



Partitioning Расширенное секционирование в СУБД Postgres Pro

Владимир Пудовченко
28 января 2025 года

PGProDay



Что такое секционирование?

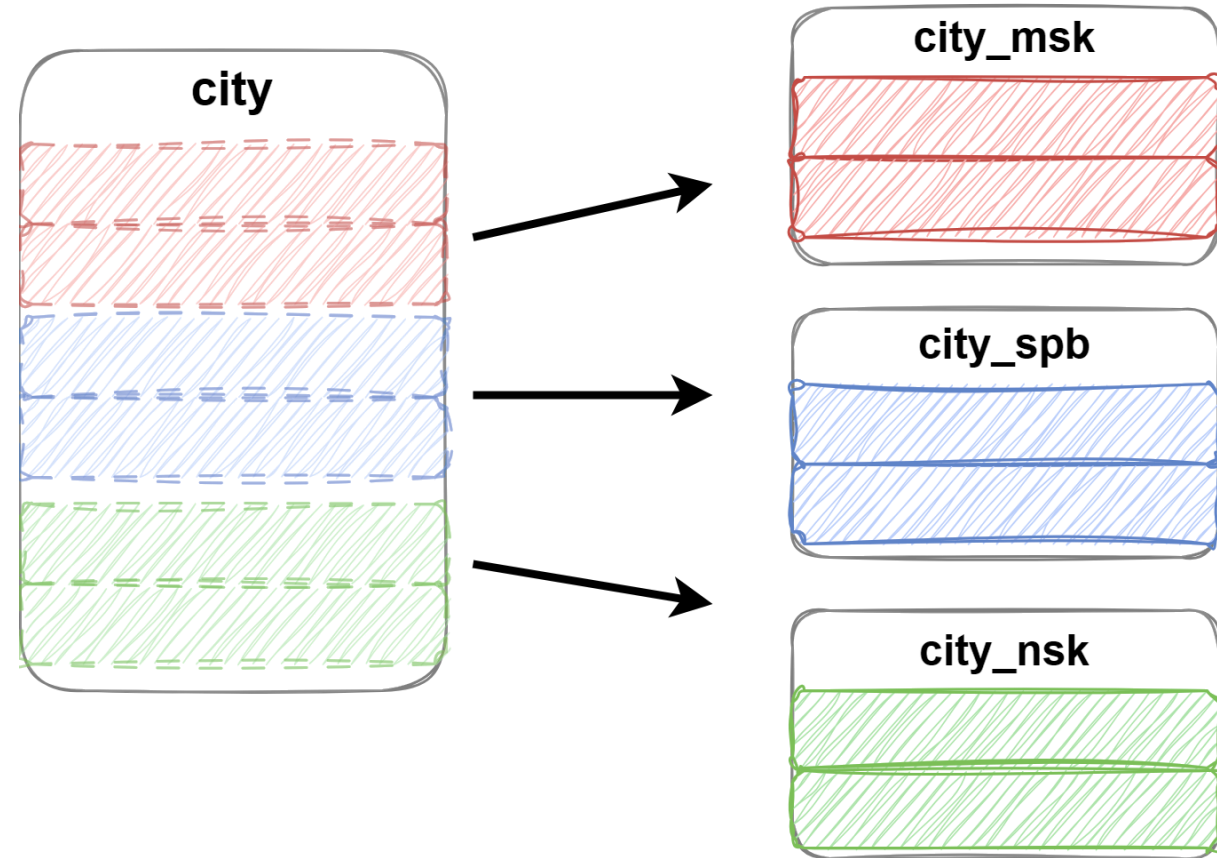


Что такое секционирование?

Под секционированием (partitioning) обычно понимают разделение одной логической таблицы на отдельные физические части (секции). При этом, для приложений таблица выглядит единой. Разделение проводится по строкам таблицы, на основании значений ключа секционирования.

Зачем нужно секционирование?

1. Повышение производительности запросов
2. Большой размер таблицы не позволяет производить обслуживание в рамках окна:
 - VACUUM FULL
 - REINDEX
 - pg_dump --jobs=n
 - и тд.
3. Перенос исторических секций на медленные(дешевые) носители.



Декларативное секционирование

Механизм, позволяющий декларировать таблицу как секционированную при ее создании, а также присоединять и отсоединять секции таблицы. Транзакционность DDL превращает этот механизм в очень мощный инструмент управления секционированными таблицами и предоставляет большие возможности для автоматизации.

```
CREATE TABLE measurement (  
    city_id          int not null,  
    logdate          date not null,  
) PARTITION BY RANGE (logdate);
```

```
CREATE TABLE measurement_y2006m02 PARTITION OF measurement  
    FOR VALUES FROM ('2006-02-01') TO ('2006-03-01');
```

```
CREATE TABLE measurement_y2006m03 PARTITION OF measurement  
    FOR VALUES FROM ('2006-03-01') TO ('2006-04-01');
```

Ограничения

- Все секции должны иметь в точности тот же набор столбцов, что и секционируемая таблица.
- Ключ секционирования может и должен включать только столбы первичного ключа. Аналогично с ограничением уникальности.

```
create table t ( id int primary key , n timestamp ) partition by range (n) ;
```

ОШИБКА: ограничение уникальности в секционированной таблице должно включать все секционирующие столбцы
ПОДРОБНОСТИ: В ограничении PRIMARY KEY таблицы "t" не хватает столбца "n", входящего в ключ секционирования.

- Уникальность поддерживается только по ключу секционирования (нельзя создать уникальный индекс по другим полям секционированной таблицы)
- Триггеры BEFORE ROW для INSERT не могут менять секцию, в которую в итоге попадёт новая строка.
- Нельзя декларативно секционировать существующую таблицу.

Стратегии секционирования

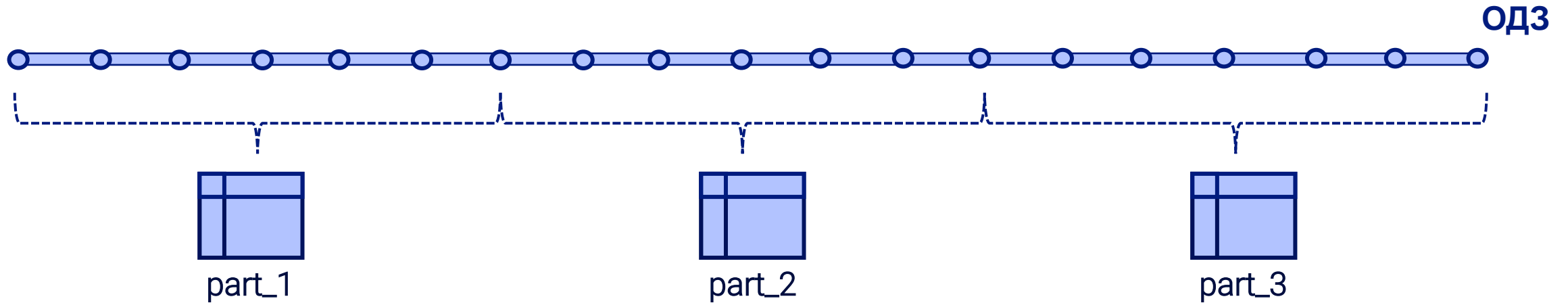
Декларативный вариант предлагает поддержку следующих стратегий секционирования:

1. по диапазонам (range)
2. по списку (list)
3. по хэшу (hash)

Ключ секционирования может быть составной.

При использовании декларативного секционирования для исключения секций нужно использовать параметр `enable_partition_pruning = on`

Секционирование по хэшу



```
CREATE TABLE orders_1 PARTITION OF orders FOR VALUES WITH (MODULUS 3, REMAINDER 0);
```

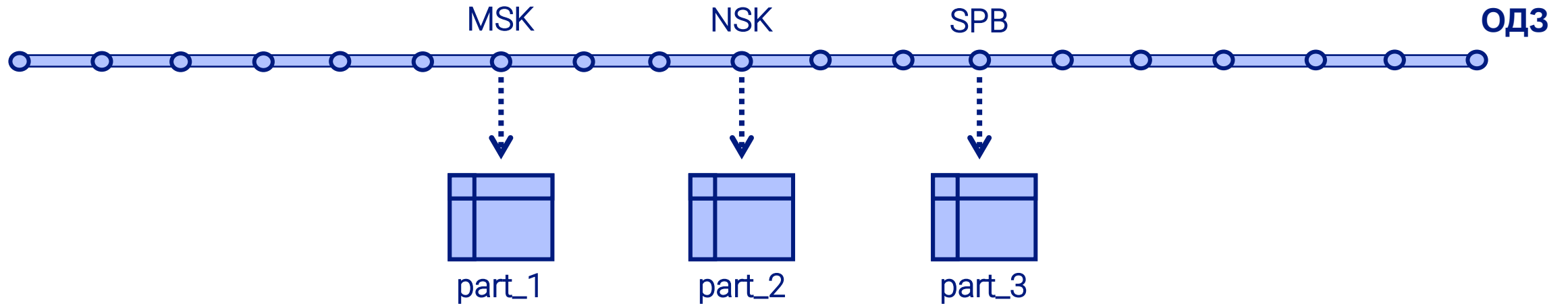
```
CREATE TABLE orders_2 PARTITION OF orders FOR VALUES WITH (MODULUS 3, REMAINDER 1);
```

```
CREATE TABLE orders_3 PARTITION OF orders FOR VALUES WITH (MODULUS 3, REMAINDER 2);
```

Признаком секционирования является остаток от деления значения ключа нацело.

- Количество секций объявляется сразу.
- Нельзя изменять общее количество секций, но можно выполнять attach и detach.
- Отсечка секций планировщиком работает только при эквисоединениях или в запросах с условием строгого равенства, т.к. операции сравнения для хэш-значений не поддерживаются.

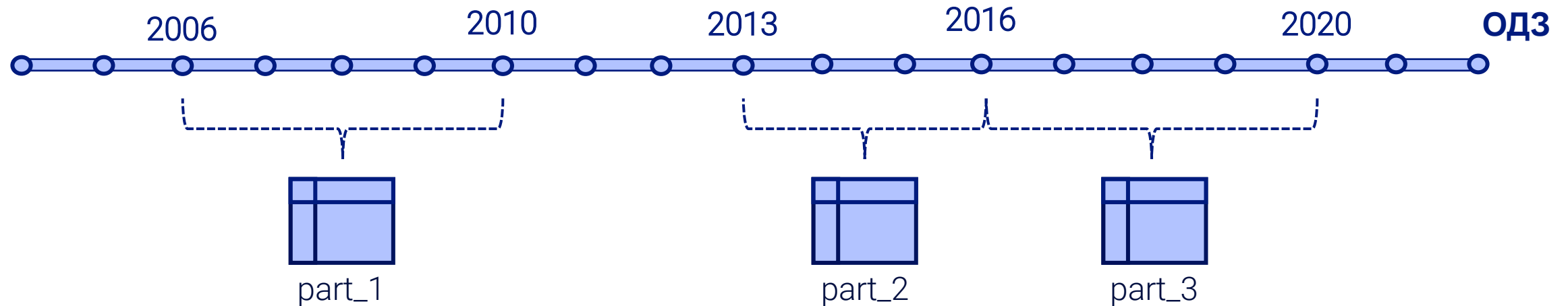
Секционирование по списку



```
CREATE TABLE part_1 PARTITION OF city FOR VALUES IN ('MSK');
CREATE TABLE part_2 PARTITION OF city FOR VALUES IN ('NSK');
CREATE TABLE part_3 PARTITION OF city FOR VALUES IN ('SPB');
```

Признаком секционирования является значение ключа, сопоставляемое с заданным списком значений

Секционирование по диапазонам



```
CREATE TABLE part_1 PARTITION OF sales FOR VALUES FROM ('2006-01-01') TO ('2010-01-01');
CREATE TABLE part_2 PARTITION OF sales FOR VALUES FROM ('2013-01-01') TO ('2016-01-01');
CREATE TABLE part_3 PARTITION OF sales FOR VALUES FROM ('2016-01-01') TO ('2020-01-01');
```

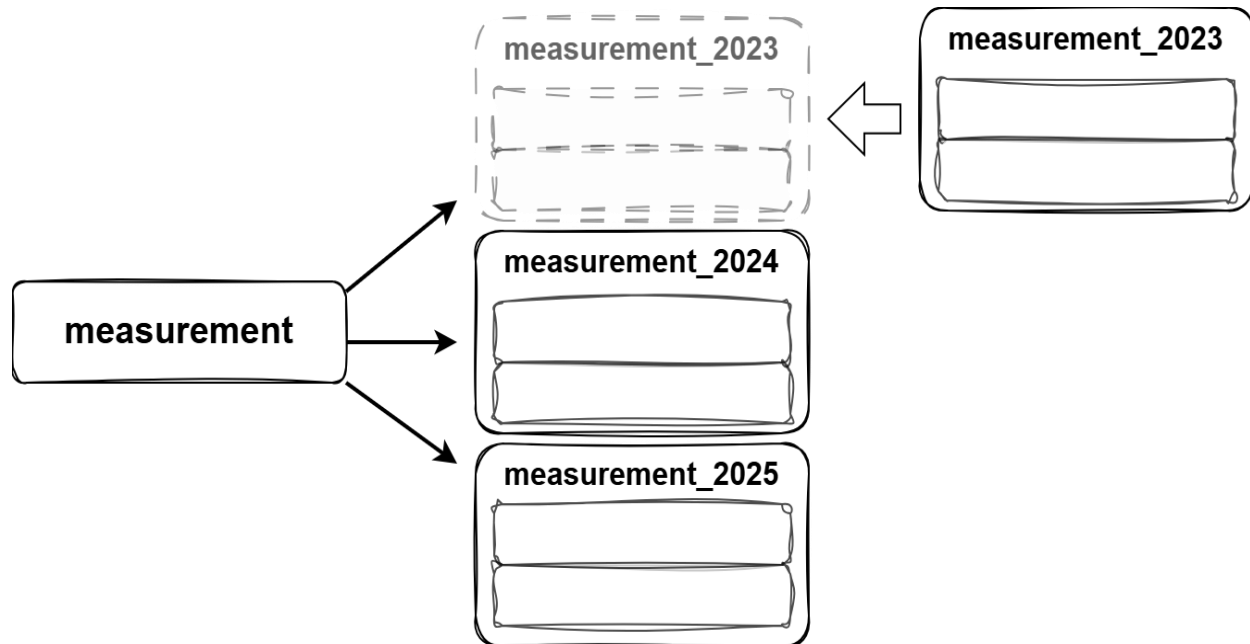
Признаком секционирования является значение ключа, сопоставляемое с заданным диапазоном

Ванильные методы работы с секциями

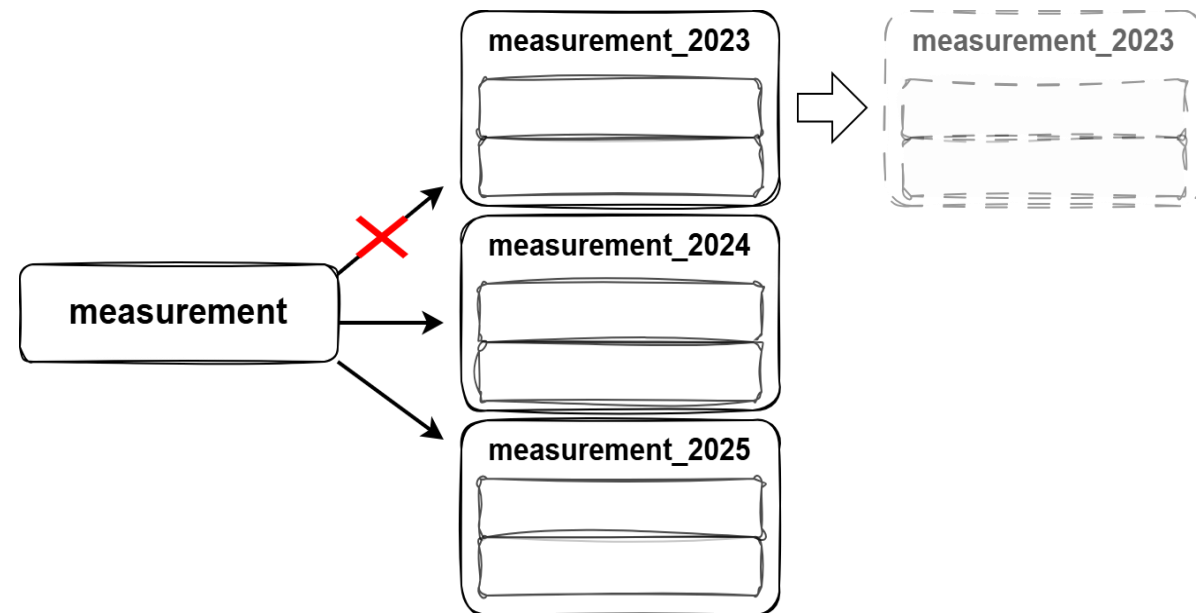
Работа с секциями ATTACH, DETACH

Присоединение и отсоединение секций

```
ALTER TABLE measurement ATTACH PARTITION
measurement_2023 FOR VALUES FROM ('2023-01-01') TO
('2024-01-01');
```



```
ALTER TABLE measurement DETACH
PARTITION measurement_2023;
```



Секция DEFAULT

В секцию «по умолчанию» попадают строчки, которые по ключу секционирования не попадают ни в одну секцию.

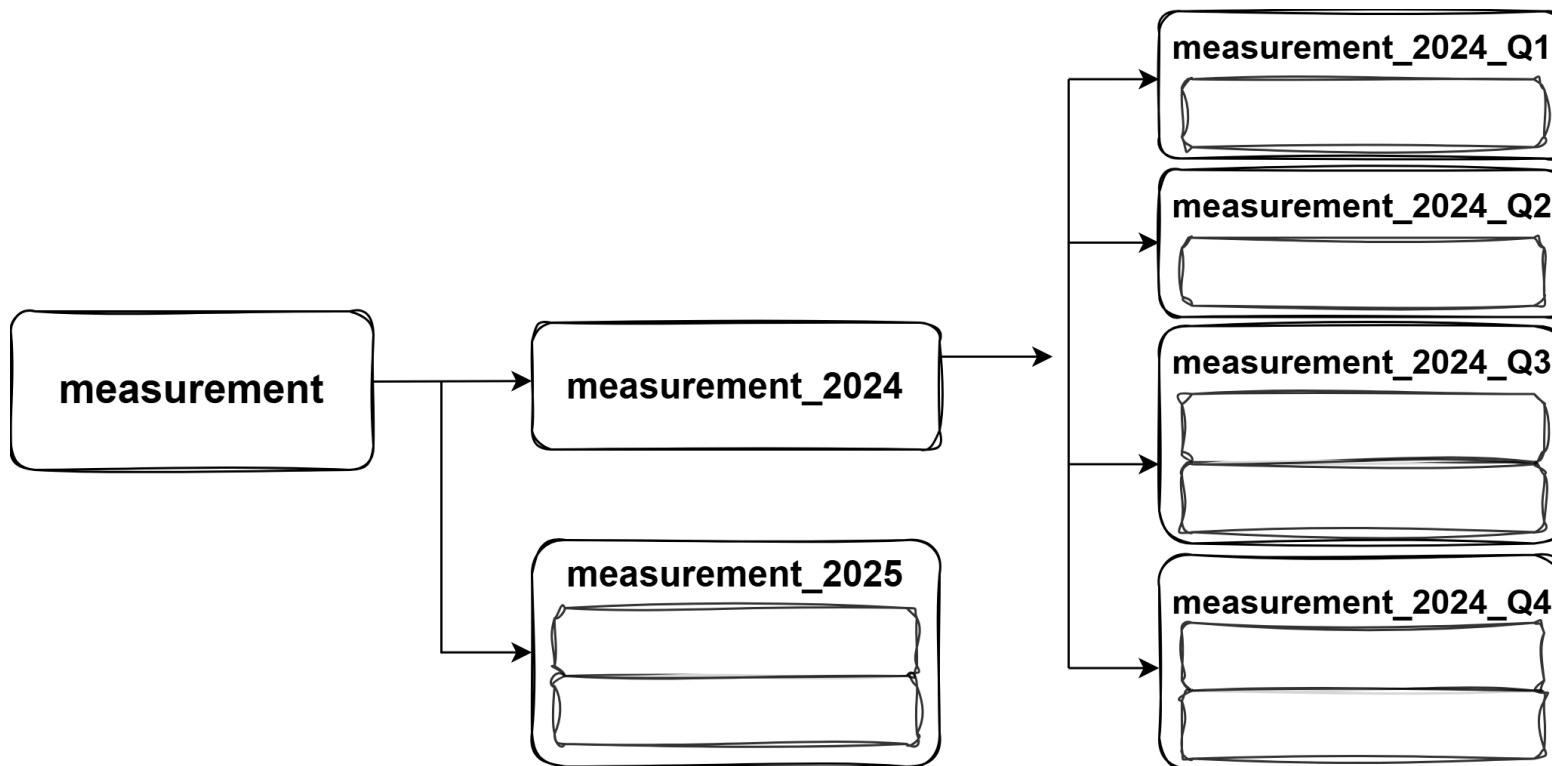
```
CREATE TABLE ... PARTITION OF ... DEFAULT ;
```



Многоуровневое секционирование

Многоуровневое секционирование или subpartitioning подразумевает, что секции также являются секционированными таблицами.

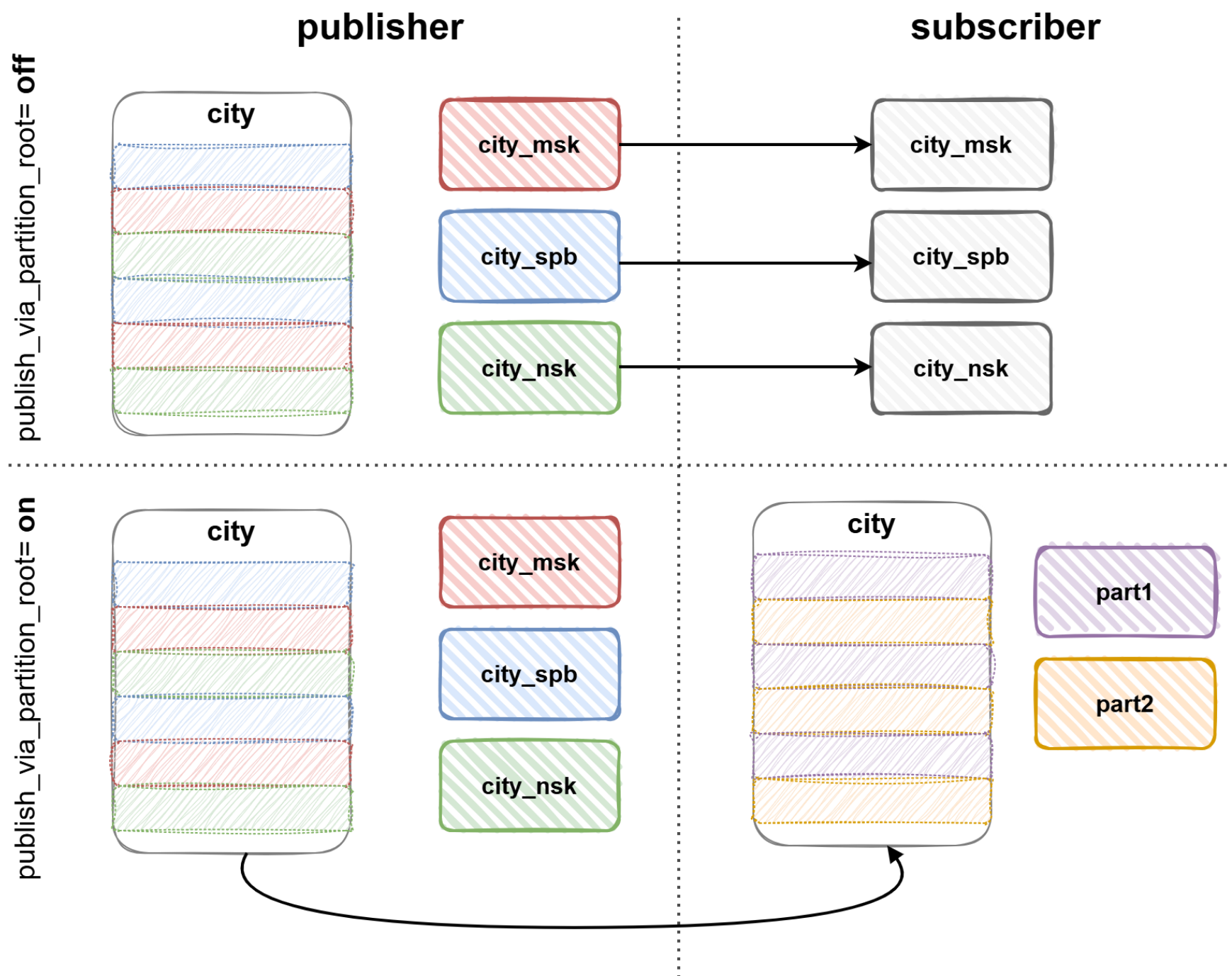
Методы секционирования на разных уровнях могут отличаться.



Логическая репликация

```
CREATE PUBLICATION имя [ WITH ( publish_via_partition_root = on|off [, ... ] ) ]
```

`publish_via_partition_root` определяет будут транслироваться на подписчика от имени самой секционированной таблицы, либо от имени каждой секции.



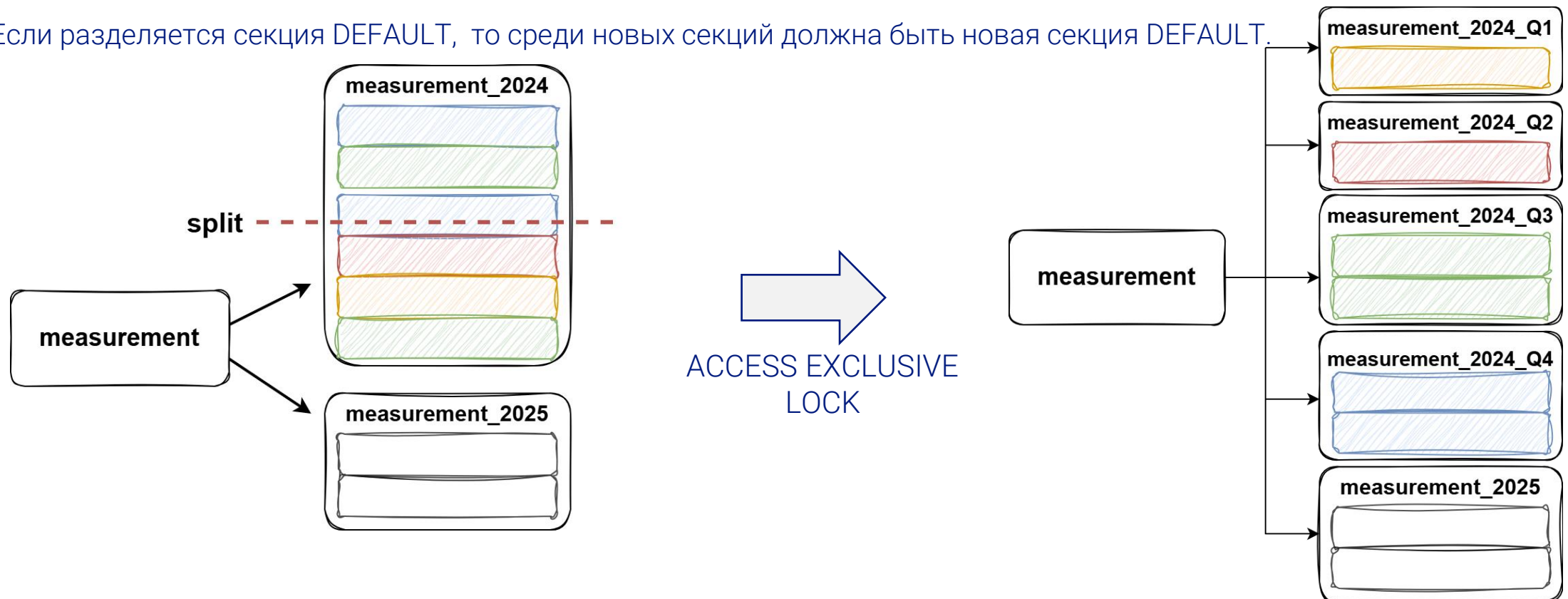
Расширенные методы работы с секциями в Postgres Pro

- SPLIT/MERGE
- Pgpro_autopart
- Pg_pathman
- Глобальные индексы на секционированных таблицах
- Ссылочное секционирование (partition by reference)

Обслуживание секций SPLIT

ALTER TABLE measurement **SPLIT PARTITION** имя_секции INTO (PARTITION имя_секции1 { FOR VALUES указание_границ_секции | DEFAULT }, PARTITION имя_секции2 { FOR VALUES указание_границ_секции | DEFAULT } [, ...])

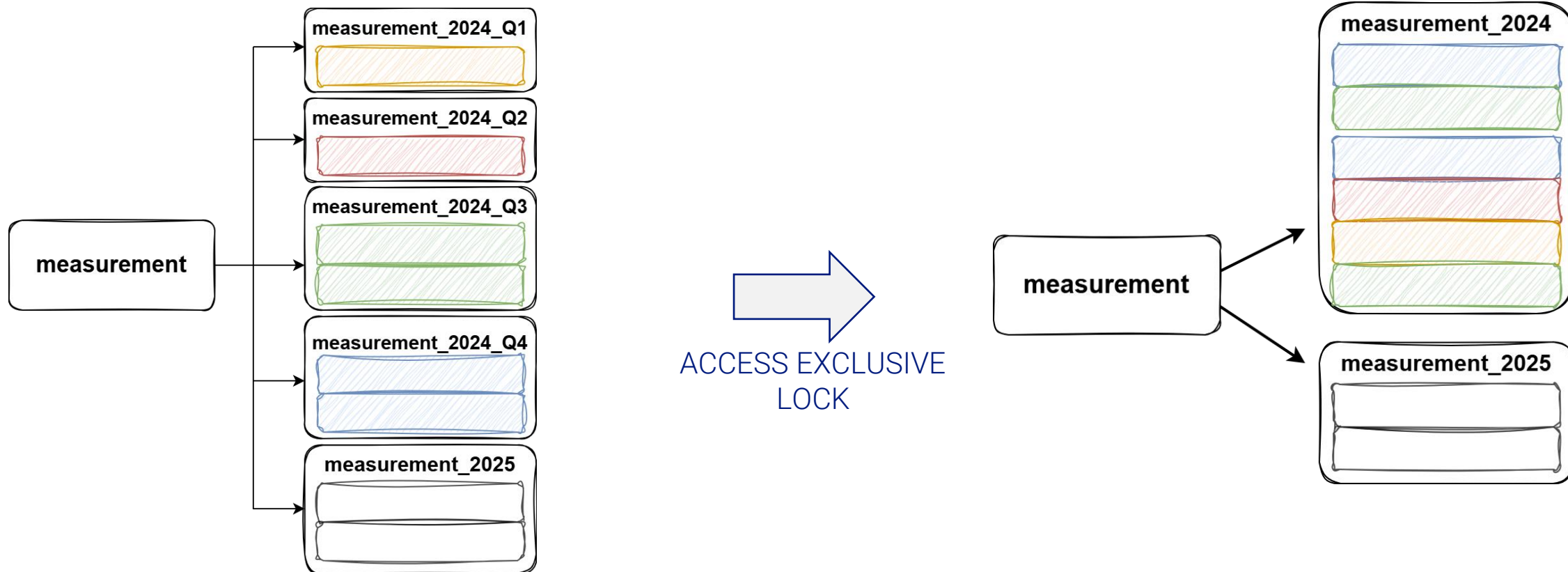
- Если в секционированной таблицы нет секции DEFAULT, то сумма границ всех новых секций должна равняться границе изначальной секции.
- Если разделяется секция DEFAULT, то среди новых секций должна быть новая секция DEFAULT.



Обслуживание секций MERGE

17

ALTER TABLE measurement MERGE PARTITIONS (имя_секции1, имя_секции2 [, ...]) INTO имя_секции



Расширение pg_pathman (old)

Возможности:

- Секционировать большие базы данных, не прерывая их работу за счет неблокирующего (параллельного) секционирования.
- Ускорять выполнение запросов с секционированными таблицами (partition pruning).
- Управлять существующими и добавлять новые секции на лету.
- Добавлять в качестве секций сторонние таблицы.
- Соединять секционированные таблицы для операций чтения и записи.
- Многоуровневое секционирование

`pathman_config` — собственное хранилище конфигурации.

Начиная с 12 версии, не рекомендуется использование. Начиная с 17 версии, расширения pg_pathman больше не будет поддерживаться.

Причина — в нём реализован отличающийся от ванильного механизм секционирования и с каждым годом этот механизм всё труднее поддерживать.

Поэтому к разрабатываемому функционалу было предъявлено требование:

должен поддерживаться ванильный механизм секционирования (возможно, с очень небольшими патчами).

Автоматическое создание секций pgpro_autopart

Расширение pgpro_autopart позволяет создавать секции динамически, то есть производит автоматическое секционирование при добавлении или изменении данных в таблице.

Расширение pgpro_autopart использует для автоматического секционирования функцию `ap_enable_automatic_partition_creation( tablename , interval , first_value )`.

Расширение работает только с новыми таблицами. Его нельзя использовать для существующих секционированных таблиц, поскольку оно не может отслеживать секции, которые были созданы пользователями ранее.

Представление `ap_tables_view` позволяет отслеживать таблицы с включенным автосекционированием.

В разработке (планы на будущее)

1. Глобальные индексы

1Q2025

Глобальный индекс на секционированной таблице представляет собой единый индекс для всех секций данной таблицы, поддерживающий уникальность.

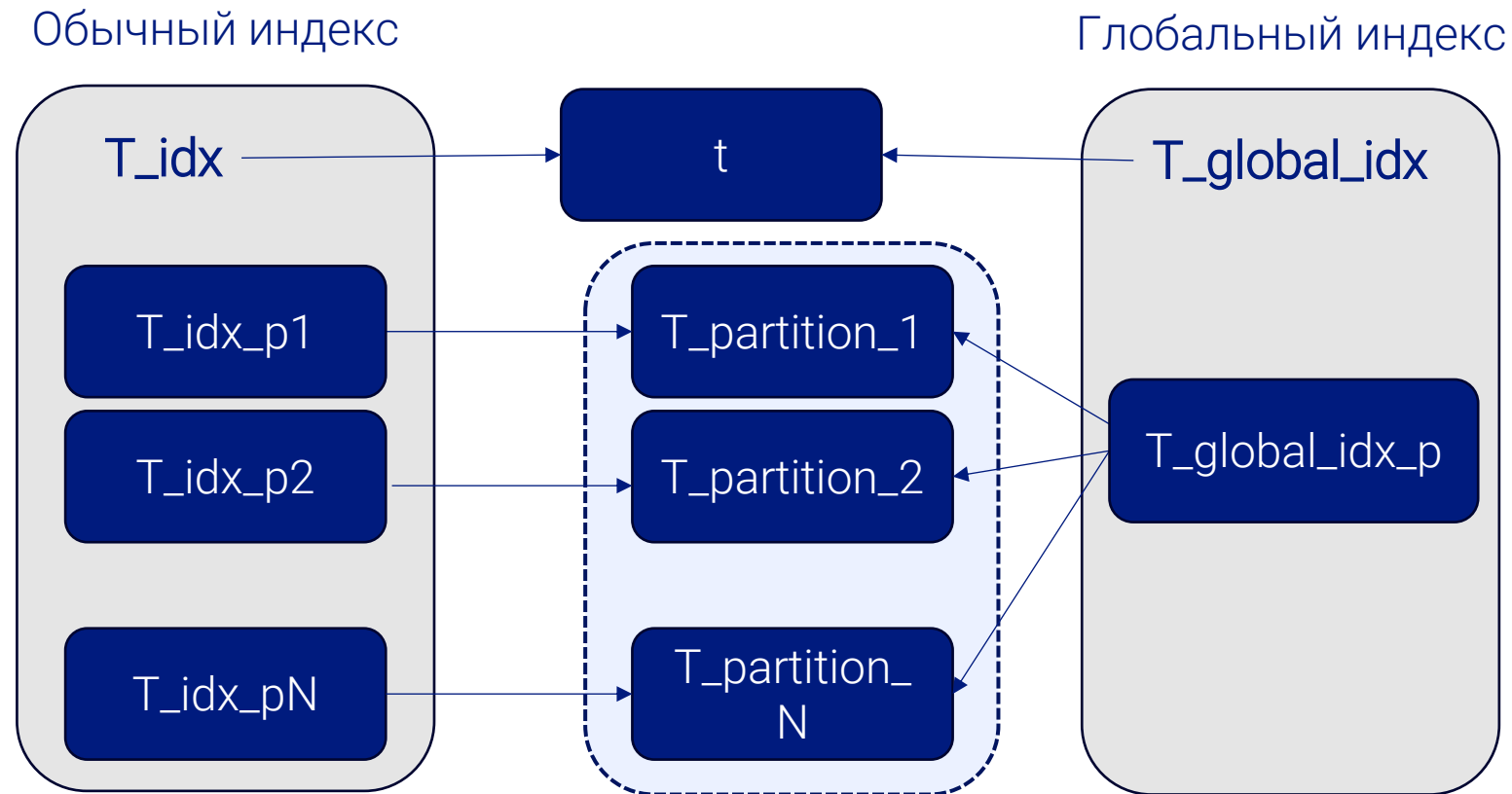
2. Референсное секционирование

1Q2026

Что это такое? — Когда создаваемая таблица партиционируется не явно, а через ссылку на «родительскую» таблицу — не по своему РК, а по FK, с теми же диапазонами, что этот ключ имеет в «родительской» таблице. Таким образом, партии «родительской» и «дочерней» таблицы получаются связанными, и при планировании запроса, определив одну партицию — мы можем однозначно определить связанную с ней, без сканирования иерархии.

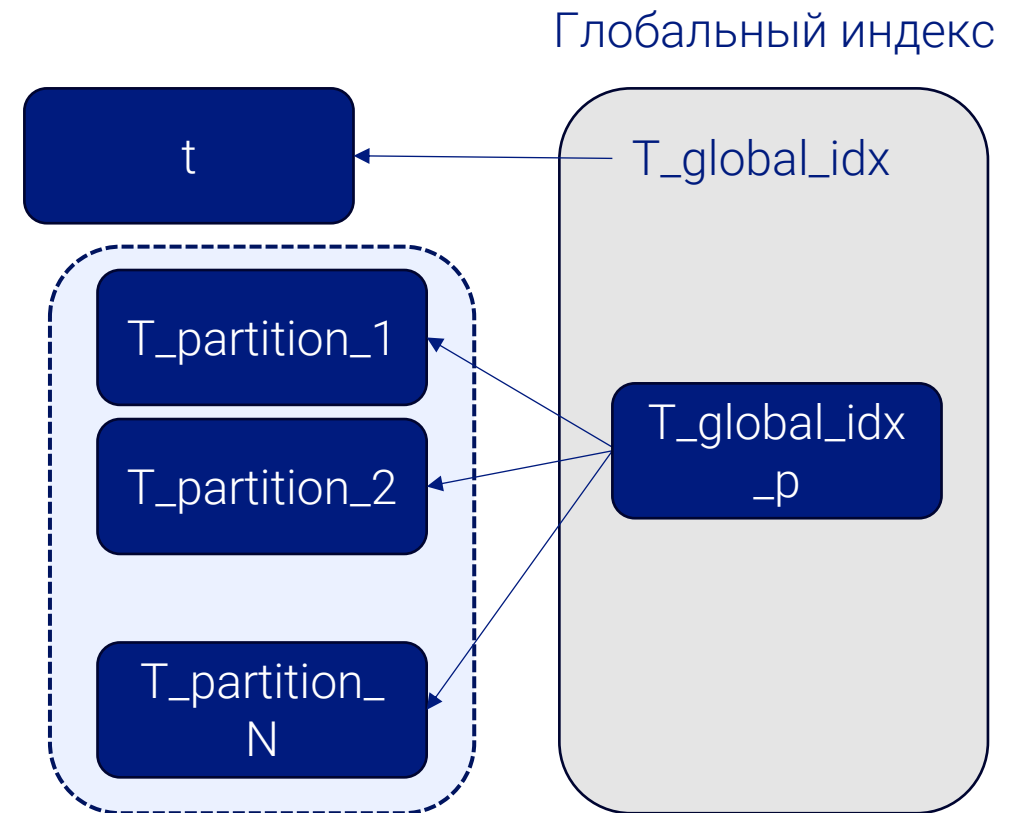
Глобальные индексы

Глобальный индекс на секционированной таблице представляет собой единый индекс для всех секций данной таблицы, поддерживающий уникальность



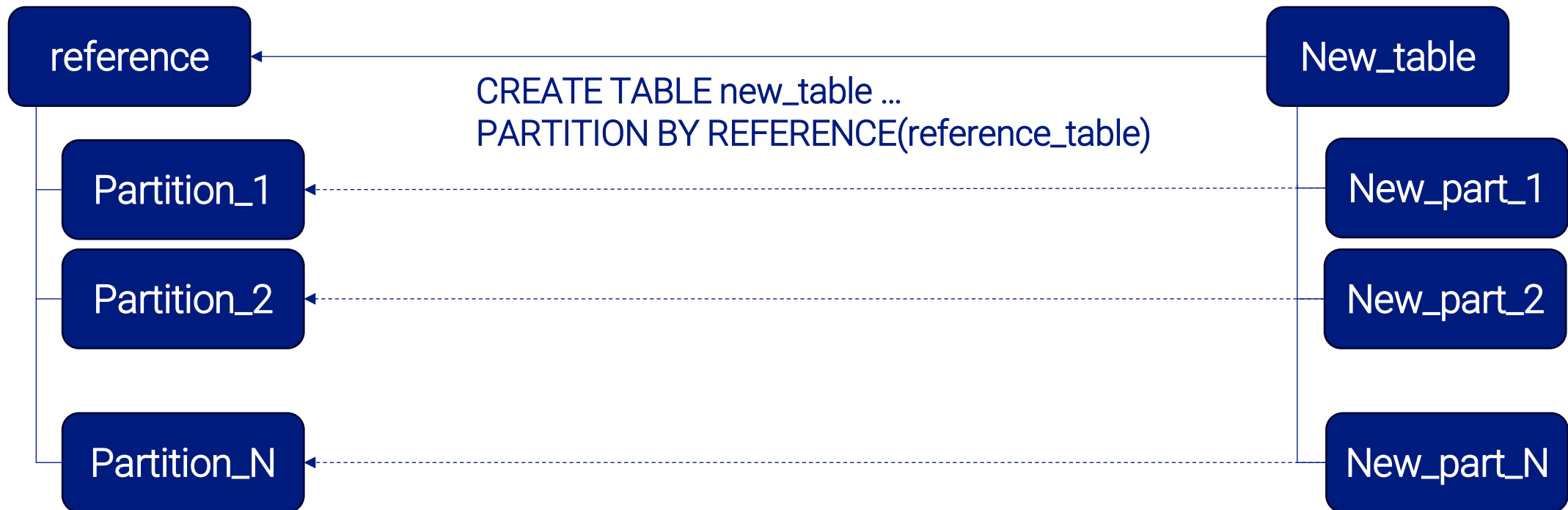
Глобальные индексы

- + Уникальность не только по РК
- + Скорость поиска при большом количестве секций
- Сложность построения
- Объем занимаемого дискового пространства
- Оверхед при присоединении секции в иерархию
- Вакуум



Ссылочное секционирование

Ссылочное секционирование – механизм, позволяющий создавать новые секционированные таблицы без явного указания типа и ключа секционирования, ссылаясь на существующую секционированную таблицу (референс). При этом стратегия и диапазоны секционирования определяются автоматически, ключом секционирования становится FK (foreign key), ссылающийся на референс-таблицу, и секции также создаются и отсоединяются автоматически.



Ссылочное секционирование (пример)

Таблица ORDERS

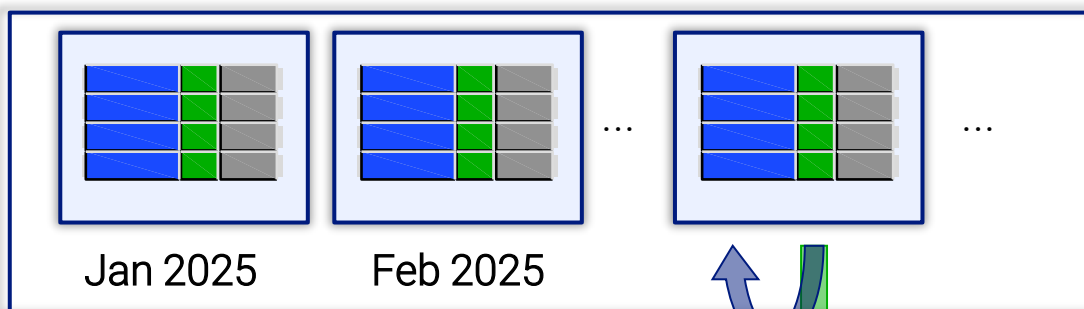
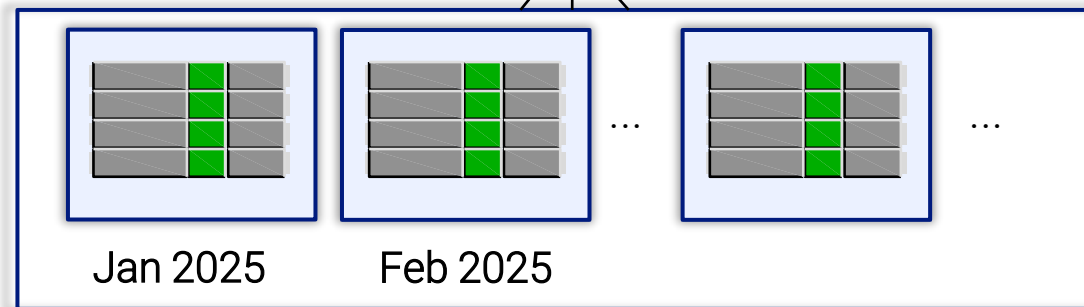


Таблица ORDER_ITEMS



PARTITION BY REFERENCE
Ключ секционирования
наследуется через связь РК-FK

RANGE(`order_date`)
Foreign key `order_id`

Ссылочное секционирование

При создании новой секции референс-таблицы автоматически создаются соответствующие секции в ссылающихся, при отсоединении или удалении также автоматически отсоединяются или удаляются соответствующие секции ссылающихся таблиц



Полезные ссылки

- Секционирование. Глава из учебного курса.
- https://edu.postgrespro.ru/dba2-9.5/dba2_21_partitioning.pdf
- Встроенное партицирование в постгресе и сторонние решения. Доклад pgconf.russia
- <https://pgconf.ru/talk/1588235>

Appendix I – схема работы pgpro_autopart

Схема работы pgpro_autopart

1 CREATE TABLE measurement (id bigint, s text) PARTITION BY RANGE (id);

нужно создавать новую таблицу, т.к. автосекционирование не работает для существующих секционированных таблиц

measurement

Схема работы pgpro_autopart

- 1 CREATE TABLE measurement (id bigint, s text) PARTITION BY RANGE (id);
- 2 SELECT ap_enable_automatic_partition_creation('measurement', 10, 100);

10 – интервал
 100 – начальное значение
 Представление и триггер создаются автоматически

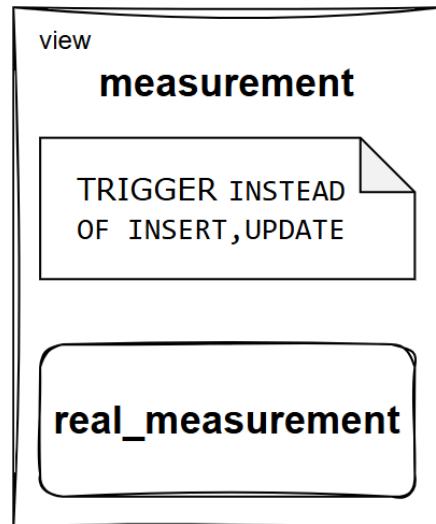


Схема работы pgpro_autopart

17

- 1 CREATE TABLE measurement (id bigint, s text) PARTITION BY RANGE (id);
 - 2 SELECT ap_enable_automatic_partition_creation('measurement', 10, 100);
 - 3 INSERT INTO measurement VALUES (111, 'text');
- NOTICE: New partition "public"."real_measurement_110_120" created

Уровень получаемых с сервера уведомлений можно настраивать в сессионном параметре client_min_messages

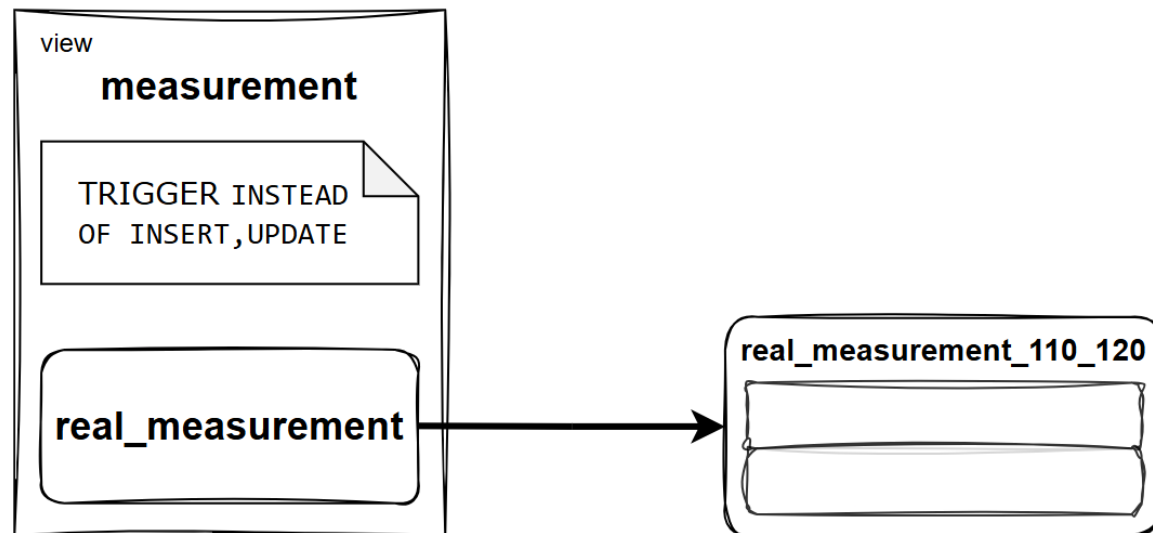


Схема работы pgpro_autopart

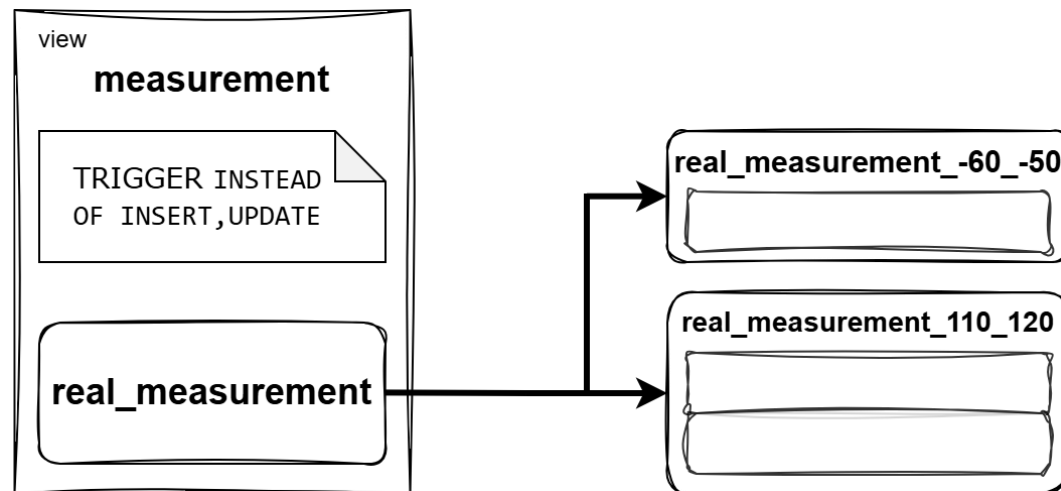
- 1 CREATE TABLE measurement (id bigint, s text) PARTITION BY RANGE (id);
- 2 SELECT ap_enable_automatic_partition_creation('measurement', 10, 100);
- 3 INSERT INTO measurement VALUES (111, 'text');

NOTICE: New partition "public"."real_measurement_110_120" created

- 4 UPDATE measurement SET id = -55 WHERE id = 111 RETURNING *;
- NOTICE: New partition "public"."real_measurement_-60_-50" created

Триггер на представление предусматривает команды с «RETURNING *»

```
id | s
----+---
-55 | text
(1 row)
UPDATE 1
```



Appendix II –секционирование существующей таблицы

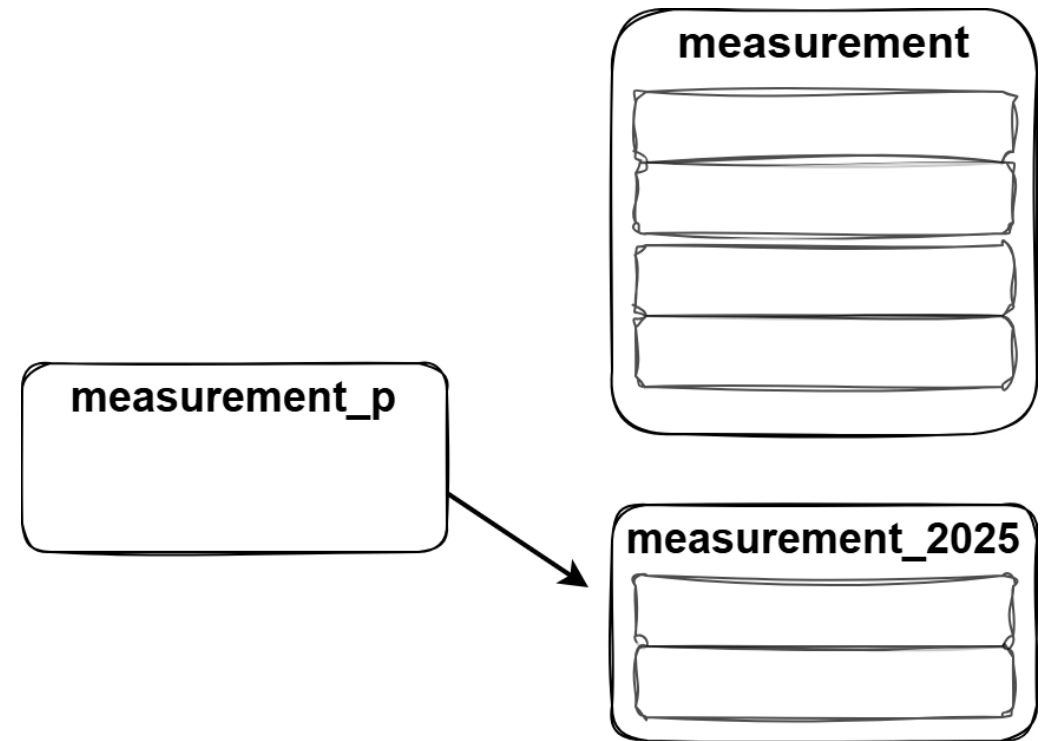
Секционирование существующей таблицы

Чтобы не устраивать длительный даунтайм сервиса, немного подготовим таблицу с данными:

```
begin;
set local statement_timeout to '1s';
alter table measurements add constraint
measurements_partbound_check check (logdate < '2024-01-01' and
created_at is not null) not valid;
commit;
```

```
alter table measurements validate constraint
measurements_partbound_check ;
```

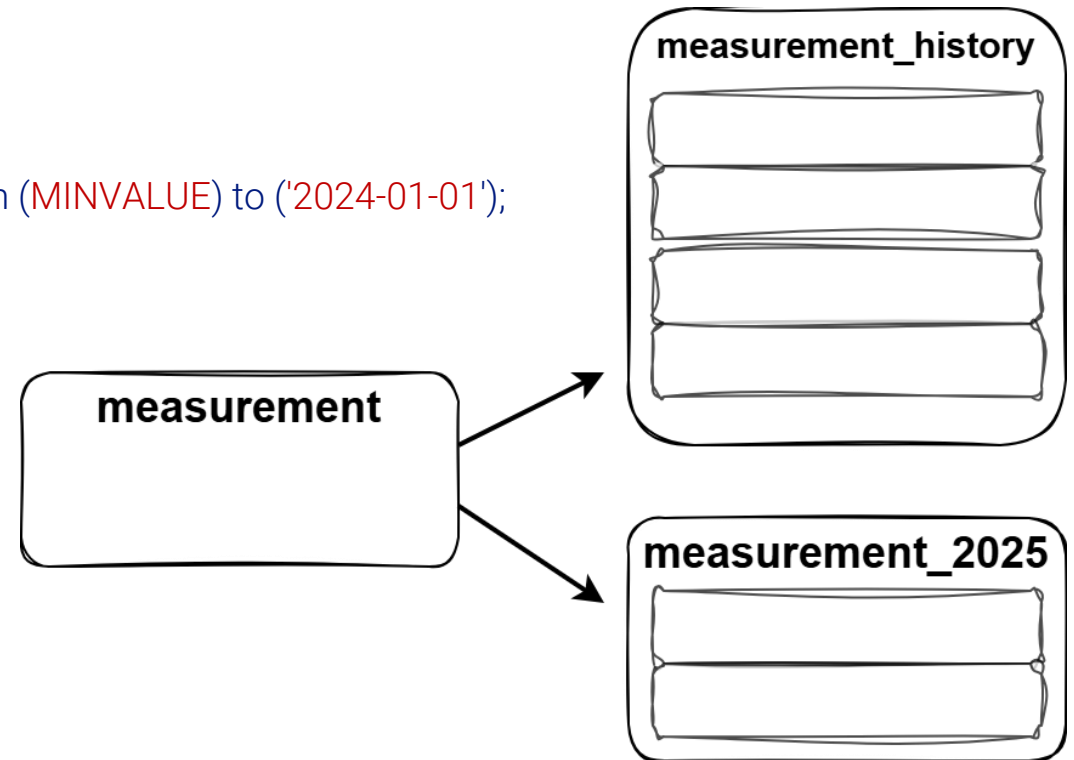
check
logdate < '2024-01-01'



Секционирование существующей таблицы

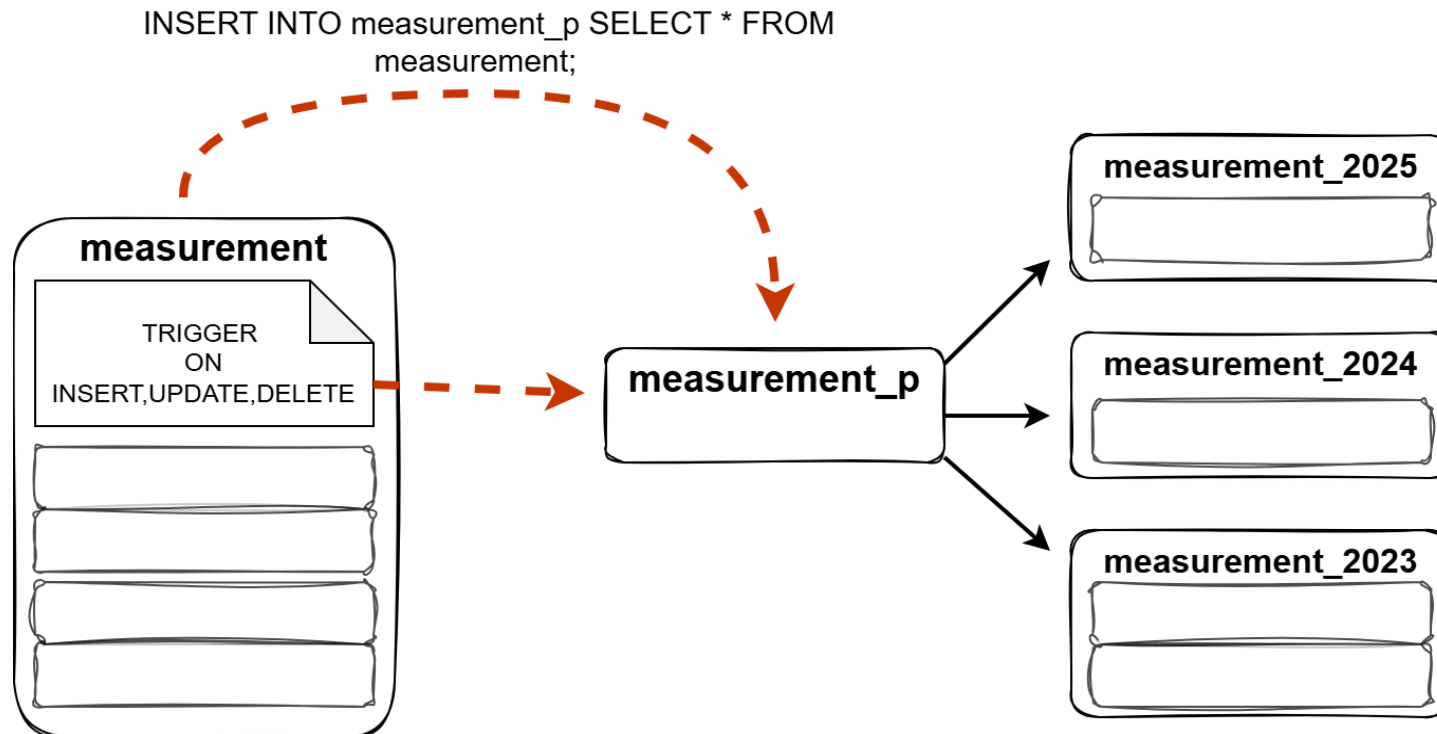
```
begin;
set statement_timeout to '1s';
alter table measurements rename to measurements_history;
alter table measurements_p rename to measurements;
alter table measurements attach partition measurements_archive for values from (MINVALUE) to ('2024-01-01');
commit;
```

Теперь таблица **measurements** является партицированной за две блокировки длительностью менее секунды.
Данные до 2025-01-01 продолжат писаться фактически в старую таблицу, затем начнут писаться в новую.



Честное секционирование существующей таблицы

Если требуется именно «разрезать» существующую таблицу, то это тоже возможно. Для этого потребуется создать рядом новую секционированную таблицу, с отличным именем. И подготовить необходимое количество секций под данные. Скопировать туда данные (`insert .. select ...`, либо `copy`) и каким-либо образом проконтролировать, что вы меняете данные в старой большой таблице и в новых разделах одновременно.





Спасибо
за внимание!

