



КАК МЫ РАСПРОЩАЛИСЬ С MONGODB И ПЕРЕШЛИ НА POSTGRESQL

Литуненко Андрей



О СЕБЕ



1. Работаю в компании два года

2. Бэкенд-разработчик



**О ЧЕМ
РАССКАЖУ?**



1. Про 2ГИС и Личный кабинет

2. Боль при использовании двух СУБД

3. Решение

4. Советы



300+

городов

40+ млн.

пользователей

3000+

сотрудников



ВОЗМОЖНОСТИ ЛИЧНОГО КАБИНЕТА



1. Управление информацией о компании

2. Управление рекламой

3. Управление медиа-контентом

4. Просмотр статистики



АРХИТЕКТУРА ЛИЧНОГО КАБИНЕТА



1. PHP

2. PostgreSQL, MongoDB



ПРОБЛЕМЫ



1. Сложные отчёты



1. Сложные отчёты

2. Мерджинг данных на клиенте



1. Сложные отчёты
2. Мерджинг данных на клиенте
3. Допотопная версия MongoDB



mongoDB



РЕШЕНИЕ

В рамках одной задачи переносим одну коллекцию



1. Документы с плоской структурой
2. Документы с вложенным массивом
3. Документы с разнородной структурой



ДОКУМЕНТЫ С ПЛОСКОЙ СТРУКТУРОЙ

Запросы на восстановление пароля



<https://api.auth.2gis.com/reset/JJf349xdfa3>



<https://api.auth.2gis.com/reset/JJf349xdfa3>

```
{  
  "userEmail": "test@test.ru",  
  "code" : "JJf349xdfa3",  
  "dateCreated": ISODate("2017.12.09 09:21:34"),  
  "dateExpired": ISODate("2017.12.12 09:21:34")  
}
```



ПЕРЕНОС ДАННЫХ



1. <https://github.com/stripe/mosql>



1. <https://github.com/stripe/mosql>

2. Простой запуск



1. <https://github.com/stripe/mosql>

2. Простой запуск

3. 2 млн. документов за 4 минуты



1. <https://github.com/stripe/mosql>

2. Простой запуск

3. 2 млн. документов за 4 минуты

4. Смена первичного ключа



```
ALTER TABLE user_recover_possword DROP CONSTRAINT id_pkey;
```

```
ALTER TABLE user_recover_possword ALTER column _id DROP NOT NULL;
```

```
ALTER TABLE user_recover_password ADD PRIMARY KEY(code);
```



ПОДДЕРЖАНИЕ ИЗМЕНЕНИЙ



1. Замена источника данных

2. Тестирование



РЕЛИЗ ЗАДАЧИ

Оценили время, когда минимальная нагрузка на коллекцию





ВЫВОДЫ ПРО КОЛЛЕКЦИИ С ПЛОСКОЙ СТРУКТУРОЙ



1. mosql нам подошёл



1. mosql нам подошёл

2. Перенос документирован по шагам



1. mosql нам подошёл

2. Перенос документирован по шагам

3. Обошлись без даунтайма



1. mosql нам подошёл

2. Перенос документирован по шагам

3. Обошлись без даунтайма

4. Целостность данных



ДОКУМЕНТЫ С ВЛОЖЕННЫМ МАССИВОМ

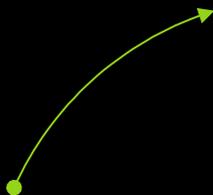
Пользователи

```
{  
  "email": "test@test.ru",  
  "role": "client",  
  "status": "activated",  
  "first_name": "Test",  
  "last_name": "Test",  
  "firmLinks" : [  
    {  
      "firmId": 1,  
      "status": "approved"  
    }  
  ]  
}
```

Пользователи



Запросы на восстановление пароля





1. Декомпозируем на подзадачи



1. Декомпозируем на подзадачи

2. Рефакторим



1. Декомпозируем на подзадачи

2. Рефакторим

3. Тестируем



1. Декомпозируем на подзадачи

2. Рефакторим

3. Тестируем

4. Выпускаем на бой

Код без рефакторинга 

```
public function actionSave($firstName)
{
    $this->user->setFirstName($firstName);
    $this->mongoDb->persist($this->user);
}
```

Код после рефакторинга 

```
public function actionSave($firstName)
{
    $this->user->setFirstName($firstName);
    UserComponent::create()
        ->setUser($this->user)
        ->save();
}
```



ПЕРЕНОС ДАННЫХ



1. 1.5 млн. документов за 15 минут



1. 1.5 млн. документов за 15 минут

2. Смена первичного ключа



1. 1.5 млн. документов за 15 минут

2. Смена первичного ключа

3. Перенос вложенных фирм в отдельную таблицу за 20 минут



```
BEGIN
```

```
  i := 0;
```

```
  sql_query := '';
```

```
  FOR row_user IN SELECT * FROM users
```



```
BEGIN
  i := 0;
  sql_query := '';
  FOR row_user IN SELECT * FROM users
  LOOP
    FOR firm_link IN SELECT * FROM jsonb_array_elements(row_user.firm_links)
```



```
BEGIN
  i := 0;
  sql_query := '';
  FOR row_user IN SELECT * FROM users
  LOOP
    FOR firm_link IN SELECT * FROM jsonb_array_elements(row_user.firm_links)
      i := i + 1;
      firm_id := (firm_link->>'firmId')::bigint;
      user_firm_status := (firm_link->>'status')::character varying(256);
```



```
BEGIN
  i := 0;
  sql_query := '';
  FOR row_user IN SELECT * FROM users
  LOOP
    FOR firm_link IN SELECT * FROM jsonb_array_elements(row_user.firm_links)
      i := i + 1;
      firm_id := (firm_link->>'firmId')::bigint;
      user_firm_status := (firm_link->>'status')::character varying(256);
      sql_query := sql_query || '(' || row_user.id || ', ' || firm_id || ', ' ||
quote_literal(user_firm_status) || '), ';
```



```
IF i > 1000 THEN
```

```
    sql_query := rtrim(sql_query, ', ');
```

```
    i := 0;
```

```
    EXECUTE ('INSERT INTO user_firm VALUES' || sql_query);
```

```
    sql_query := '';
```

```
END IF;
```



РЕЛИЗ





1. Выключаем проект, предупредив коллег



1. Выключаем проект, предупредив коллег
2. Показываем заглушку



1. Выключаем проект, предупредив коллег
2. Показываем заглушку
3. Переносим данные



1. Выключаем проект, предупредив коллег
2. Показываем заглушку
3. Переносим данные
4. Релизим код



1. Выключаем проект, предупредив коллег
2. Показываем заглушку
3. Переносим данные
4. Релизим код
5. Проверяем целостность данных



1. Выключаем проект, предупредив коллег
2. Показываем заглушку
3. Переносим данные
4. Релизим код
5. Проверяем целостность данных
6. Включаем проект



ВЫВОДЫ ПРО КОЛЛЕКЦИИ С ВЛОЖЕННЫМИ МАССИВАМИ

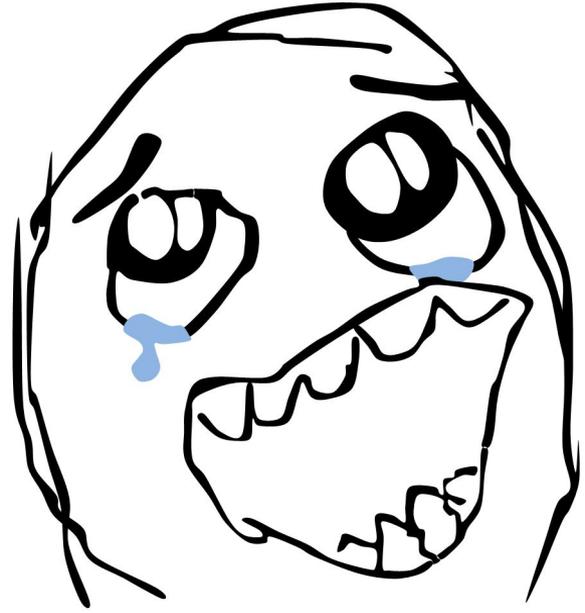


1. Переезд занял две недели



1. Переезд занял две недели
2. Можем получить данные одним запросом

```
SELECT u.* FROM users AS u
INNER JOIN user_firm AS uf ON uf.user_id = u.id
INNER JOIN firm AS f ON f.id = uf.firm_id
INNER JOIN project AS p ON p.id = f.project_id
WHERE u.status = 'activated'
      AND u.last_visit_date > '2017-01-01'
      AND p.name = 'Новосибирск'
```



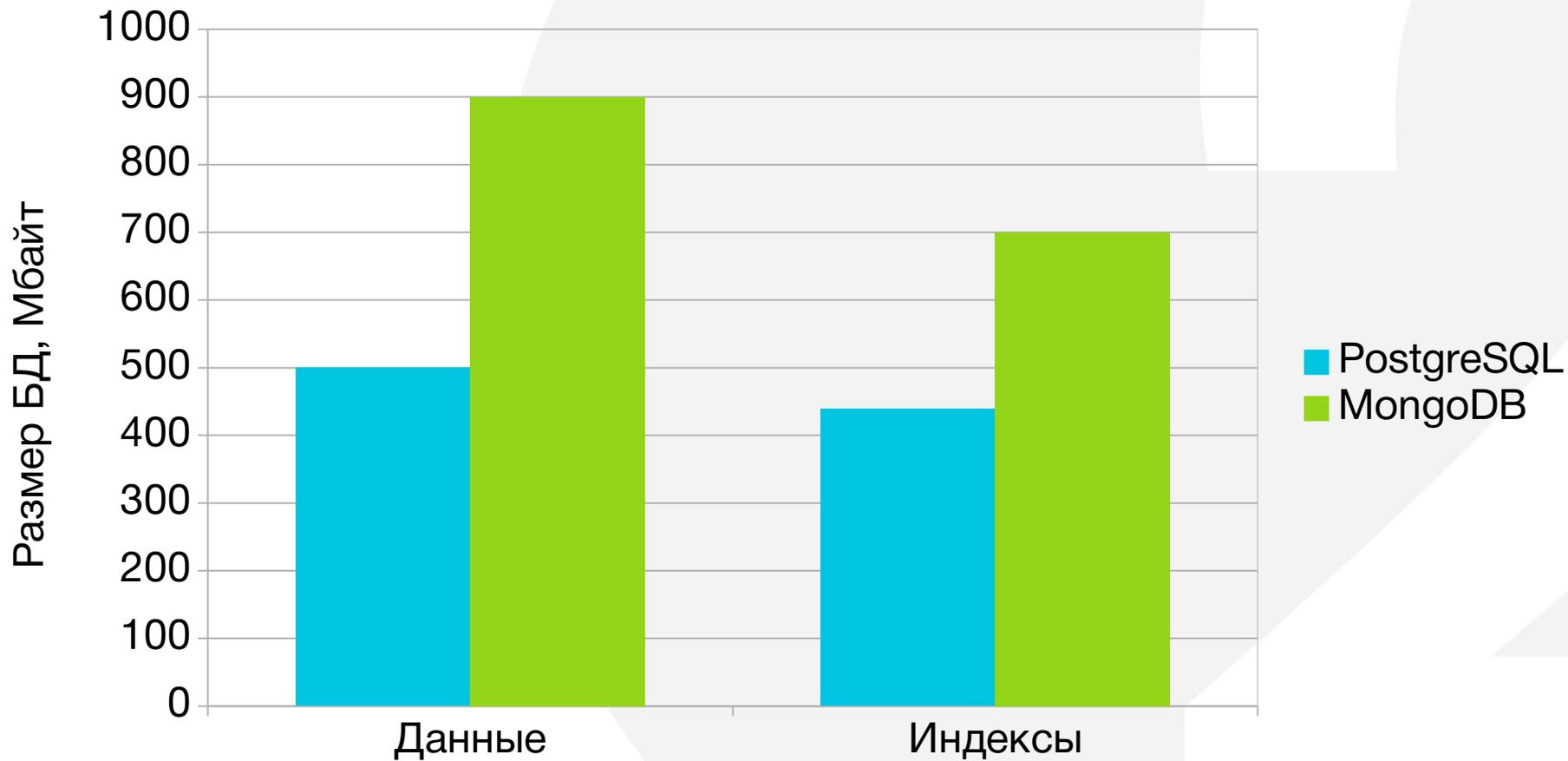


1. Переезд занял две недели

2. Можем получить данные одним запросом

3. Размер БД уменьшился

Размер БД





1. Переезд занял две недели

2. Можем получить данные одним запросом

3. Размер БД уменьшился

4. Выборки стали быстрее

Время ответа и количество активных клиентов

```
SELECT * FROM users AS u  
INNER JOIN user_firm AS uf ON uf.user_id = u.id WHERE uf.firm_id = :firm_id
```

```
db.getCollection('users').find({'firmLinks.firmId': :firm_id})
```





ДОКУМЕНТЫ С РАЗНОРОДНОЙ СТРУКТУРОЙ

Журнал изменений



фирма

id	address	name	entrance
1	ул. Новогодняя, 21	Рога и копыта	{"lat": 12, "lon": 14}

фирма

id	address	name	entrance
1	ул. Новогодняя, 21	Рога и копыта	{"lat": 12, "lon": 14}

запись в журнале

firm_id	operation	user_email	user_name	address
1	edit	user@mail.ru	Андрей	ул. Новогодняя, 22



фирма

id	address	name	entrance
1	ул. Новогодняя, 21	Рога и копыта	{"lat": 12, "lon": 14}

КОНТАКТ

id	firm_id	type	value
10	1	phone	222-33



фирма

id	address	name	entrance
1	ул. Новогодняя, 21	Рога и копыта	{“lat”: 12, “lon”: 14}

КОНТАКТ

id	firm_id	type	value
10	1	phone	222-33

запись в журнале

firm_id	operation	user_email	user_name	contacts
1	edit	user@mail.ru	Андрей	{ «id»: «10», «value»: «111-22» }





ДВА ВАРИАНТА СТРУКТУРЫ ДАННЫХ



firm_id	operation	user_email	address	entrance	и ещё 15 полей..
1	edit	user@mail.ru	ул. Новогодняя, 21	{"lat": 12, "lon": 14}	

firm_id	operation	user_email	fields
1	edit	user@mail.ru	{ «address»: «ул. Новогодняя, 21», «name»: «Рога и копыта» }



ПЕРЕНОС ДАННЫХ



1. Контакты переносим в таблицу



1. Контакты переносим в таблицу
2. Переносим общие поля



1. Контакты переносим в таблицу
2. Переносим общие поля
3. Остальные поля сливаем в JSONB-поле



1. Контакты переносим в таблицу
2. Переносим общие поля
3. Остальные поля сливаем в JSONB-поле
4. Перенос через mosql 20 минут



1. Контакты переносим в таблицу
2. Переносим общие поля
3. Остальные поля сливаем в JSONB-поле
4. Перенос через mosql 20 минут
5. Перенос контактов 10 минут



РЕЛИЗ



1. Выключаем проект, предупредив коллег
2. Показываем заглушку
3. Переносим данные
4. Релизим код
5. Проверяем целостность данных
6. Включаем проект

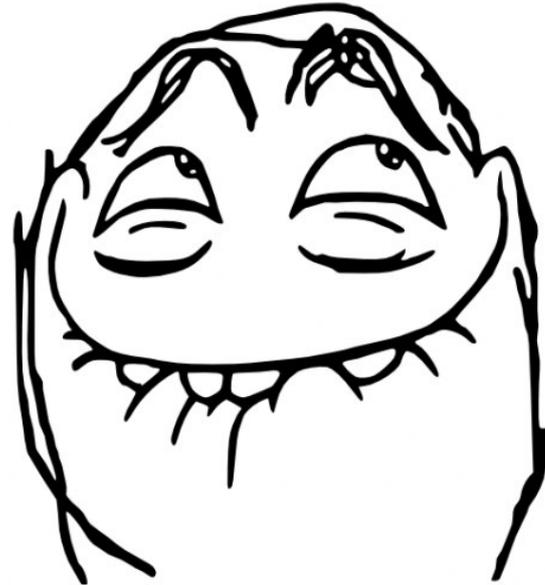


ВЫВОДЫ ПРО КОЛЛЕКЦИЯМ С РАЗНОРОДНОЙ СТРУКТУРОЙ



1. Получаем данные с учётом логов

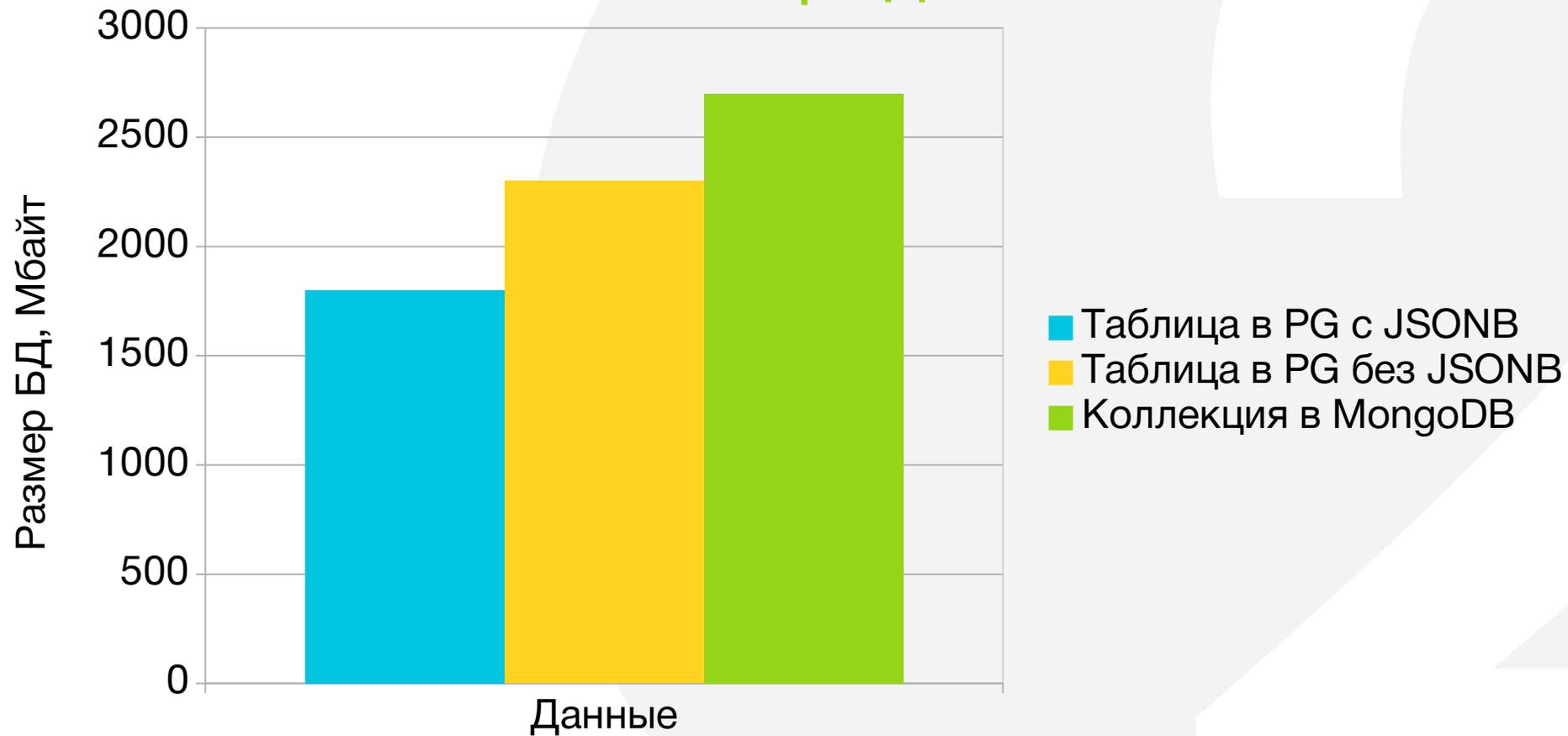
```
SELECT (  
    CASE WHEN fl.changes->>"name" IS NOT NULL  
        THEN fl.changes->>"name"  
    ELSE  
        f.name  
    END  
) AS name  
FROM firm AS f  
LEFT JOIN firm_log AS fl  
    ON fl.firm_id = f.id
```





1. Получаем данные с учётом логов
2. Сохранение колонок в JSONB самое компактное

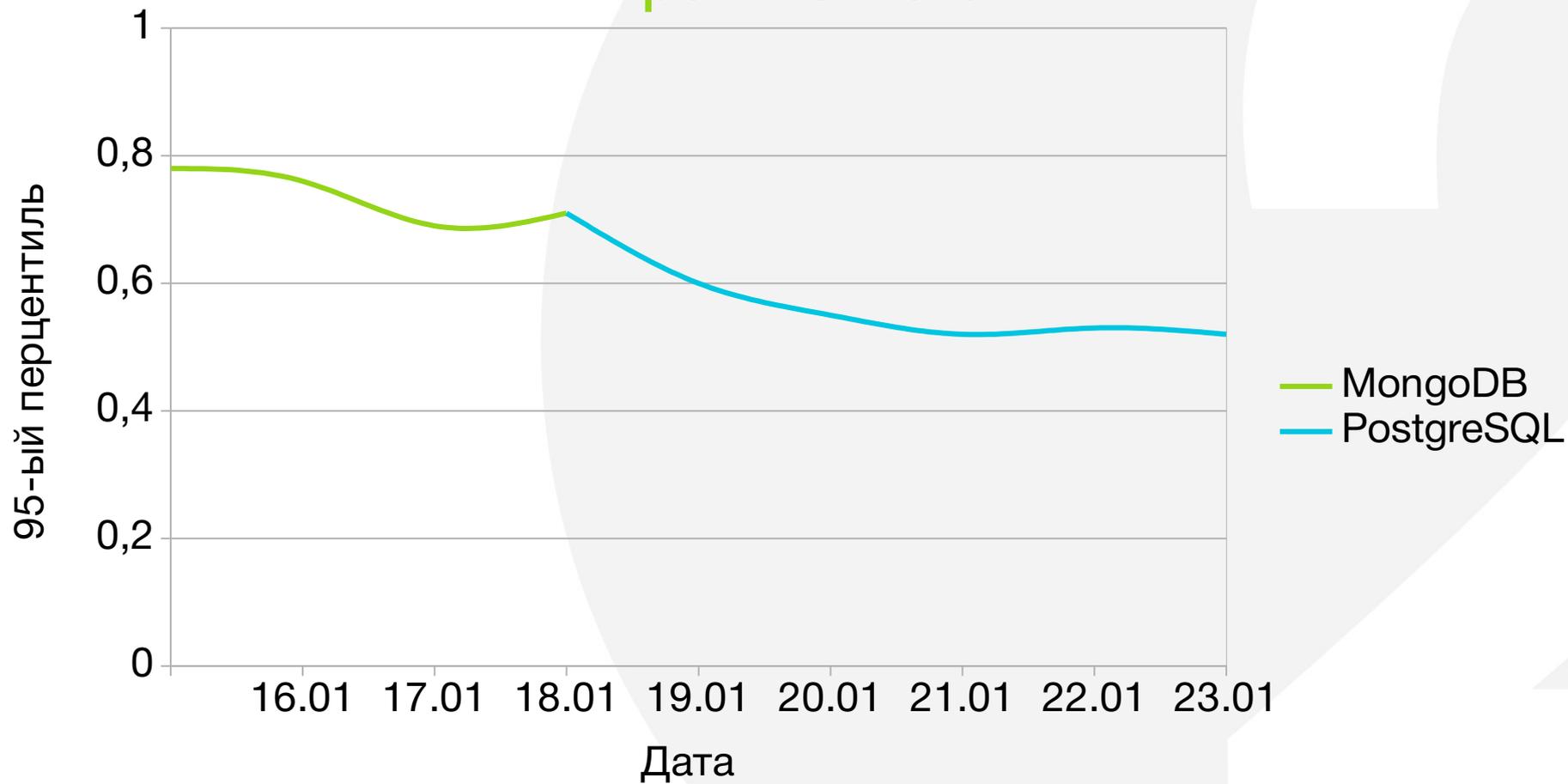
Размер БД





1. Получаем данные с учётом логов
2. Сохранение колонок в JSONB самое компактное
3. Без кастомных мерджингов и джоинов приложение стало быстрее

Время ответа





ПОДВОДНЫЕ КАМНИ MOSQL



1. Режим работы по-умолчанию - репликация из MongoDB в PostgreSQL



1. Режим работы по-умолчанию - репликация из MongoDB в PostgreSQL
2. Таблица реинициализируется при разовом импорте



1. Режим работы по-умолчанию - репликация из MongoDB в PostgreSQL
2. Таблица реинициализируется при разовом импорте
3. У пользователя должен быть доступ к БД admin в MongoDB



СОВЕТЫ

1

Подумайте, точно ли нужно использовать две СУБД

2

Воспользуйтесь утилитой `mosql` для переноса данных

3

Пост-изменения схемы под нужный формат

4

Переносите по одной коллекции

5

Декомпозируйте задачу, если необходимо

6

Определите, нужно ли останавливать проект



ВЫВОДЫ



1. Отчёты можем сделать одним запросом



1. Отчёты можем сделать одним запросом
2. Удалили мерджинг данных на клиенте



1. Отчёты можем сделать одним запросом
2. Удалили мерджинг данных на клиенте
3. Приложение стало в 2 раза быстрее



1. Отчёты можем сделать одним запросом
2. Удалили мерджинг данных на клиенте
3. Приложение стало в 2 раза быстрее
4. Избавились от MongoDB



СПАСИБО!

Литуненко Андрей