



PostgreSQL

Секционирование



Проверить, идет ли запись

Меня хорошо видно && слышно?



Ставим "+", если все хорошо
"-", если есть проблемы



Тема вебинара

Секционирование (partition) в PostgreSQL



Ведущий разработчик PostgreSQL/Greenplum в Сбере

Специалист в области разработки и проектирования витрин данных в PostgreSQL/Greenplum, а также в области разработки хранимых процедур в таких СУБД как PostgreSQL/Greenplum, Oracle, MS SQL Server.



Правила вебинара



Активно
участвуем



Off-topic обсуждаем
в учебной группе ТГ



Задаем вопрос
в чат или голосом



Вопросы вижу в чате,
могу ответить не сразу



Маршрут вебинара

1. Термин секционирование (партиционирование)

2. История развития секционирования в PostgreSQL

3. Виды секционирования

4. Практика

Цели вебинара

К концу занятия вы сможете

1. Создавать секционированные таблицы;
2. Обслуживать секционированные таблицы;
3. Увеличить производительность запросов.

Смысл

Зачем вам это уметь

1. Для лучшего хранения большого количества данных;
2. Для более быстрого доступа к данным;
3. Для ускорения запросов.

Что мы хотим сделать

- **Есть таблица на десять миллионов строк:**

```
create table orders (  
    order_id bigint generated always as identity primary key,  
    client varchar(63) not null,  
    order_date date not null,  
    order_total numeric(12, 2)  
);
```

- **По таблице часто необходимо выполнять запросы вида:**

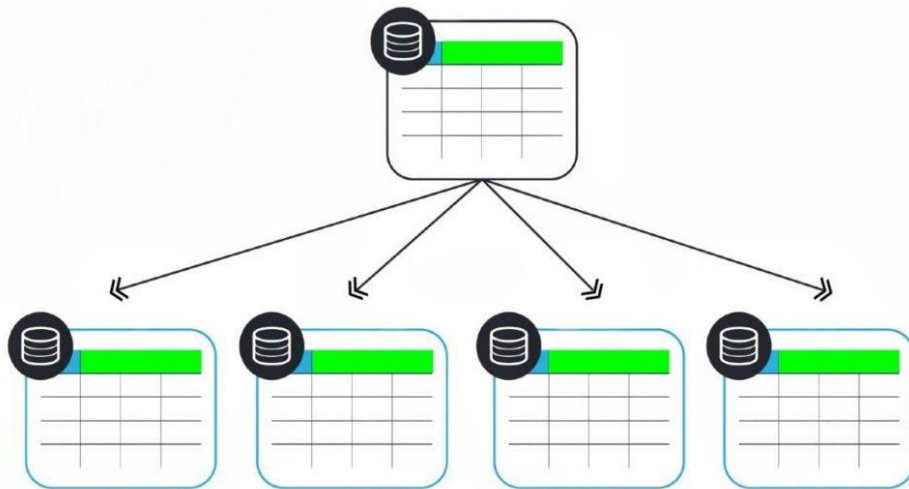
```
select * from orders  
where order_date = 'некоторая дата'  
group by client;
```

- **Каким образом мы можем увеличить скорость выполнения подобных запросов?**

Терминология

Секционирование

- **Партиционирование (partitioning)** — это способ повышения производительности запросов за счет логического разбиения больших таблиц (например, таблиц фактов) на небольшие части, называемые партициями (partitions). Партиционирование позволяет оптимизаторам запросов сканировать ограниченное число строк в таблице (на основе условий предикатов) вместо чтения всего содержимого таблицы.



История Inheritance & Declaration

<https://habr.com/ru/companies/postgrespro/articles/353472/>

Inheritance & Declaration

Секционирование наследованием:

- До версии 10 секционирование в PostgreSQL базировалось на механизме наследования таблиц, где дочерние таблицы создавались как наследники одной родительской.
- Запрос на чтение родительской таблицы включал данные не только из родительской, но и из дочерних таблиц, но последующие дополнения позволили делать выборочное чтение, когда ограничения целостности гарантировали, что данные не содержатся в дочерних таблицах.
- Запись происходит по-другому: вставка в родительскую таблицу не перенаправляется автоматически в соответствующие дочерние таблицы, требуется добавление триггера для перенаправления данных.
- Управлять секциями требует большой работы при создании, и необходимо гарантировать корректное создание секций с самого начала для корректной работы триггеров вставки.
- В целом, этот механизм может быть интересным в специфических случаях, но может создать проблемы, особенно в производительности, и в большинстве других случаев лучше обойтись без такого секционирования.

<https://postgrespro.ru/docs/postgresql/10/ddl-partitioning#DDL-PARTITIONING-IMPLEMENTATION-INHERITANCE>

Inheritance & Declaration

Декларативное секционирование:

- В версии 10 появилась новая система секционирования, основанная на инфраструктуре, уже существовавшей в PostgreSQL. Целью разработчиков было упрощение создания и администрирования секционированных таблиц. К уже существующим SQL-операторам (например CREATE TABLE и ALTER TABLE) добавлены специальные предложения для создания, присоединения/отсоединения секций. Вот список изменений:
 - Создание и администрирование секций упростилось благодаря интеграции в ядро СУБД.
 - Теперь триггеры больше не нужны. Вставка ускорилась, появилось автоматическое перенаправление вставляемых данных в правильную секцию, и генерируется ошибка в случае направления в неподходящую секцию.
 - При работе с секциями можно присоединять/отсоединять секции, есть явные ограничения целостности секций, возможно секционирование по выражению в ключе разбиения, и можно создавать подсекции.
 - Изменения в системном каталоге: новые столбцы в pg_class и новый каталог pg_partitioned_table.
- В упрощении создания секций легко убедиться: больше нет необходимости создавать специальные функции-триггеры для управления вставками и обновлениями. Данные теперь маршрутизируются автоматически функциями, создаваемыми при определении секции. В случае, когда для вводимых данных не находится подходящей секции, просто генерируется ошибка. Такая автоматическая маршрутизация не только удобна, она и ускоряет вставку.
- Каталог pg_class был модифицирован и теперь содержит информацию о том, является ли таблица секцией (если да, то: relispartition = 't'), секционирована ли таблица (если да, то: relkind = 'p') или это обычная таблица (relkind = 't'), и внутреннее представление границ секций (relpartbound).

<https://postgrespro.ru/docs/postgresql/10/ddl-partitioning#DDL-PARTITIONING-DECLARATIVE>

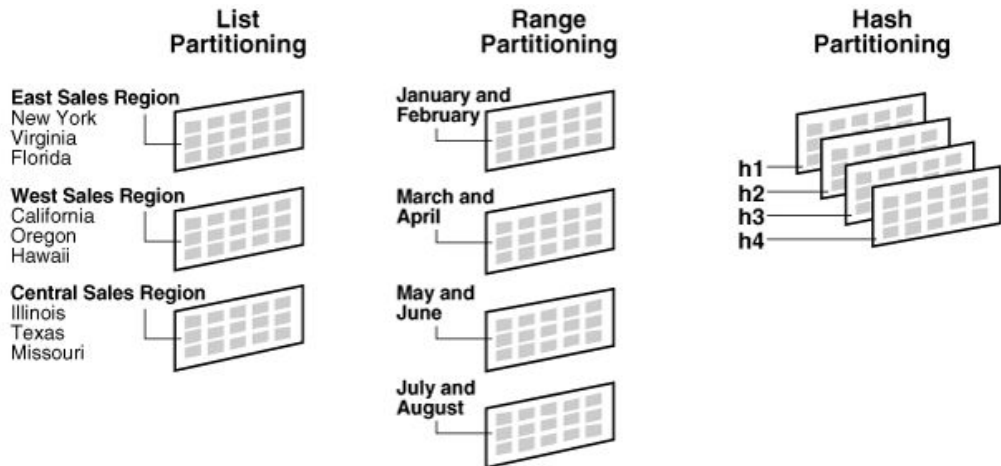
Виды секционирования

<https://habr.com/ru/companies/postgrespro/articles/353472/>



Виды секционирования

три вида:



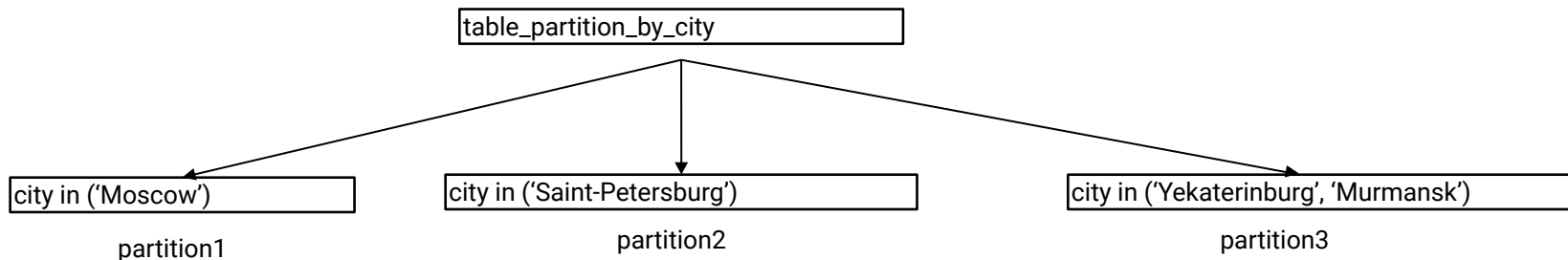
- Общий синтаксис для секционирования таблицы:
 - `PARTITION BY { RANGE | LIST | HASH } ({ имя_столбца | (выражение) } [COLLATE правило_сортировки] [класс_операторов] [...])`
 - `CREATE TABLE section_name PARTITION OF table_name FOR VALUES FROM ('2006-02-01') TO ('2006-03-01');`
- Удаление партиции:
 - `DROP TABLE section_name`

https://docs.oracle.com/cd/B10500_01/server.920/a96524/c12parti.htm

<https://postgrespro.ru/docs/postgresql/10/sql-createtable>

Виды секционирования

по списку:

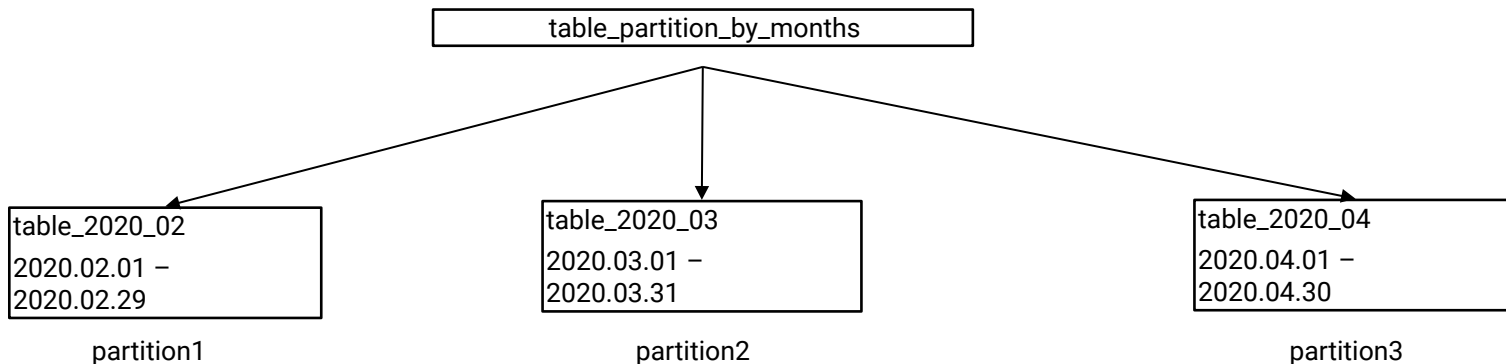


```
CREATE TABLE city_moscow PARTITION OF city FOR VALUES IN ('Moscow');
```



Виды секционирования

по диапазону:



```
CREATE TABLE table_2020_02 PARTITION OF table FOR VALUES FROM ('2020-02-01') TO ('2020-03-01');
```

Виды секционирования

по хэшу:

- Когда необходимо равномерное распределение;
- Нет явного ключа, по которому можно разбить таблицу;
- Для равномерного распределения необходимо уникальное или почти уникальное поле.

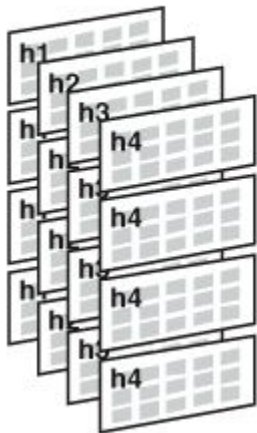


```
CREATE TABLE table_p1 PARTITION OF table FOR VALUES WITH (MODULUS 5, REMAINDER 0);
```

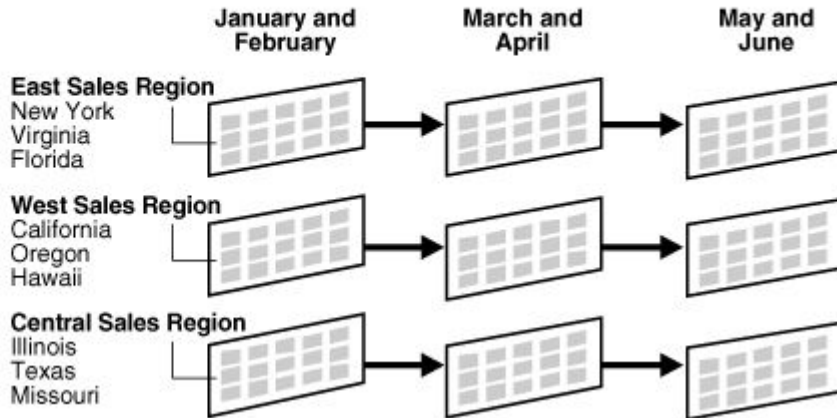
Виды секционирования

комбинирование (subpartition):

Composite Partitioning
Range-Hash



Composite Partitioning
Range - List



https://docs.oracle.com/cd/B10500_01/server.920/a96524/c12parti.htm

МИНИТЕСТ

<https://forms.gle/1qUxrgdv2F1ADpJ6A>

5 минут



Практика IDE DBeaver

Практика

При помощи наследования:

- Создается таблица и от неё наследуются секции с помощью команды:
 - `CREATE TABLE table_s1 () INHERITS (table);`
- В дочерних таблицах создаются ограничения, которые не пересекаются между секциями:
 - `CHECK (date between...)`
- Создается триггер, который распределяет строки:
 - `CREATE TRIGGER trigger_name
BEFORE INSERT ON table_name
FOR EACH ROW EXECUTE PROCEDURE func_name();`

Практика

Декларативный способ:

- Можно создавать дефолтную секцию:
 - `create table part_name partition of main_table default;`
- Можно использовать минимальное, максимальное значение для range partition
 - `create table part_name partition of main_table for values from (MINVALUE) to (MAXVALUE);`
- Можно отсоединять секции:
 - `alter table main_table detach partition part_name;`
- Можно добавлять секции:
 - `alter table main_table attach partition part_name for values from ('2020-01-01') to ('2020-02-01');`
- Не забыть включить **enable_partition_pruning** для оптимизации (*default on*).

Трудности:

- В *partition by range* не получится использовать в ключе секционирования *null* значения;
- Не умеет создавать секции самостоятельно (можно использовать *cron* и прочее);
- Не получится создать уникальное ограничение на часть.



Практика

pg_pathman:

- Доступно разбиение по хэшу и диапазонам Автоматическое создание секций при секционировании по диапазонам;
- Можно соединять или разделять секции;
- Не факт, что будут развивать в 14 версии (хотя тоже самое говорили про 13).

<https://postgrespro.ru/docs/postgrespro/13/pg-pathman>

https://github.com/postgrespro/pg_pathman

<https://www.programmersonsought.com/article/25792165890/>

Вопросы?

Рефлексия

<https://docs.google.com/forms/d/1DjKMeYpiPE9EYmrLVWZW5KI3W2feBmGVAHmGWduuuvo/edit#responses>

Домашнее задание

ДЗ

- Секционировать большую таблицу из демо-базы **flights**.

ДЗ

1. Выполнение ДЗ: 10 баллов
 - + 2 балла за красивое решение
 - - 2 балла за рабочее решение, и недостатки указанные преподавателем не устранены

2. Рекомендуемый путь задавать вопросы:
 - в чат дз в ЛК Отуса
 - или общий чат ТГ без тега преподавателя

**Заполните, пожалуйста,
опрос о занятии
по ссылке в чате**

Спасибо за внимание!

Приходите на следующие вебинары



Ведущий разработчик PostgreSQL/Greenplum в Сбере

Специалист в области разработки и проектировании витрин данных в PostgreSQL/Greenplum, а также в области разработки хранимых процедур в таких СУБД как PostgreSQL/Greenplum, Oracle, MS SQL Server.

