetycznej tablicy miliona wyników z tabelą geochronologiczną w postaci znormalizowanej, reprezentowaną przez złączenia pięciu tabel:

```
SELECT COUNT(*) FROM geo.Milion INNER JOIN geo.Pietro ON
(mod(geo.Milion.liczba,102)=geo.Pietro.id_pietro) NATURAL JOIN geo.Epoka NATURAL
JOIN
geo.Okres NATURAL JOIN geo.Era NATURAL JOIN geo.Eon;
```

- Zapytanie 3 (3 ZG), którego celem jest złączenie syntetycznej tablicy miliona wyników z tabelą geochronologiczną w postaci zdenormalizowanej, przy czym złączenie jest wykonywane poprzez zagnieżdżenie skorelowane:

```
SELECT COUNT(*) FROM geo.Milion WHERE mod(Milion.liczba,102)=
(SELECT id_pietro FROM GeoTabela WHERE mod(Milion.liczba,102)=(id_pietro));
```

- Zapytanie 4 (4 ZG), którego celem jest złączenie syntetycznej tablicy miliona wyników z tabelą geochronologiczną w postaci znormalizowanej, przy czym złączenie jest wykonywane poprzez zagnieżdżenie skorelowane, a zapytanie wewnętrzne jest złączeniem tabel poszczególnych jednostek geochronologicznych:

```
SELECT COUNT(*) FROM geo.Milion WHERE Milion.liczba % 102 IN
(SELECT geo.Pietro.id_pietro FROM geo.Pietro
NATURAL JOIN geo.Epoka
NATURAL JOIN geo.Okres
NATURAL JOIN geo.Era
NATURAL JOIN geo.Eon);
```

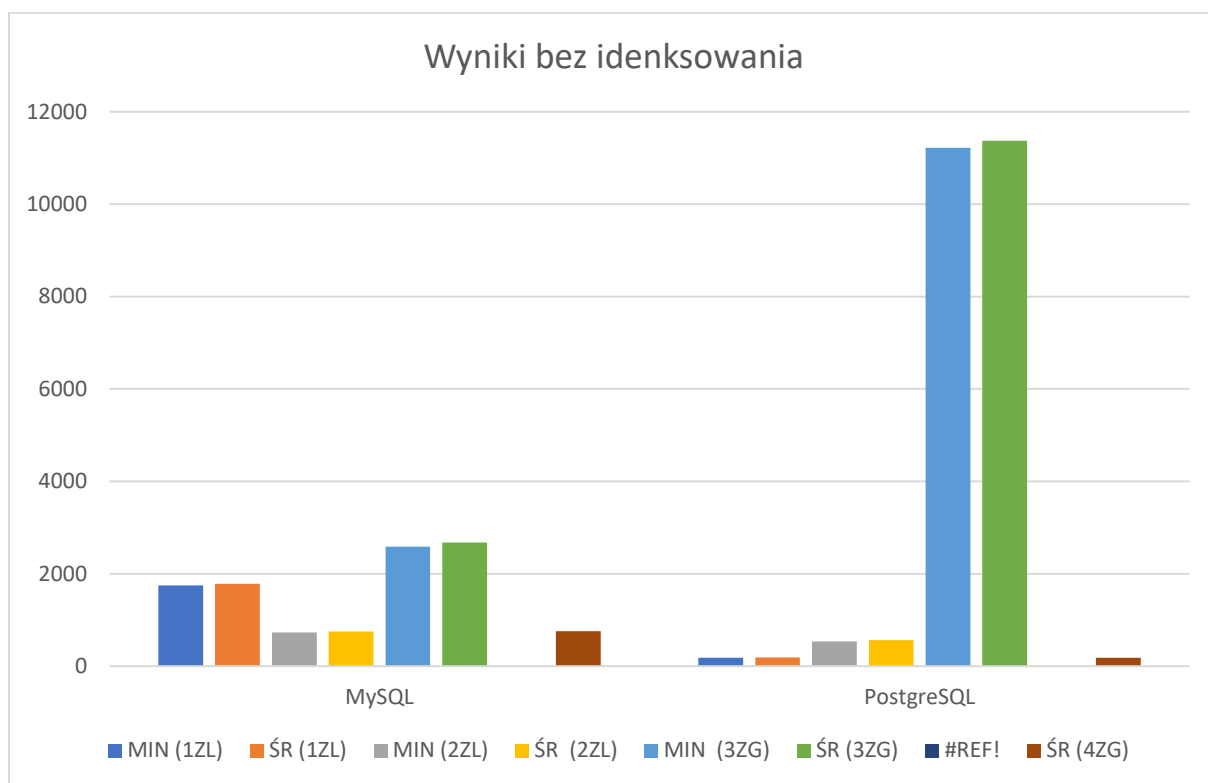
## Wyniki testów:

Każde zapytanie zostało wykonane 15 razy, w przypadku otrzymania wyniku skrajnego był on omijany. Wyniki najmniejsze oraz średnie, uzyskane podczas testów zostały przedstawione w tabeli (tab.1)

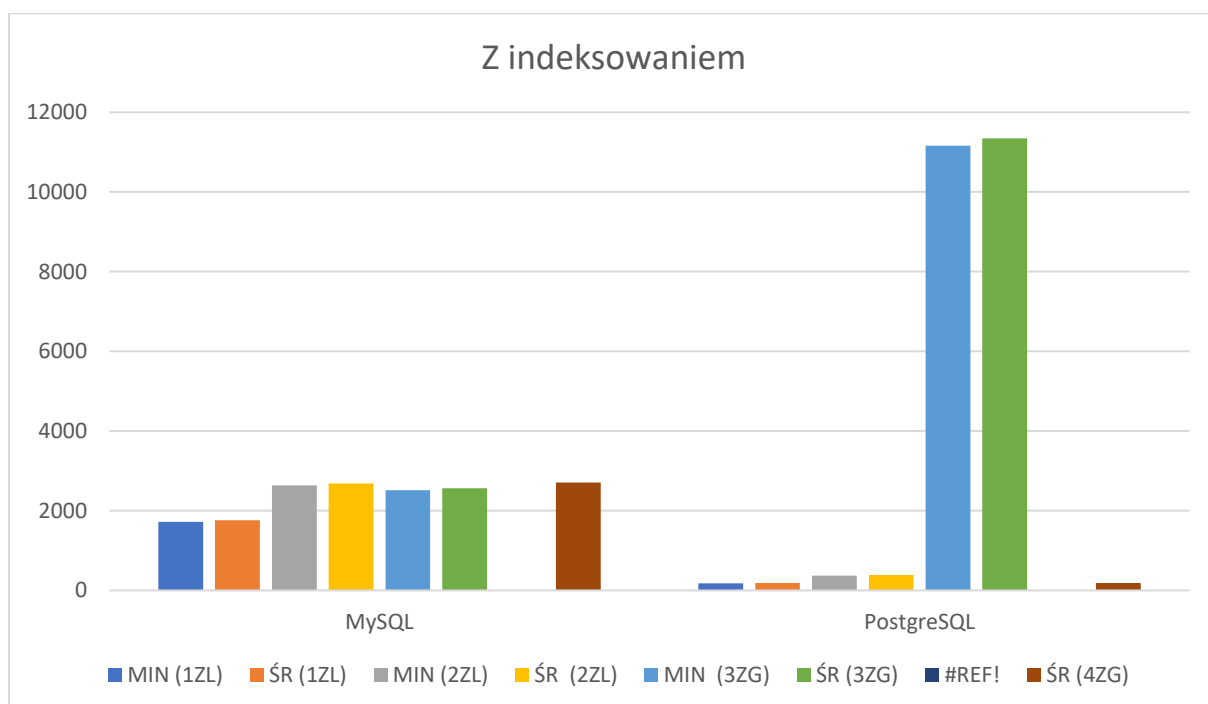
Kolumna1	1ZL		2ZL		3ZG		4ZG	
<b>BEZ INDEKSÓW</b>	MIN	ŚR	MIN	ŚR	MIN	ŚR	MIN	ŚR
MySQL	1746	1786	729	751	2590	2674	731	758
PostgreSQL	183	191	536	562	11223	11370	179	183
<b>Z INDEKSAMI</b>								
MySQL	1722	1764	2633	2688	2515	2567	2622	2710
PostgreSQL	180	185	374	384	11164	11347	177	183

Tab 1. Wyniki najniższe oraz wartości średnie uzyskane podczas testów

Dokładne wartości uzyskane podczas testów znajdują się w tabeli, znajdującej się w załączniku nr.2. Dla lepszego zobrazowania wyników utworzono wykresy(rys.3-4):



Rysunek 3 Wykres przedstawiający wartości minimalne oraz średnie bez indeksowania



Rysunek 4 Wykres przedstawiający wartości minimalne oraz średnie z indeksowania

## Wnioski:

Otrzymane wyniki pozwalają dokonać kilku wniosków:

- W *MySQL*, w przypadku tabeli zdenormalizowanej, dodanie indeksów nie wpłynęło w znaczący sposób na czas wykonywania zapytań, natomiast w przypadku tablicy znormalizowanej, indeksowanie wydłużyło czas wykonywania zapytania;
- *MySQL* szybciej wykonał zapytania obejmujące tablice znormalizowane szybciej niż w przypadku tablic zdenormalizowanych;
- W *PostgreSQL* indeksowanie nie wpłynęło w znaczący sposób na czas wykonywania zapytań. Jedynie w przypadku zapytania 2ZL da się zauważyć różnicę, co do której możemy być pewni, że nie wynika z błędu statystycznego;
- System *PostgreSQL* w większości przypadków był szybszy niż system *MySQL*. Wyjątek może stanowić jedynie zapytanie 3ZG, przy którym *MySQL* znacznie lepiej sobie poradził niż *PostgreSQL*;
- Porównywanie systemu *PostgreSQL* z systemem *MySQL* jest niejednoznaczne, tj. nie da się jednoznacznie określić który system jest „lepszy” – *PostgreSQL* w większości przypadków, wykonywał polecenia szybciej, lecz zapytanie 3GZ znacząco różni się czasem wykonywania od innych zapytań, *MySQL* wykonywał wszystkie zapytania w porównywalnym czasie i różnice między nimi nie były wielkie;

Reasumując można zauważyć znaczącą przewagę postaci znormalizowanej nad postacią zdenormalizowaną. Widać to w szczególności w przypadku zapytania 3ZG dla *PostgreSQL*, gdzie różnica ta wynosi kilka sekund. Wyniki te są zaskakujące, gdyż teoretycznie powinny wyjść wyniki przeciwne. Trudno jest stwierdzić, z czego to wynika. Problem należy poddać dalszej analizie.

## Załącznik 1

fanerozoik	kenozoik	czwartorzęd	holocen	megalaj
				northgrip
				grenland
			plejstocen	późny[b]
				chiban
				kalabr
				gelas
		neogen	pliocen	piacent
				zankl
			miocen	messyn
				torton
				serrawal
				lang
				burdygał
				akwitan
		paleogen	oligocen	szat
				rupel
			eocen	priabon
				barton
				lutet
				iprez
			paleocen	tanet
				zeland
				dan
	mezozoik	kreda	późna kreda	mastrycht
				kampan
				santon
				koniak
				turon
				cenoman
			wczesna kreda	alb
				apt
				barrem
				hoteryw
				walanżyn
				berrias
		jura	jura późna	tyton
				kimeryd
				oksford
			jura środkowa	kelowej

paleozoik	trias				baton	
					bajos	
					aalen	
			jura wczesna	toark		
				pliensbach		
				synemur		
		późny trias	hettang			
			retyk			
			noryk			
			karnik			
			środkowy trias	ladyn		
				anizyk		
	wczesny trias	olenek				
		ind				
	perm	loping			czangsing	
					wucziaping	
			gwadalup	kapitan		
				word		
				road		
			cisural	kungur		
				artinsk		
				sakmar		
				assel		
			karbon	pensylwan	późny pensylwan	gzel
						kasimow
		środkowy pensylwan			moskow	
					wczesny pensylwan	baszkir
		missisip		późny missisip	serpuchow	
				środkowy missisip	wizen	
wczesny missisip				turnej		
dewon		późny dewon		famen		
				fran		
		środkowy dewon		żywet		
				eifel		
		wczesny dewon		ems		
				prag		
lochkow		sylur	przydol			
	ludlow		ludford			
			gorst			

			wenlok	homer
				szejnwud
			landower	telicz
				aeron
				ruddan
		ordowik	ordowik późny	hirnant
				kat
				sandb
			ordowik środkowy	darriwil
				daping
			ordowik wczesny	flo
				tremadok
		kambr	furong	piętro 10[b]
				dziangszan
				paib
			miaoling	gužang
				drum
				wuliuan
			oddział 2	piętro 4
				piętro 3
			terenew	piętro 2
				fortun

Tabela stratygraficzna zgodna ze standardami ustalonymi przez *International Commission on Stratigraphy* na 2023/04



## Załącznik 2

1.	Pomiar czasu [ms]			
LP.	1ZL	2ZL	3ZG	4ZG
1.	213	564	11533	185
2.	189	555	11282	183
3.	183	556	11577	185
4.	185	558	11284	185
5.	191	569	11631	181
6.	186	595	11377	179
7.	191	614	11675	186
8.	190	558	11331	185
9.	193	555	11223	184
10.	186	554	11277	186
11.	189	585	11282	180
12.	188	537	11385	180
13.	192	554	11239	187
14.	187	538	11223	183
15.	196	536	11225	180

3.	Pomiar czasu [ms]			
LP	1ZL	2ZL	3ZG	4ZG
1.	1764	758	2639	765
2.	1765	756	2635	768
3.	1778	758	2590	758
4.	1760	744	2690	731
5.	1850	749	2738	741
6.	1811	754	2651	733
7.	1779	744	2614	738
8.	1787	773	2619	821
9.	1828	729	2767	741
10.	1839	746	2610	751
11.	1791	771	2725	741
12.	1746	739	2662	771
13.	1757	752	2766	812
14.	1761	750	2686	758
15.	1757	747	2686	750

2.	Pomiar czasu [ms]			
LP.	1ZL	2ZL	3ZL	4ZL
1.	184	417	11215	187
2.	181	376	11329	185
3.	185	381	11418	180
4.	192	381	11164	182
5.	185	375	11308	186
6.	180	385	11345	177
7.	184	387	11464	185
8.	184	377	11385	186
9.	181	383	11285	184
10.	184	385	11368	184
11.	191	374	11393	183
12.	189	375	11218	181
13.	180	384	11276	182
14.	185	389	11661	178
15.	187	389	11373	178

4.	Pomiar czasu [ms]			
LP.	1ZL	2ZL	3ZL	4ZL
1.	1734	2665	2515	2633
2.	1727	2694	2554	2622
3.	1766	2706	2581	2774
4.	1866	2679	2682	2674
5.	1769	2728	2539	2863
6.	1747	2662	2543	2755
7.	1722	2646	2599	2679
8.	1738	2722	2587	2702
9.	1825	2653	2523	2661
10.	1818	2719	2554	2671
11.	1798	2747	2583	2778
12.	1763	2682	2535	2703
13.	1723	2646	2571	2666
14.	1737	2743	2517	2712
15.	1725	2633	2629	2751

Dokładne wyniki pomiarów

1-PostgreSQL bez indeksów

2-PostgreSQL z indeksami

3-MySQL bez indeksów

4-MySQL z indeksami