**Доп по 3 лабораторной РСХД.**

**Задание**: реализовать потоковую репликацию между основным и резервным узлом.

**Выполнение**:

**Настройка master-сервера**

Создаём новый кластер: initdb --username=postgres6 -D $HOME/DOP/ifg51/

Вносим изменения в postgresql.conf:  
listen\_addresses = '\*' # what IP address(es) to listen on;

port = 9455 # (change requires restart)

Запускаем сервер: pg\_ctl -D /var/db/postgres6/DOP/ifg51/ -l logfile start

Подключаемся к БД: psql --port=9455 postgres

Создаем пользователя с правами на репликацию: create user repuser replication;

Меняем конфиг подключения:

host replication repuser all trust

Перезапускаем сервер: pg\_ctl -D /var/db/postgres6/DOP/ifg51/ -l logfile restart

**Настройка доп. сервера (slave)**

Копируем структуру на доп сервер: pg\_basebackup -h pg157.cs.ifmo.ru --port=9455 -D $HOME/DOP/ifg51/ -P -U repuser --wal-method=stream

Далее вносим изменения в файл postgresql.conf:

hot\_standby = on

primary\_conninfo = 'user=repuser host=pg157.cs.ifmo.ru port=9455 sslmode=prefer sslcompression=1 krbsrvname=postgres' # connection string to sending server

Чтобы репликация заработала, также в каталоге с данными нужно создать файл standby.signal.

touch standby.signal

Запускаем сервер: pg\_ctl -D /var/db/postgres1/DOP/ifg51/ -l logfile start

**Проверка:**

Создадим на основном сервере базу AmazingReplication, а в ней таблицу "TestReplication".

CREATE TABLE "TestReplication" ("SomveValue" varchar(100));

INSERT INTO "TestReplication" VALUES ('test 1');

INSERT INTO "TestReplication" VALUES ('test 2');

INSERT INTO "TestReplication" VALUES ('test 3');

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Практически сразу на дополнительном сервере появится эта база данных и можно выполнить запрос.

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание