



Алтайский государственный технический университет им. И.И. Ползунова  
Факультет информационных технологий  
Кафедра прикладной математики

# Разработка системы инвалидации кеша для PostgreSQL

Студент: Ткаченко Алексей, ПИ-21

**Научный руководитель: Целебровский О.Б.,  
младший разработчик PostgresPro**

# Цель работы

Разработка программного обеспечения, отслеживающего изменения в PostgreSQL и оперативно удаляющее неактуальные ключи из кеша на Redis.

## Задачи

- 1) Спроектировать модульную архитектуру системы
- 2) Настроить подключение сервиса к PostgreSQL и Redis
- 3) Разработать механизм чтения журнала БД, преобразование записей из журнала в ключи для Redis и их удаление
- 4) Реализовать гибкий механизмы настройки параметров системы
- 5) Написать комплексные интеграционные тесты и проверить работу системы

# Актуальность

Существующие подходы к инвалидации и их недостатки:

<b>Инвалидация на стороне приложения</b>	<b>TTL инвалидация</b>
Совмещение бизнес-логики и логики инвалидации	Возможность выдачи устаревших данных
Необходимость дублирования логики инвалидации в каждом приложении	Резкий всплеск нагрузки на БД

# Change Data Capture (CDC)

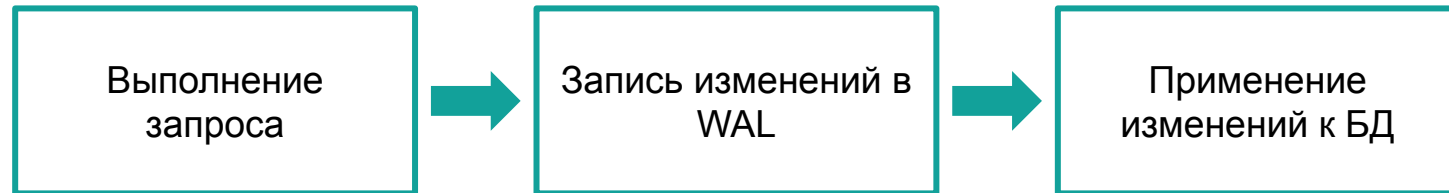
Создание отдельного сервиса для инвалидации кеша позволит:

- отделить логику инвалидации от бизнес-логики
- осуществить централизованную инвалидацию, независимую от приложений, использующих кеш и БД
- производить инвалидацию только при реальных изменениях в БД

# Журнал предзаписи

Write Ahead Logging (WAL) - это стандартный метод обеспечения целостности данных.

## Работа СУБД и WAL



Стек технологий: БД



# Стек технологий: инструменты



**Golang** - язык программирования

jackc/pgx

PostgreSQL driver and toolkit for Go



**pgx** - работа с PostgreSQL

jackc/pglogrepl

PostgreSQL logical replication library for Go.



**pglogrepl** - работа с логической репликацией

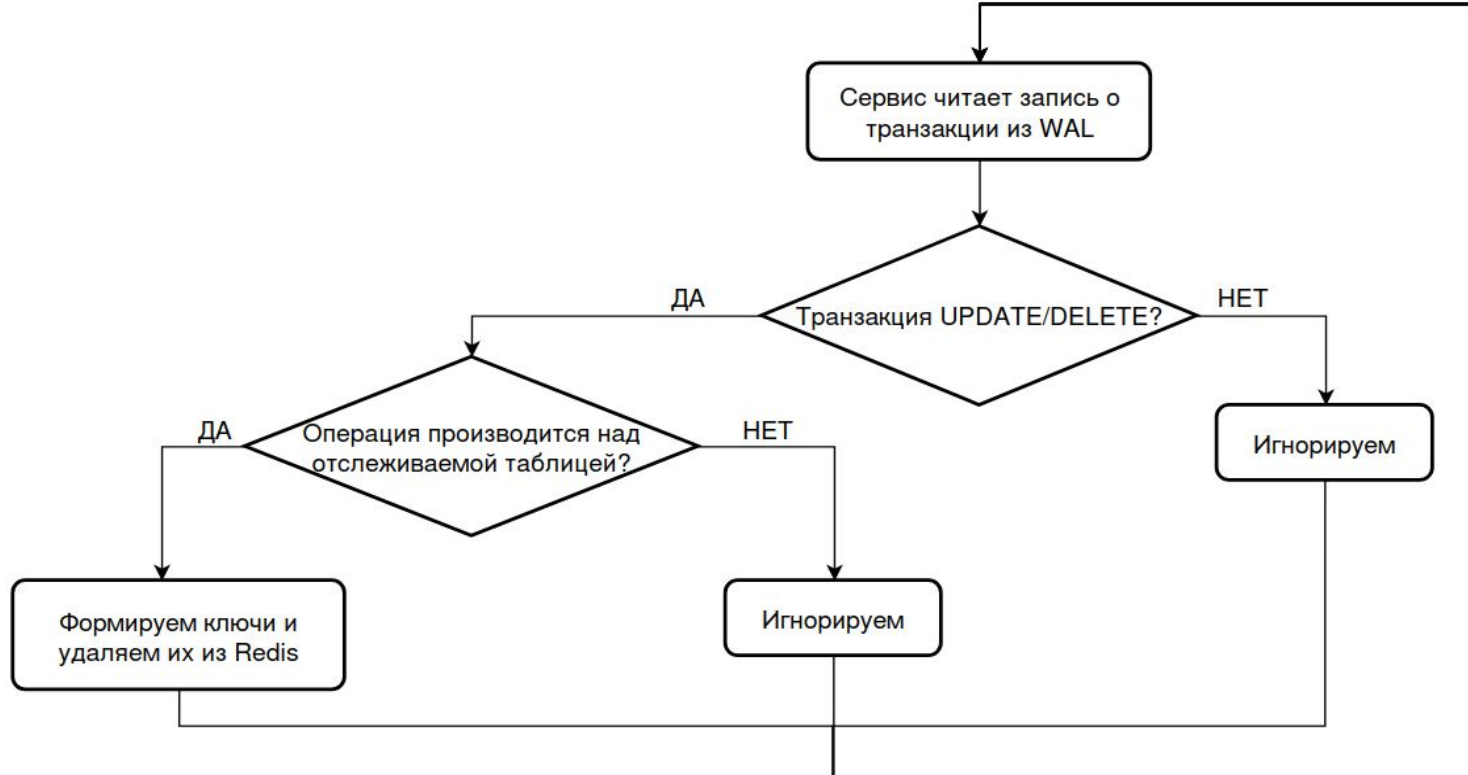
redis/go-redis

Redis Go client



**go-redis** - работа с Redis

# Алгоритм работы



# Настройка сервиса

Настройка правил отображения таблиц на ключи Redis (правил маппинга) производится через конфигурационный **.yaml** файл.

В конфигурационном файле:

```
mapping:
  rules:
    - table: user
      key_pattern: "user:{id}"
```

В базе данных:

user	
● id	🔑 INTEGER
● name	? VARCHAR(127)
● age	? SMALLINT

# Тестирование

## Unit-тесты:

- Корректная генерация ключей
- Проверка конфигурации
- Параметры по-умолчанию

## Интеграционные тесты:

- Инвалидация записей
- Корректное завершение работы
- Отмена транзакции
- Отсутствие ключа в кеше



Тестовое окружение для ручного тестирования

# Описание реализованного сервиса

- отдельный сервис с возможностью запуска на хосте и в docker контейнере
- производит инвалидацию без внесения изменений в бизнес логику
- настраивается через .yaml файл
- гибкая настройка маппинга
- покрытие unit и интергационными тестами

# Перспективы развития

- Поддержка других СУБД и систем кеширования - позволит расширить сферу применения проекта
- Изменение параметров сервиса без перезапуска - позволит наиболее оперативно реагировать на изменения архитектуры кеша
- WEB интерфейс - позволит увеличить удобство настройки сервиса

# Результаты работы

- выбран оптимальный технологический стек
- спроектирована модульная архитектура
- реализована инвалидация кеша Redis при изменении и/или удалении записей в определенных таблицах PostgreSQL
- написаны модульные и интеграционные тесты
- создан docker контейнер сервиса



Алтайский государственный технический университет им. И.И. Ползунова  
Факультет информационных технологий  
Кафедра прикладной математики

# Разработка системы инвалидации кеша для PostgreSQL

Студент: Ткаченко Алексей, ПИ-21

**Научный руководитель: Целебровский О.Б.,  
младший разработчик PostgresPro**