1. Создайте произвольную таблицу (~ мастер таблица).

CREATE TABLE master\_table(

id SERIAL PRIMARY KEY,

data VARCHAR(100),

created\_at TIMESTAMP NOT NULL

);

1. Создайте триггер с типом before INSERT который автоматически создает наследуемую таблицу (inherited table) на основании вашего выбранного ключа разделения данных (например по дате / месяцу / году), используя динамический SQL.

CREATE FUNCTION insert\_into\_partition()

RETURNS TRIGGER

AS $$

DECLARE

slave\_table TEXT;

year\_month TEXT;

BEGIN

year\_month := to\_char(NEW.created\_at, 'YYYY\_MM');

slave\_table := 'partition\_' || year\_month;

IF NOT EXISTS (

SELECT 1

FROM information\_schema.tables

WHERE table\_name = slave\_table

) THEN

EXECUTE format('

CREATE TABLE %I (

CHECK (to\_char(created\_at, ''YYYY\_MM'') = %L)

) INHERITS (master\_table);',

slave\_table, year\_month

);

END IF;

EXECUTE format('INSERT INTO %I (created\_at, data) VALUES ($1, $2)', slave\_table)

USING NEW.created\_at, NEW.data;

RETURN NULL;

END;

$$ LANGUAGE plpgsql;

CREATE TRIGGER master\_insert\_trigger

BEFORE INSERT ON master\_table

FOR EACH ROW

EXECUTE FUNCTION insert\_into\_partition();

1. Сгенерируйте 1 000 000 строк в мастер таблицу.

INSERT INTO master\_table (data, created\_at)

SELECT

'Sample data ' || gs.i AS data,

timestamp '2025-05-04' + (random() \* interval '365 days') AS created\_at

FROM generate\_series(1, 1000000) AS gs(i);

1. График распределения данных между всеми наследуемыми таблицами после генерации 1 000 000 строк.

Список всех наследуемых таблиц.

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание