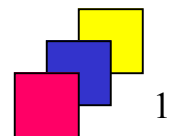


Optymalizacja poleceń SQL

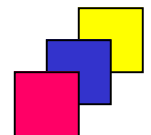
Metody dostępu do danych



Metody dostępu do danych

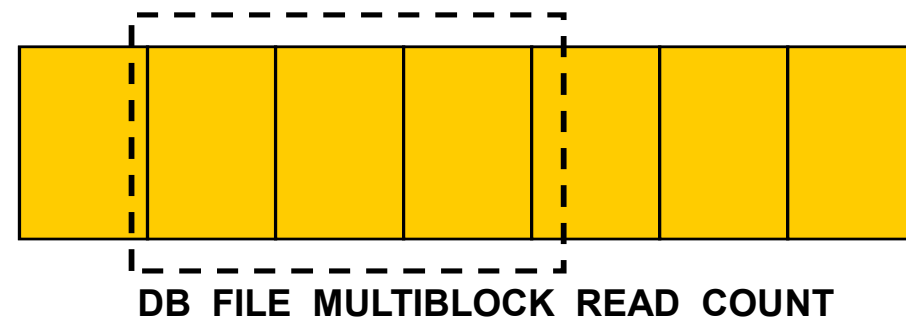
- **Określają, w jaki sposób dane polecenia SQL są odczytywane z miejsca ich fizycznej lokalizacji.**
- **Dostęp do tabeli:**
 - pełne przeglądnięcie,
 - dostęp przy pomocy adresu rekordu.
- **Dostęp do indeksu:**
 - unikalne przeglądnięcie indeksu,
 - (odwrócone) zakresowe przeglądnięcie indeksu,
 - przeglądnięcie indeksu z pominięciem kolumn,
 - pełne przeglądnięcie indeksu,
 - szybkie pełne przeglądnięcie indeksu,
 - dostęp do indeksu bitmapowego,
 - połączenie indeksów.

Przy dostępie do indeksu dane zwykle zwracane w kolejności rosnącej.
- **Ogólne zasady dostępu do danych:**
 - odczyt dużej części rekordów relacji – pełne przeglądnięcie relacji,
 - odczyt pojedynczych rekordów relacji – dostęp za pomocą indeksu.



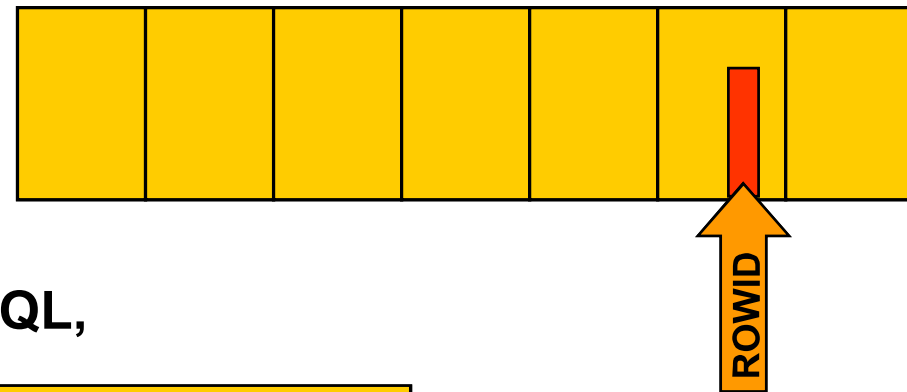
Pełne przeglądanie tabeli

- Ang. full table scan
- Sekwencyjny odczyt wszystkich bloków danych, w których tabela przechowuje swoje rekordy, odfiltrowanie rekordów nie spełniających zdefiniowanych w poleceniu SQL kryteriów selekcji (np. w klauzuli WHERE).
- Stosowane gdy:
 - brak indeksu dla relacji lub nie można użyć istniejących indeksów,
 - zostanie odczytana duża część wszystkich bloków, w których tabela składa swoje dane,
 - rozmiar tabeli jest niewielki.
- Możliwy odczyt wieloblokowy – pobranie w jednej operacji I/O wielu przyległych bloków danych, bardziej efektywne niż wiele odczytów pojedynczych bloków.



Dostęp przy pomocy adresu rekordu

- Ang. rowid scan
- Odszukanie rekordu relacji na podstawie dostarczonego adresu rekordu (rowid).
- Najszybszy sposób dostępu do rekordów tabeli.
- Źródło adresu rekordu:

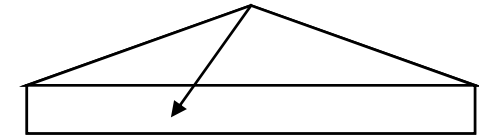


- warunek selekcji polecenia SQL,

```
SELECT * FROM pracownicy  
WHERE rowid = 'AAAMMUAAEAAAAAAtAAG';
```

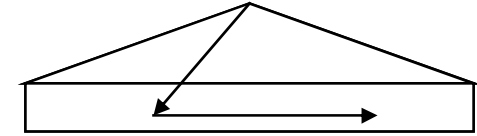
- pobranie z indeksu tabeli.

Unikalne przeglądnięcie indeksu



- **Ang. index unique scan**
- **Dostęp do indeksu unikalnego, operacja zwraca co najwyżej jeden adres rekordu.**
- **Stosowane, gdy w poleceniu SQL zastosowano warunek równościowy z atrybutem, na którym zdefiniowano indeks unikalny (również ograniczenia klucz podstawowy i klucz unikalny).**

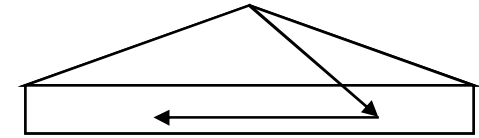
Zakresowe przeglądnięcie indeksu



- **Ang. index range scan**
- **Dostęp do indeksu unikalnego (warunek inny niż równościowy) lub nieunikalnego, operacja zwraca zakres adresów rekordów.**
- **Stosowane, gdy w poleceniu SQL:**
 - **warunek selekcji z kolumnami z części wiodącej indeksu, taki jak:**
 - **kolumna = 'wartość', kolumna > 'wartość', kolumna < 'wartość',**
 - **kolumna like 'ABC%' (% nie może być na początku wzorca),**
 - **warunek złożony z ww. warunków ze spójnikiem AND,**
 - **klauzula ORDER BY lub GROUP BY z atrybutami z części wiodącej indeksu.**

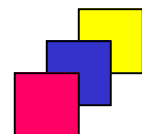
Odwrócone zakresowe przeglądnięcie indeksu

- **Ang. index range scan descending**
- **Odmiana zakresowego przeglądnięcia indeksu.**
- **Dane zwracane w kolejności malejącej .**
- **Stosowane, gdy:**
 - **w poleceniu konieczne posortowanie danych w porządku malejącym,**
 - **przy poszukiwaniu wartości mniejszych niż wartość wyspecyfikowana.**

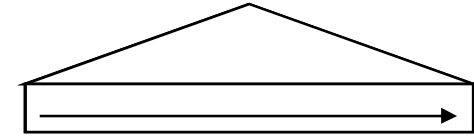


Przełknięcie indeksu z pominięciem kolumn

- **Ang. index skip scan**
- **Operacja korzystająca z indeksu złożonego dla polecenia, w którym nie występuje kolumna z początku części wiodącej klucza indeksowego:**
 - **indeks dzielony jest na mniejsze podindeksy, liczba podindeksów jest równa liczbie wartości pierwszej kolumny w kluczu indeksowym,**
 - **podindeksy skanowane są kolejno – operacja zastępuje pełne przełknięcie relacji.**
- **Przykład:**
 - **relacja Pracownicy(id_prac, adres, płeć), indeks o strukturze klucza: (płeć, id_prac), zapytanie: `select * from Pracownicy where id_prac = 100`**
 - **indeks zostaje podzielony na dwa podindeksy: dla wartości płeć = 'M' i dla wartości płeć = 'K', podindeksy zostają przeskanowane kolejno.**

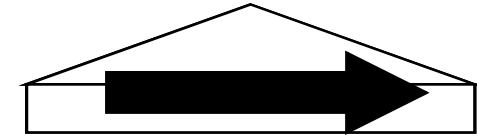


Pełne przeglądnięcie indeksu



- **Ang. full index scan**
- **Stosowane, gdy:**
 - w warunku polecenia SQL odwołania do kolumn z klucza indeksowego, kolumny nie muszą być częścią wiodącej klucza,
 - brak odwołań do poindeksowanych kolumn w warunku polecenia, ale:
 - wszystkie kolumny, do których występuje odwołanie w poleceniu (np. w klauzuli SELECT), znajdują się w kluczu indeksowym,
 - przynajmniej jedna z tych kolumn ma zdefiniowane ograniczenie NOT NULL.
- **Odczytywane są wszystkie liście indeksu w porządku, bloki indeksu odczytywane pojedynczo.**
- **Używane głównie do eliminacji operacji sortowania relacji – dane są posortowane wg klucza indeksowego.**

Szybkie pełne przeglądnięcie indeksu



- **Ang. fast full index scan**
- **Stosowane, gdy:**
 - **wszystkie kolumny, które są używane w poleceniu SQL, występują w kluczu indeksowym,**
 - **co najmniej jedna z tych kolumn ma zdefiniowane ograniczenie NOT NULL.**
- **Zastępuje pełne przeglądnięcie relacji – wynik polecenia SQL uzyskuje się bezpośrednio z indeksu, bez konieczności dostępu do relacji.**
- **Odczytywane są wszystkie liście indeksu przy zastosowaniu odczytu wieloblokowego – większa wydajność niż pełne przeglądnięcie indeksu, ale nie zostaje zachowane uporządkowanie.**
- **Nie może być używany do eliminacji operacji sortowania relacji – dane nie są posortowane wg klucza indeksowego.**

Dostęp do indeksu bitmapowego

- Składa się z dwóch kroków:
 1. dostęp do bitmapy,
 2. konwersja bitmapy do adresów rekordów (krok opuszczany w przypadku możliwości realizacji polecenia bez dostępu do relacji).
- W przypadku poleceń z warunkiem złożonym (spójniki AND i OR, negacja), operacje koniunkcji, alternatywy i negacji wykonywane bezpośrednio na bitmapach (widoczne w planie wykonania polecenia).

Połączenie indeksów

- Ang. index join
- Stosowane w przypadku, gdy wszystkie kolumny, używane w poleceniu SQL, znajdują się w kluczach kilku różnych indeksów.
- Wynik polecenia uzyskuje się tylko z indeksów, bez konieczności dostępu do relacji.
- Nie może być stosowane do eliminacji operacji sortowania relacji.
- Przykład:

SELECT id_prac **FROM** pracownicy **WHERE** placa_pod >1000;

Range scan(indeks na *placa_pod*)

placa_pod	ROWID
1600	00000001.001.001
...	...
2000	00000001.0A1.01E

Fast Full Scan(indeks na *id_prac*)

id_prac	ROWID
120	00000001.0A1.01E
...	...
140	00000001.001.001



join (hash)

120
...
140

Operacje sortowania (1)

- **SORT ORDER BY** – gdy w poleceniu wyrażenie **ORDER BY**.

```
SELECT * FROM zespoly  
ORDER BY adres DESC;
```

- **SORT AGGREGATE** – gdy w poleceniu wyliczana funkcji grupowa na całym zbiorze rekordów.

```
SELECT MAX(zatrudniony)  
FROM pracownicy;
```

- **SORT (HASH) GROUP BY** – gdy w poleceniu wyliczana funkcji grupowa dla kilku grup rekordów.

```
SELECT etat, AVG(placa_pod)  
FROM pracownicy GROUP BY etat;
```

Operacje sortowania (2)

- **SORT (HASH) UNIQUE** – gdy w poleceniu użyto klauzuli **DISTINCT**.

```
SELECT DISTINCT etat  
FROM pracownicy;
```

Uwaga!

Nie można zakładać uzyskania posortowanego zbioru rekordów przy operacjach **GROUP BY** i **DISTINCT**.

- **SORT JOIN** – przy wykonywaniu operacji połączenia wg algorytmu sort merge.

```
SELECT *  
FROM pracownicy JOIN etaty ON placa_pod between  
placa_min and placa_max;
```