

## **ЖУРНАЛИРОВАНИЕ ИЗМЕНЕНИЙ БАЗЫ ДАННЫХ ВЕБ-ПРИЛОЖЕНИЯ**

**Антипов М. А.**, Университет ИТМО, Санкт-Петербург

**Научный руководитель – тьютор Афанасьев Д. Б.**, Университет ИТМО, Санкт-Петербург

Современные веб-приложения часто подразумевают совместную работу пользователей с общими данными. Веб-редакторы документов, облачные файловые хранилища, различное ПО для автоматизации бизнеса или отдельных бизнес-процессов компаний. Как массовые, так и узкоспециализированные программные продукты для совместной работы испытывают потребность вести журнал изменений, которые вносят пользователи, и отображать этот журнал изменений в наглядной и понятной пользователям форме. Такой журнал может быть полезен пользователю, например, для анализа того, кто и когда внес в систему некорректные данные, куда пропали загруженные данные и т. п.

В данной работе решается задача создания журнала изменений, в котором будут отображены данные обо всех операциях добавления, изменения и удаления данных в целевой системе. Целевой системой является специализированное веб-приложение для автоматизации работы логистических компаний, значительная часть бизнес-логики которого реализована с помощью триггеров в базе данных. В ходе работы производится сравнение подходов к организации журналирования в веб-приложении, анализ их применимости в рамках целевой системы и разработка решения.

В работе выделяются две группы способов логирования изменений данных: на уровне приложения и на уровне СУБД. Логирование на уровне СУБД включает в себя использование триггеров и анализ журнала транзакций. Журналирование на уровне приложения – эффективное решение в большинстве ситуаций, но в некоторых случаях оно невозможно или затруднено. Например, в случае если часть бизнес-логики приложения реализуется при помощи триггеров и хранимых процедур в базе данных. В таком случае использование логирующих триггеров – закрепившееся в отрасли решение, но оно не лишено проблем и ограничений. В системе со сложной бизнес-логикой в базе данных, каждое изменение может приводить к срабатыванию цепочки триггеров и изменению связанных данных. Это повлечет за собой большое количество вызовов логирующих процедур и даст некоторую дополнительную задержку во времени отклика базы данных. Анализ журнала транзакций – менее распространенный способ, который подойдет в ситуациях, когда нельзя или затруднительно использовать другие методы.

В рамках работы была произведена разработка микросервиса, использующего механизм логического декодирования PostgreSQL – потоковую передачу изменений в базе данных внешним потребителям. Микросервис подключается к основной базе данных, получает от неё изменения, обрабатывает их, производя анализ и подготовку к записи, сохраняет информацию об изменениях. Сервис предоставляет REST API для работы с созданным журналом изменений. Когда используется журнал транзакций базы данных для извлечения изменений, реализация не зависит от того, кто является источником изменения. Это важно, так как изменение может исходить не только от приложения, но и, например, от администратора базы данных. При таком решении также отсутствует дополнительная нагрузка на основную базу данных в виде логирования изменений в таблицу, а место в основной базе данных не тратится на журнал изменений, так как его можно вынести в другую базу данных и, например, разместить на выделенном сервере.

Результат работы – микросервис в составе веб-приложения – уже используется в целевой системе в режиме опытной эксплуатации.