Задание №5: Шардирование

# Настройка

Для шардирования данных переписки пользователей (/dialog/{user\_id}/send и /dialog/{user\_id}/list) использовалось расширение [Citus](https://www.citusdata.com/) для PostgreSQL.

Для этого был развернут дополнительный набор контейнеров:

* citus\_db\_master
* citus\_worker
* citus\_manager

В соответствии с официальным [руководством](https://github.com/citusdata/docker/blob/master/README.md).

Методы монолита были переключены на работы с новым набором контейнеров.

# Создание таблиц

Была создана таблица User\_Dialog для хранения данных переписки:

CREATE TABLE "User\_Dialog" (   
  "Id"             SERIAL NOT NULL,   
  "From\_UserId"    INTEGER NOT NULL,   
  "To\_UserId"      INTEGER NOT NULL,   
  "Text"           TEXT NOT NULL,   
  "CreatedAt"      TIMESTAMP NOT NULL,   
  
  CONSTRAINT "PK\_User\_Dialog\_Id\_From\_UserId\_To\_UserId" PRIMARY KEY ("Id", "From\_UserId", "To\_UserId")   
);

Подключено расширение для БД:

SELECT create\_distributed\_table('"User\_Dialog"', 'From\_UserId');

Ключом шардирования выбран идентификатор отправителя, на случай, если некоторые пользователи будут писать гораздо активнее остальных.

Проверка работы шардирования:

otus\_social\_network=> EXPLAIN SELECT \* FROM "User\_Dialog";   
                                            QUERY PLAN                                                
---------------------------------------------------------------------------------------------------   
 Custom Scan (Citus Adaptive)  (cost=0.00..0.00 rows=100000 width=52)   
   Task Count: 32   
   Tasks Shown: One of 32   
   ->  Task   
         Node: host=localhost port=5432 dbname=otus\_social\_network   
         ->  Seq Scan on "User\_Dialog\_102008" "User\_Dialog"  (cost=0.00..20.20 rows=1020 width=52)   
(6 rows)

Для масштабирования кластера можно использовать команду:

docker-compose -p citus up --scale citus\_worker=5

Которая добавит дополнительные узлы воркеров и подключит их к мастер-ноде.

Чтобы перераспределить данные по воркерам, нужно использовать команду:

SELECT rebalance\_table\_shards();

Для того чтобы подключить шардирование к существующей таблицу, не прерывая работу приложения, можно использовать команду:

SELECT create\_distributed\_table\_concurrently

('"User\_Dialog"', 'From\_UserId');