PostgreSQL для администраторов и разработчиков

Оптимизация настроек и структуры базы данных для повышения эффективности работы запросов

Меня хорошо видно & слышно?

👀 🔊

Защита проекта

Тема: Оптимизация настроек и структуры базы данных для повышения эффективности работы запросов.

Фёдор Сныткин

Специалист группы администрирования БД. АО «АЛСЕКО».
Сбор и обработка данных, разработка ПО, тех. сопровождение.

План защиты

- Цель и задачи проекта
- Какие технологии использовались
- Что получилось
- Выводы
- Вопросы и рекомендации

Цель и задачи проекта

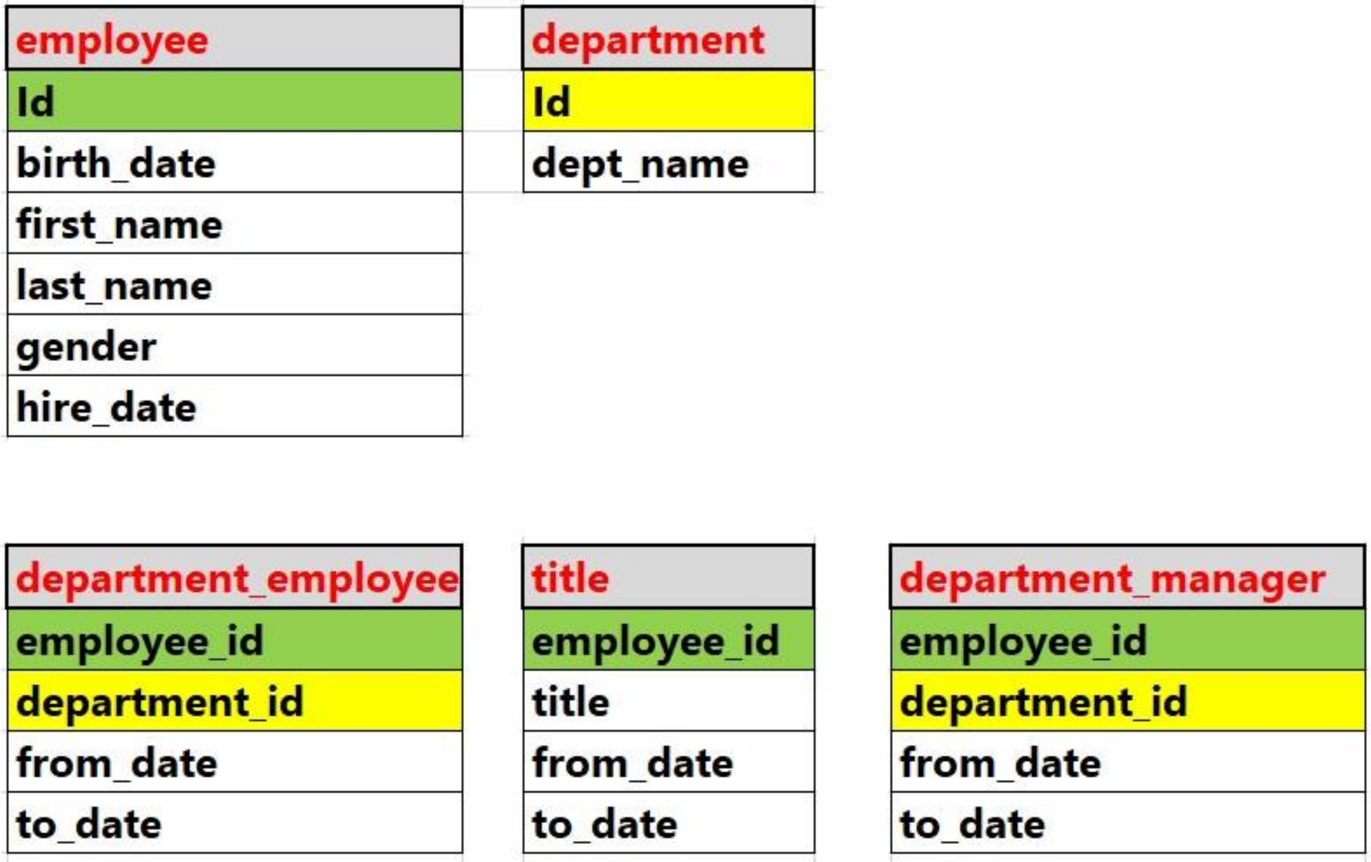
Цель проекта: оптимизировать структуру данных произвольного DataSet для оптимальной работы разработанных под его структуру запросов.

1. Развернуть виртуальную машину с СУБД и восстановить на ней DataSet (только данные).
2. Разработать запросы к DataSet и провести анализ их планов.
3. Оптимизировать структуру базы DataSet под разработанные запросы.
4. Отработать запросы и провести анализ их производительности.

Какие технологии использовались

1. Создание виртуальной машины на Oracle VM Virtual Box
2. Установка на VM ОС - Ubuntu 22.04.
3. Установка PostgreSQL15 и настройка по рекомендациям PGTune.
4. Доп. средства - PuTTY, WinSCP, Vim, mc …

Структара данных базы employees



employee
Id
birth_date
first_name
last_name
gender
hire_date

department
Id
dept_name

department_employee
employee_id
department_id
from_date
to_date

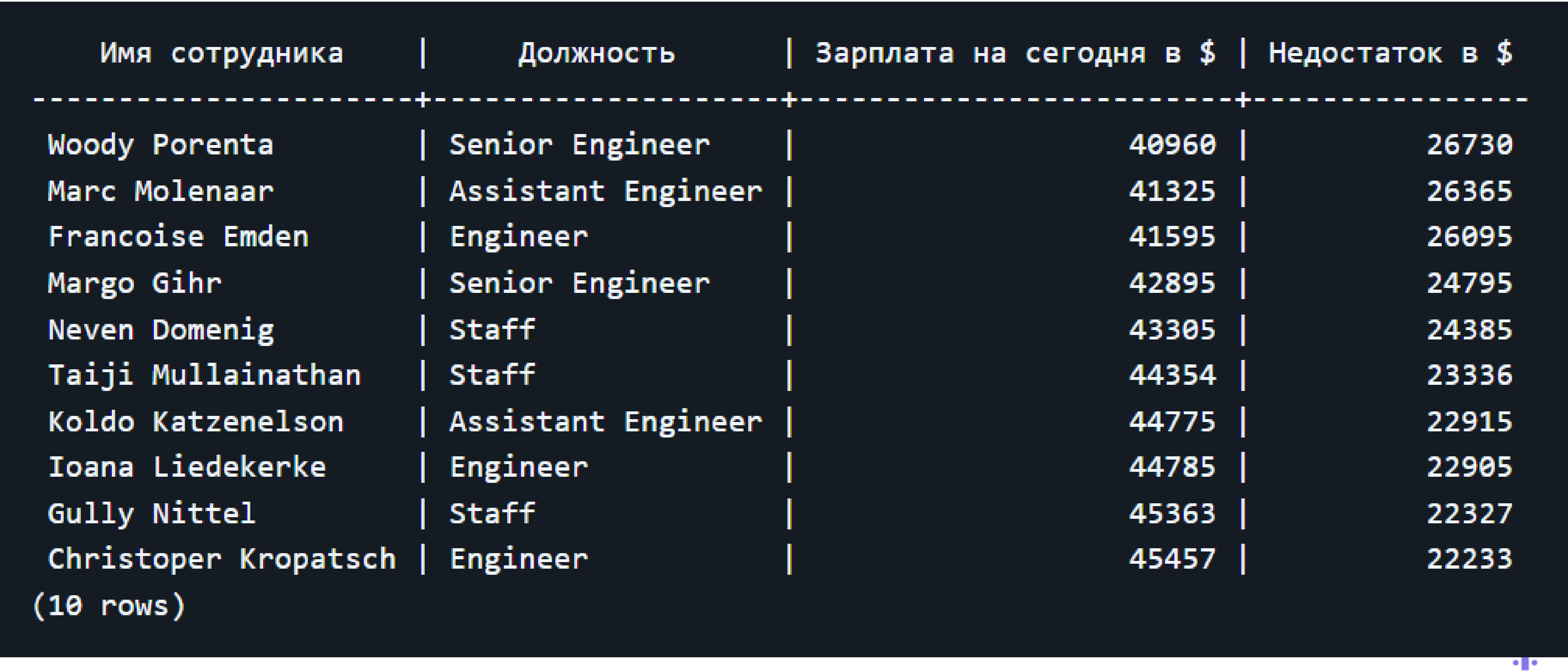
title
employee_id
title
from_date
to_date

department_manager
employee_id
department_id
from_date
to_date

Запрос Select

```sql
select
 e.first_name ||' '|| e.last_name AS "Имя сотрудника",
 t.title AS "Должность",
 s.amount AS "Зарплата на сегодня в $",
 67690 - s.amount AS "Недостаток в $"
 from employees.salary as s
Join employees.employee as e
 on s.employee_id = e.id
join employees.title as t
 on s.employee_id = t.employee_id
join employees.department_employee as de
 on e.id = de.employee_id
join employees.department as d
 on d.id = de.department_id
 where
  s.from_date >= (select max(from_date) from employees.salary)
   and s.to_date = (select max(to_date) from employees.salary)
   and t.to_date > (select current_date)
   and de.to_date > (select current_date)
   and s.amount < 67690
ORDER BY s.amount limit 10;
```

Результат запроса Select limit 10



```
   Имя сотрудника     |      Должность       | Зарплата на сегодня в $ | Недостаток в $
----------------------+----------------------+-------------------------+----------------
 Woody Porenta        | Senior Engineer      |                   40960 |          26730
 Marc Molenaar        | Assistant Engineer   |                   41325 |          26365
 Francoise Emden      | Engineer             |                   41595 |          26095
 Margo Gihr           | Senior Engineer      |                   42895 |          24795
 Neven Domenig        | Staff                |                   43305 |          24385
 Taiji Mullainathan   | Staff                |                   44354 |          23336
 Koldo Katzenelson    | Assistant Engineer   |                   44775 |          22915
 Ioana Liedekerke     | Engineer             |                   44785 |          22905
 Gully Nittel         | Staff                |                   45363 |          22327
 Christoper Kropatsch | Engineer             |                   45457 |          22233
(10 rows)
```

Запрос Update

```
update employees.salary
 set amount = (67690 + (amount/3))
where employee_id in
 (select s.employee_id
  from employees.salary as s
  Join employees.employee as e
   on s.employee_id = e.id
  join employees.title as t
   on s.employee_id = t.employee_id
  join employees.department_employee as de
   on e.id = de.employee_id
  join employees.department as d
   on d.id = de.department_id
  where
   s.from_date >= (select max(from_date) from employees.salary)
   and s.to_date = (select max(to_date) from employees.salary)
   and t.to_date > (select current_date)
   and de.to_date > (select current_date)
   and s.amount < 67690
)
and from_date >= (select max(from_date) from employees.salary)
and to_date = (select max(to_date) from employees.salary)
and amount < 67690;
```

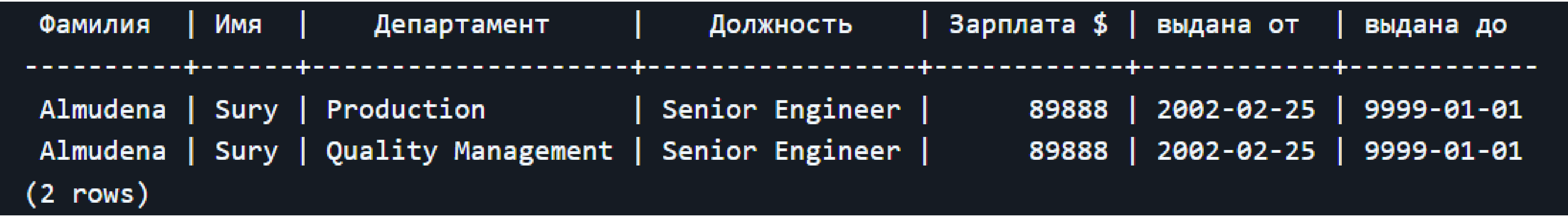
Структура индексов для запросов

```
CREATE UNIQUE INDEX idx_employee_id_primary ON employees.employee USING btree (id);
CREATE UNIQUE INDEX idx_department_id_primary ON employees.department USING btree (id);
CREATE INDEX idx_title_todate ON employees.title USING btree (to_date);
CREATE INDEX idx_title_empid ON employees.title USING btree (employee_id);
CREATE INDEX idx_salary_empid ON employees.salary USING btree (employee_id);
CREATE INDEX idx_salary_todate ON employees.salary USING btree (to_date);
CREATE INDEX idx_salary_fromdate ON employees.salary USING btree (from_date);
CREATE INDEX idx_salary_amount ON employees.salary USING btree (amount);
CREATE INDEX idx_department_employee_to_date ON employees.department_employee USING btree
(to_date);
CREATE INDEX idx_department_employee_depid ON employees.department_employee USING btree
(department_id);
CREATE INDEX idx_department_employee_employeeid ON employees.department_employee USING
btree (employee_id);
```

Запрос Select

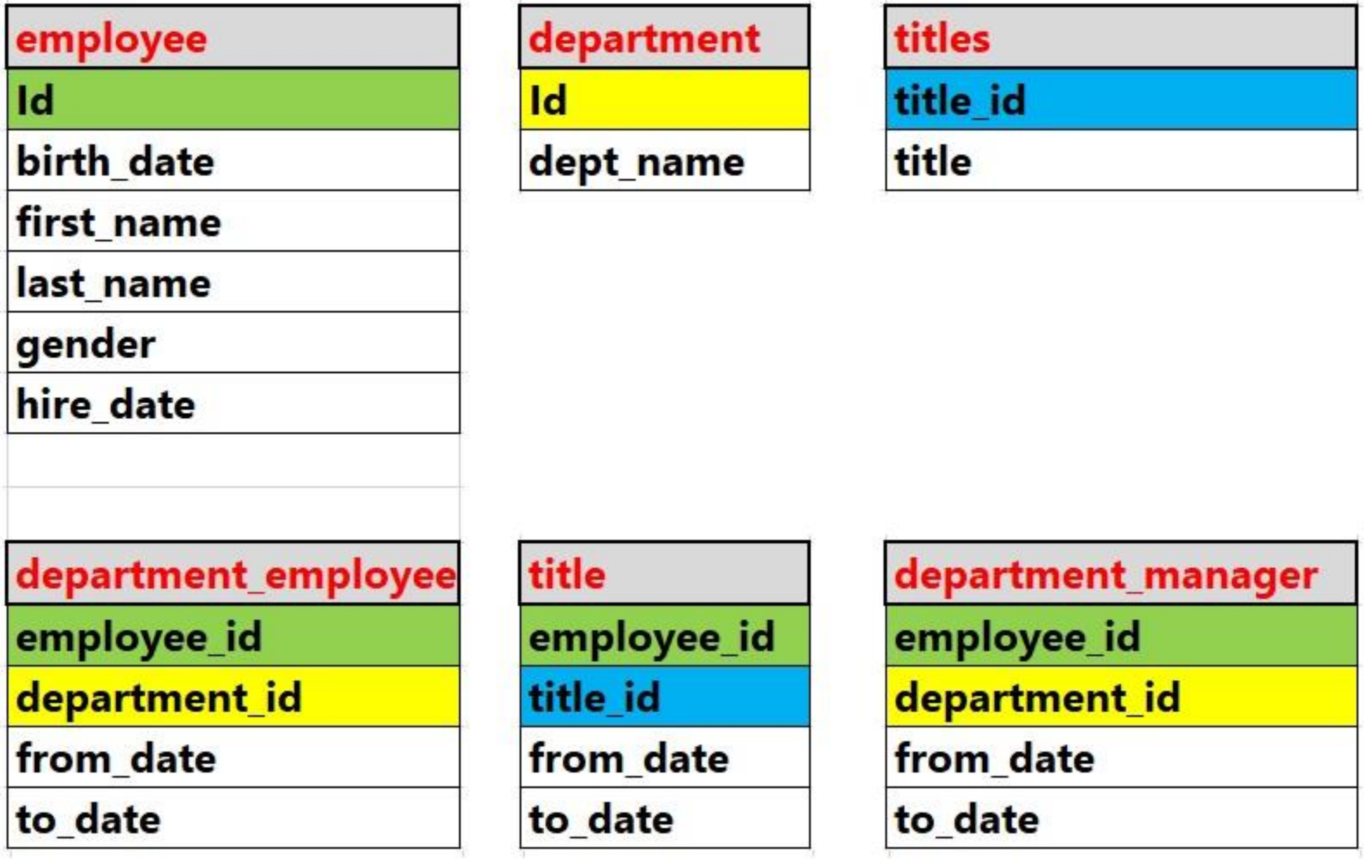
```sql
Select
 e.first_name AS "Фамилия",
 e.last_name AS "Имя",
 d.dept_name AS "Департамент",
 t.title AS "Должность",
 s.amount AS "Зарплата $",
 s.from_date AS "выдана от",
 s.to_date  AS "выдана до"
 from employees.employee as e
join employees.department_employee as de
 on e.id = de.employee_id
join employees.department as d
 on de.department_id = d.id
join employees.title as t
 on e.id = t.employee_id
join employees.salary as s
 on e.id = s.employee_id
where
 e.first_name like 'Almudena'
and e.last_name like 'Sur%'
and t.to_date = (select max(to_date) from employees.title)
and s.to_date = (select max(to_date) from employees.salary);
```

Результат запроса Select



```
 Фамилия  | Имя  |    Департамент     |    Должность    | Зарплата $ | выдана от  | выдана до
----------+------+--------------------+-----------------+------------+------------+------------
 Almudena | Sury | Production         | Senior Engineer |      89888 | 2002-02-25 | 9999-01-01
 Almudena | Sury | Quality Management | Senior Engineer |      89888 | 2002-02-25 | 9999-01-01
(2 rows)
```

Структара данных базы employees нормализована



employee
Id
birth_date
first_name
last_name
gender
hire_date

department
Id
dept_name

titles
title_id
title

department_employee
employee_id
department_id
from_date
to_date

title
employee_id
title_id
from_date
to_date

department_manager
employee_id
department_id
from_date
to_date

Запрос Select доработан

```
Select t_name AS "Фамилия",
 e.first_name AS "Фамилия",
 e.last_name AS "Имя",
 d.dept_name AS "Департамент",
 ts.title AS "Должность",
 s.amount AS "Зарплата $",
 s.from_date AS "выдана от",
 s.to_date  AS "выдана до"
 from employees.employee as eloyee as e
join employees.department_employee as de
 on e.id = de.employee_id
join employees.department as d
 on de.department_id = d.id
join employees.title as t
 on e.id = t.employee_id
join employees.titles as ts
 on t.title_id = ts.title_id
join employees.salary as s
 on e.id = s.employee_id
Where e.first_name like 'Almudena'
and e.last_name like 'Sur%'
and t.to_date = (select max(to_date) from employees.title)
and s.to_date = (select max(to_date) from employees.salary);
```

Доработана структура индексов

```sql
CREATE UNIQUE INDEX idx_employee_id_primary ON employees.employee USING btree (id);
CREATE INDEX idx_employee_firstname_lastname ON employees.employee USING btree
(first_name, last_name);
CREATE UNIQUE INDEX idx_department_id_primary ON employees.department USING btree
(id);
CREATE INDEX idx_title_todate ON employees.title USING btree (to_date);
CREATE INDEX idx_title_empid ON employees.title USING btree (employee_id);
CREATE INDEX idx_title_titleid ON employees.title USING btree (title_id);
CREATE INDEX idx_salary_empid ON employees.salary USING btree (employee_id);
CREATE INDEX idx_salary_todate ON employees.salary USING btree (to_date);
CREATE INDEX idx_department_employee_depid ON employees.department_employee USING
btree (department_id);
CREATE INDEX idx_department_employee_employeeid ON employees.department_employee
USING btree (employee_id);
```

Выводы

1. Достижение цели – цель достигнута.

2. Возникшие проблемы – Трудно было понять логику работы оптимизатора.

3. Временная оценка – 2 недели.

4. Личная польза – 9 баллов из 10.

5. Вопросы по проекту – нет.

6. Дальнейшее развитие – Изучение логики оптимизатора запросов PostgreSql с учетом изменений структуры БД на физическом уровне.

Вопросы и рекомендации

+ если есть вопросы

— если вопросов нет

Спасибо за внимание!✨