

Национальный исследовательский университет ИТМО, г. Санкт-Петербург

Факультет ПиИКТ



Информационные системы и базы данных

Лабораторная работа № 4

Вариант: 37283

Работу выполнили:

Велюс Арина Костас

Группа: № Р33151

Преподаватель:

Байрамова Хумай Бахруз Кызы

г. Санкт-Петербург

2024

Задание:

По варианту, выданному преподавателем, составить и выполнить запросы к базе данных "Учебный процесс".

Команда для подключения к базе данных `uscheb`:

```
psql -h pg -d usheb
```

Составить запросы на языке SQL (пункты 1-2).

Для каждого запроса предложить индексы, добавление которых уменьшит время выполнения запроса (указать таблицы/атрибуты, для которых нужно добавить индексы, написать тип индекса; объяснить, почему добавление индекса будет полезным для данного запроса).

Для запросов 1-2 необходимо составить возможные планы выполнения запросов. Планы составляются на основании предположения, что в таблицах отсутствуют индексы. Из составленных планов необходимо выбрать оптимальный и объяснить свой выбор.

Изменяются ли планы при добавлении индекса и как?

Для запросов 1-2 необходимо добавить в отчет вывод команды EXPLAIN ANALYZE [запрос]

Реализация запросов на SQL:

1. Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:
Таблицы: Н_ОЦЕНКИ, Н_ВЕДОМОСТИ.
Вывести атрибуты: Н_ОЦЕНКИ.КОД, Н_ВЕДОМОСТИ.ИД.
Фильтры (AND):
 - а) Н_ОЦЕНКИ.КОД < 5.
 - б) Н_ВЕДОМОСТИ.ДАТА > 2010-06-18.
 - в) Н_ВЕДОМОСТИ.ДАТА = 2010-06-18.Вид соединения: RIGHT JOIN.

Запрос:

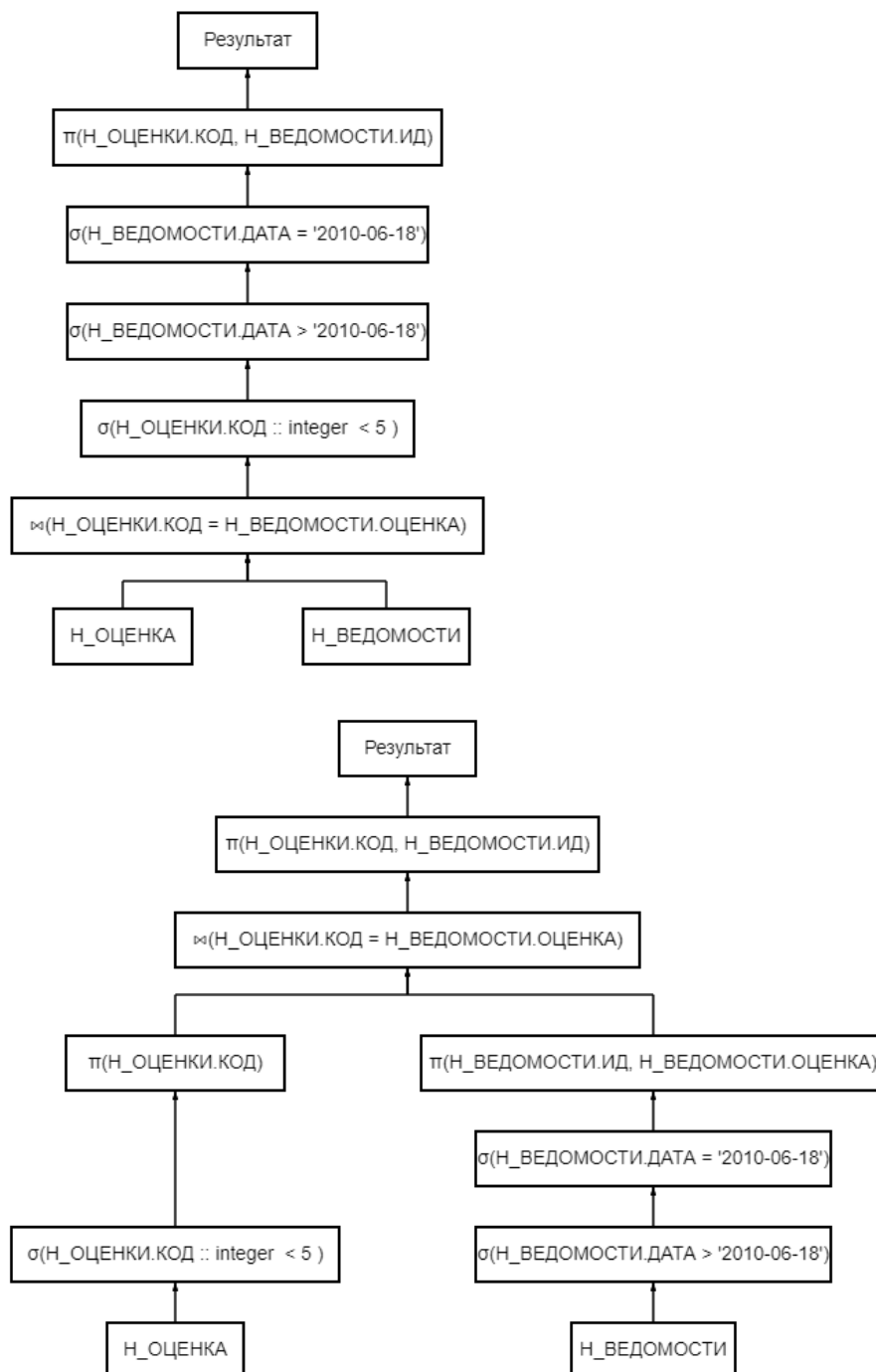
```
SELECT Н_ОЦЕНКИ.КОД, Н_ВЕДОМОСТИ.ИД  
FROM Н_ОЦЕНКИ  
RIGHT JOIN Н_ВЕДОМОСТИ ON Н_ОЦЕНКИ.КОД = Н_ВЕДОМОСТИ.ОЦЕНКА  
WHERE Н_ОЦЕНКИ.КОД :: integer < 5  
AND Н_ВЕДОМОСТИ.ДАТА > '2010-06-18'  
AND Н_ВЕДОМОСТИ.ДАТА = '2010-06-18';
```

Индексы:

Индексы позволяют эффективно выбирать строки, основываясь на их положение в дереве. Тип B-tree эффективнее в условии сравнения $<$, $>$, $=$

- CREATE INDEX ИНД_КОД ON Н_ОЦЕНКИ USING btree (КОД);
- CREATE INDEX ИНД_ОЦЕНКА ON Н_ВЕДОМОСТИ USING btree (ОЦЕНКА);
- CREATE INDEX ИНД_ДАТА ON Н_ВЕДОМОСТИ USING btree (ДАТА);

План выполнения запросов:



Из построенных планов, эффективнее будет второй, так как сначала происходит операция выборки, а потом соединения с помощью RIGHT JOIN. Наша цель: уменьшить размер промежуточных данных => уменьшить число операций чтения/записи во внешнюю память.

Explain Analyze

```
WORK PLAN
-----
Nested Loop (cost=0.29..30.18 rows=3 width=9) (actual time=0.004..0.005 rows=0 loops=1)
  Join Filter: ((("Н_ОЦЕНКИ"."КОД")::text = ("Н_ВЕДОМОСТИ"."ОЦЕНКА")::text)
    -> Index Scan using "ВЕД_ДАТА_I" on "Н_ВЕДОМОСТИ" (cost=0.29..28.68 rows=8 width=10) (actual time=0.003..0.003 rows=0 loops=1)
      Index Cond: (("ДАТА" > '2010-06-18 00:00:00'::timestamp without time zone) AND ("ДАТА" = '2010-06-18 00:00:00'::timestamp without time zone))
    -> Materialize (cost=0.00..1.17 rows=3 width=5) (never executed)
      -> Seq Scan on "Н_ОЦЕНКИ" (cost=0.00..1.16 rows=3 width=5) (never executed)
        Filter: (("КОД")::integer < 5)
Planning Time: 1.336 ms
Execution Time: 0.070 ms
(9 строк)
```

Использован вложенный цикл для выполнения операции объединения таблиц. Вложенный цикл применяется, когда Н_ВЕДОМОСТИ отфильтрована по условию с использованием индекса, и затем для каждой строки из этой отфильтрованной таблицы выполняется поиск соответствующих строк в Н_ОЦЕНКИ. (временную таблицу не создали, потому что результаты Seq Scan не требовались для выполнения запроса)

2. Сделать запрос для получения атрибутов из указанных таблиц, применив фильтры по указанным условиям:
Таблицы: Н_ЛЮДИ, Н_ВЕДОМОСТИ, Н_СЕССИЯ.
Вывести атрибуты: Н_ЛЮДИ.ФАМИЛИЯ, Н_ВЕДОМОСТИ.ДАТА, Н_СЕССИЯ.ДАТА.
Фильтры (AND):
 - а) Н_ЛЮДИ.ОТЧЕСТВО > Александрович.
 - б) Н_ВЕДОМОСТИ.ЧЛВК_ИД > 142390.Вид соединения: LEFT JOIN.

Запрос:

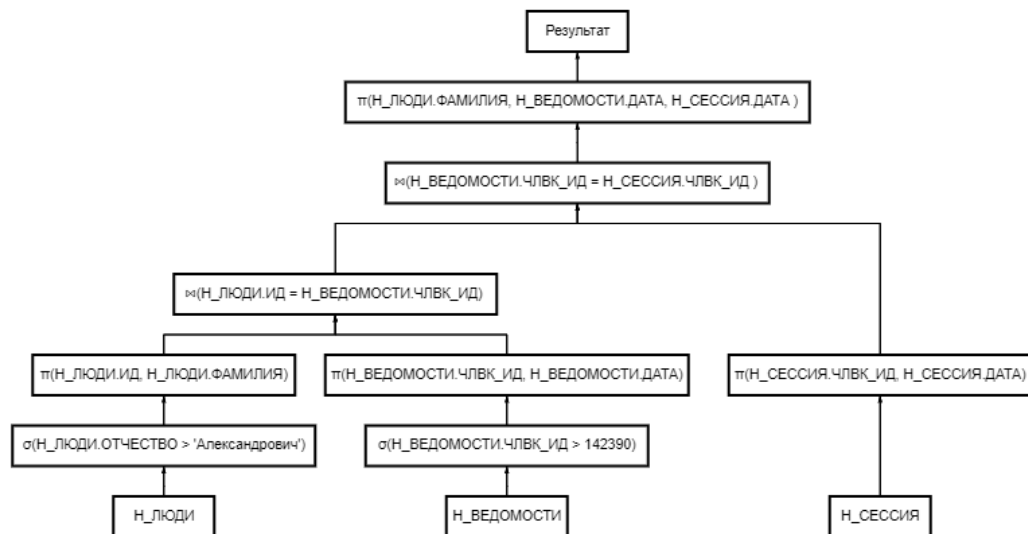
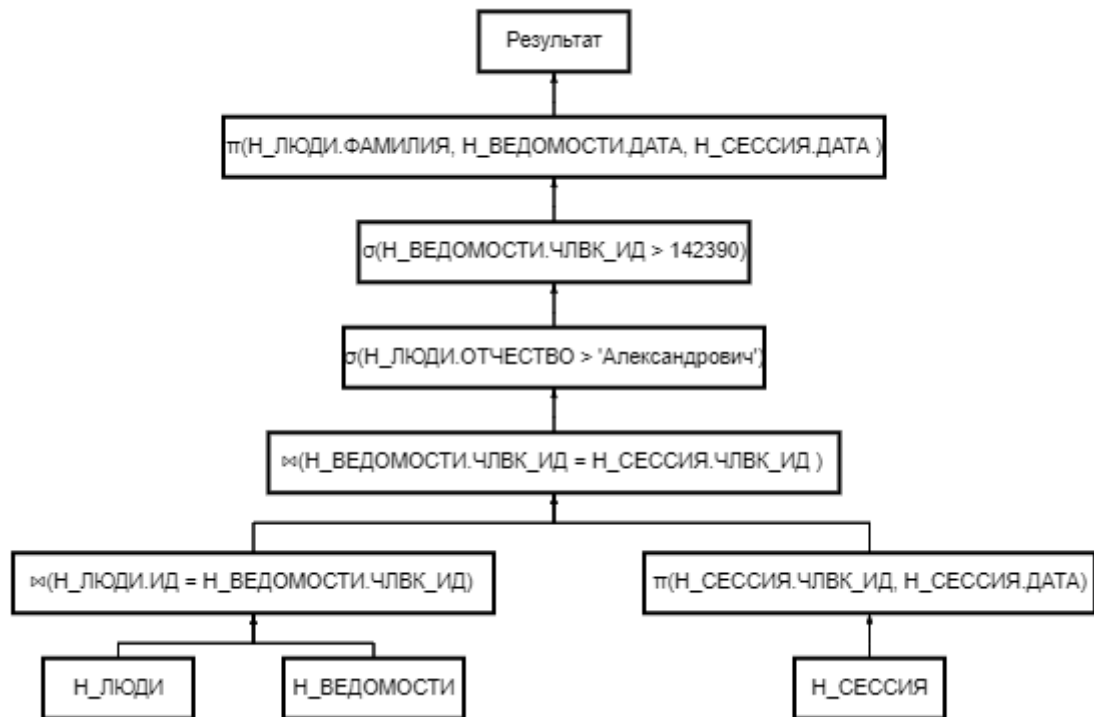
```
SELECT Н_ЛЮДИ.ФАМИЛИЯ, Н_ВЕДОМОСТИ.ДАТА, Н_СЕССИЯ.ДАТА
FROM Н_ЛЮДИ
LEFT JOIN Н_ВЕДОМОСТИ ON Н_ЛЮДИ.ИД = Н_ВЕДОМОСТИ.ЧЛВК_ИД
LEFT JOIN Н_СЕССИЯ ON Н_ВЕДОМОСТИ.ЧЛВК_ИД = Н_СЕССИЯ.ЧЛВК_ИД
WHERE Н_ЛЮДИ.ОТЧЕСТВО > 'Александрович'
AND Н_ВЕДОМОСТИ.ЧЛВК_ИД > 142390;
```

Индексы:

Индексы позволяют эффективно выбирать строки, основываясь на их положение в дереве. Тип B-tree эффективнее в условии сравнения <, >, =.

- CREATE INDEX ИНД_ОТЧЕСТВО ON Н_ЛЮДИ USING btree (ОТЧЕСТВО);
- CREATE INDEX ИНД_ЧЛВК_ИД ON Н_ВЕДОМОСТИ USING btree (ЧЛВК_ИД);

План выполнения запросов:



Из построенных планов, эффективнее будет второй. У всех трех таблиц большое количество строк. Для того, чтобы соединение происходило быстрее необходимо сначала совершить операцию выборки, а потом соединения с помощью LEFT JOIN. Наша цель: уменьшить размер промежуточных данных => уменьшить число операций чтения/запись во внешнюю память.

```

                                QUERY PLAN
-----
Hash Right Join (cost=5952.30..7372.76 rows=38187 width=32) (actual time=66.740..72.189 rows=36831 loops=1)
  Hash Cond: ("Н_СЕССИЯ"."ЧЛВК ИД" = "Н_ВЕДОМОСТИ"."ЧЛВК ИД")
    -> Seq Scan on "Н_СЕССИЯ" (cost=0.00..108.52 rows=3752 width=12) (actual time=0.016..0.448 rows=3752 loops=1)
    -> Hash (cost=5474.97..5474.97 rows=38187 width=28) (actual time=65.592..65.595 rows=36831 loops=1)
      Buckets: 65536 Batches: 1 Memory Usage: 2824kB
      -> Hash Join (cost=727.59..5474.97 rows=38187 width=28) (actual time=37.045..57.938 rows=36831 loops=1)
        Hash Cond: ("Н_ВЕДОМОСТИ"."ЧЛВК ИД" = "Н_ЛЮДИ"."ИД")
        -> Bitmap Heap Scan on "Н_ВЕДОМОСТИ" (cost=509.38..5138.42 rows=45043 width=12) (actual time=32.831..41.186 rows=45630 loops=1)
          Recheck Cond: ("ЧЛВК ИД" > 142390)
          Heap Blocks: exact=1652
          -> Bitmap Index Scan on "ВЕД_ЧЛВК_FK_IFK" (cost=0.00..498.12 rows=45043 width=0) (actual time=32.603..32.603 rows=45630 loops=1)
            Index Cond: ("ЧЛВК ИД" > 142390)
        -> Hash (cost=163.97..163.97 rows=4339 width=20) (actual time=4.168..4.169 rows=4339 loops=1)
          Buckets: 8192 Batches: 1 Memory Usage: 294kB
          -> Seq Scan on "Н_ЛЮДИ" (cost=0.00..163.97 rows=4339 width=20) (actual time=0.011..3.325 rows=4339 loops=1)
            Filter: (("ОТЧЕСТВО")::text > 'Александрович'::text)
            Rows Removed by Filter: 779
Planning Time: 1.105 ms
Execution Time: 73.927 ms
(19 строк)

```

У нас происходит объединение «Н_СЕССИЯ» и «Н_ВЕДОМОСТИ» с использованием хеш-объединения (вычисляем хеш-значения столбца, по которому происходит объединения, потом используем их для быстрого поиска соответствующих строк в хеш-таблице, простроенной для левой таблицы). Для этого читаются индексы и создается битовая карта, после чего сканируем кучи с использованием битовой карты.

Вывод:

При выполнении данной лабораторной работы я проанализировала два запроса и составила планы их выполнения. Также использовала индексы, изучила их особенности.