|  |  |  |
| --- | --- | --- |
|  |  |  |
| МИНОБРНАУКИ РОССИИ | | |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | |

Институт Информационных технологий (ИТ)

Кафедра Инструментального и прикладного программного обеспечения

(ИиППО)

**ОТЧЕТ О ПРАКТИЧЕСКОЙ РАБОТЕ №9**

**по дисциплине**

«Технологии обработки транзакций клиент-серверных приложений»

Выполнил студент группы ИКБО-20-21 Хитров Н.С.

Принял Маличенко С.В.

Москва 2024**ЗАДАНИЕ НА ПРАКТИЧЕСКУЮ РАБОТУ**

1. Напишите тест, проверяющий, что обработка очереди, показанная в демонстрации, работает корректно при выполнении в несколько потоков. Убедитесь, что тест не проходит, если убрать предложение FOR UPDATE SKIP LOCKED.
2. Добавьте в реализацию проверку «зависших» сообщений. Если такая ситуация будет обнаружена, зависшее сообщение должно быть снова принято в работу.
3. Вставьте в таблицу сообщений большое количество строк и проверьте, что:
   1. было обработано каждое сообщение;
   2. каждое сообщение было обработано ровно один раз.
4. Уберите из реализации секундную задержку (имитацию работы), чтобы тест выполнялся быстрее и с достаточным уровнем конкурентности между процессами.

**РЕФЕРАТ**

Отчёт 17 страниц, 31 рисунков, 5 источников.

POSTGRESQL, АСИНХРОННАЯ ОБРАБОТКА, ОЧЕРЕДЬ, БЛОКИРОВКИ, SQL

Объектом разработки является база данных.

Цель работы – реализация асинхронных механизмов обработки транзакций для повышения эффективности системы, исследовании проблем в области транзакционной асинхронной обработки.

В процессе работы производилось изучение механизмов асинхронной обработки транзакций и исследование проблем в области транзакционной асинхронной обработки.

Результатом являются сведения по реализации асинхронных механизмов обработки транзакций.

**ВВЕДЕНИЕ**

[ВВЕДЕНИЕ 6](#_Toc2266)

[1. ВЫПОЛНЕНИЕ ПРАКТИЧЕСКОЙ РАБОТЫ 7](#_Toc15883)

[ЗАКЛЮЧЕНИЕ 16](#_Toc14449)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 17](#_Toc24146)

**ПЕРЕЧЕНЬ СОКРАЩЕНИЙ И ОБОЗНАЧЕНИЙ**

БД – база данных;

СУБД – система управления базами данных.

**ВВЕДЕНИЕ**

В современном мире, где данные играют важнейшую роль в различных сферах, от бизнеса до повседневной жизни, вопрос обеспечения высокой скорости обработки большого объёма данных является одним из ключевых. Для решения данной задачи используется асинхронная обработка данных с распределением по разным вычислительным машинам с использованием механизма очередей.

Целью данной практической работы является реализация асинхронных механизмов обработки транзакций для повышения эффективности системы, исследовании проблем в области транзакционной асинхронной обработки.

1. **ВЫПОЛНЕНИЕ ПРАКТИЧЕСКОЙ РАБОТ****Ы**
   1. **Выполнение задания 1**

Перед выполнением работы были созданы таблицы msg\_queue и msg\_log для реализации журнала и самой очереди соответственно. (Рис. 1 - 2)

****

Рисунок 1 - Создание таблицы msg\_queue

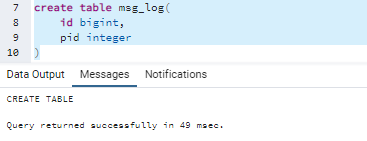
****

Рисунок 2 - Создание таблицы msg\_queue

Далее была заполнена таблица очереди. (Рис. 3)

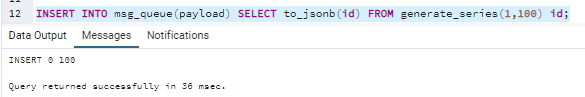


Рисунок 3 - Заполнение таблицы сообщений

Затем, были созданы вспомогательные функции и процедура обработки очереди. (Рис. 4 – 6)

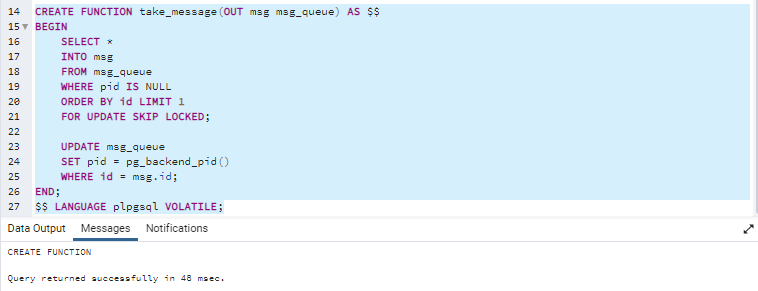


Рисунок 4 – Создание функции приема сообщения

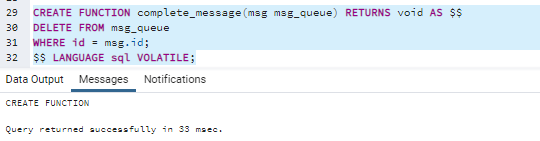


Рисунок 5 – Создание функции выполнения процесса из очереди

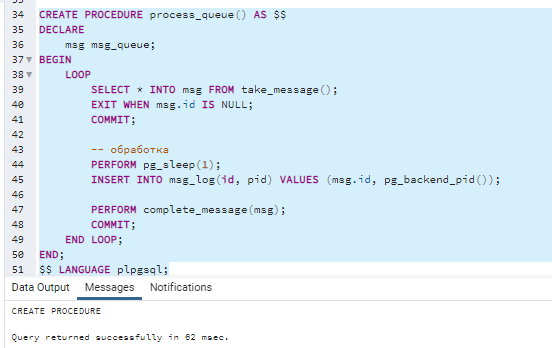


Рисунок 6 – Создание процедуры обработки очреди

После была запущена обработка очереди в сессии 1 с включённым секундомером и в сессии 2. (Рис. 7 – 8)

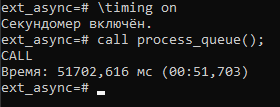


Рисунок 7 - Обработка очереди в первом терминале



Рисунок 8 - Обработка очереди во втором терминале

В итоге была выполнена проверка, что все записи очереди были обработаны и кол-во записей журнала совпадает с кол-вом обработанных позиций в очереди, что доказывает корректность обработки очереди. (Рис. 9 – 10)

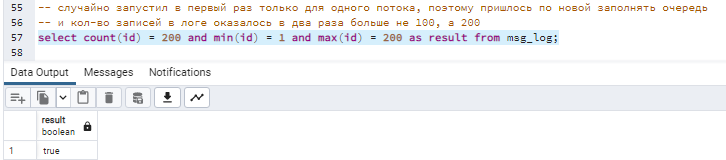


Рисунок 9 – Проверка кол-ва записей в журнале

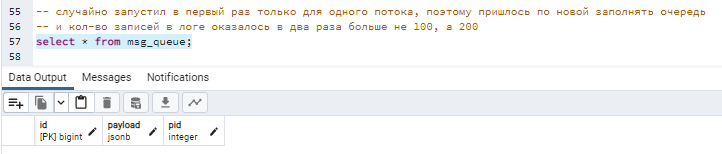


Рисунок 10 – Проверка отсутствия в очереди необработанных сообщений

Для проверки работоспособности очереди без команды FOR UPDATE SKIP LOCKED, код обработки очереди был изменён как показано на рисунке 11, после чего очередь была заполнена 100 записями (Рис. 12).

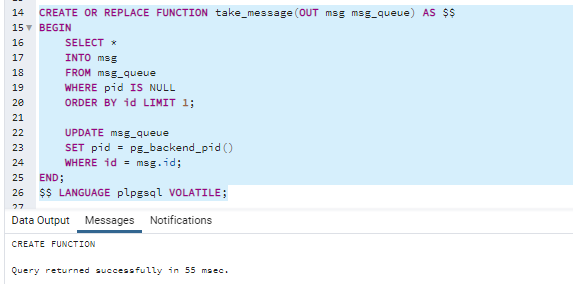


Рисунок 11 – Функция приема сообщений в очереди без FOR UPDATE

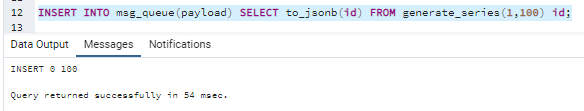


Рисунок 12 - заполнение таблицы сообщений

После заполнения была повторно запущена обработка очереди в двух терминалах. (Рис 13 – 14).

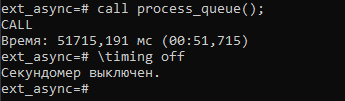


Рисунок 13 - Обработка очереди в первом терминале

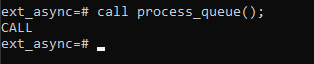


Рисунок 14 - Обработка очереди во втором терминале

В результате, после обработки видно, что часть сообщений была обработана повторно.

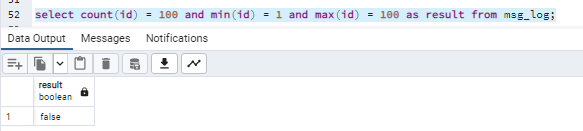


Рисунок 15 – Проверка кол-ва записей в журнале

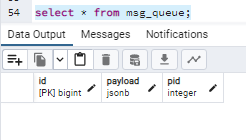


Рисунок 16 – Проверка отсутствия необработанных записей очереди

* 1. **Выполнение задания 2**

Во втором задании в реализацию процедуры обработки очереди был добавлен функционал проверки и обработки «зависших» сообщений. (Рис. 17)

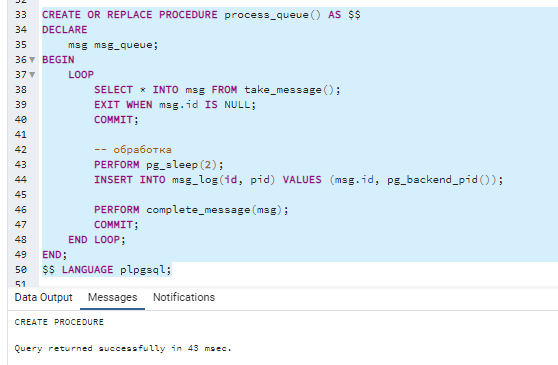


Рисунок 17 – Обновленная процедура

Перед запуском проверок таблица очереди была заполнена 100 записями (Рис. 18). После чего в первой сессии была запущена обработка очереди (Рис. 19)

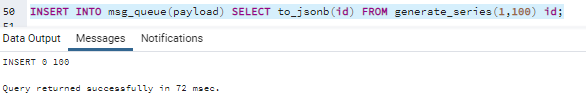


Рисунок 18 - Заполнение таблицы сообщений

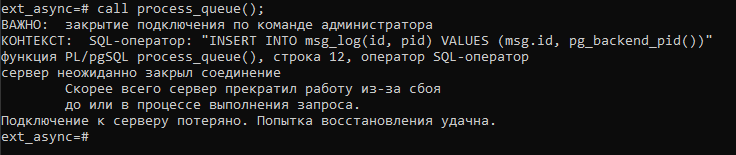


Рисунок 19 - Запуск обработки очереди в первом терминале

Для появления зависшего сообщения, из другого терминала была заблокирована работа обработчика (Рис. 20).

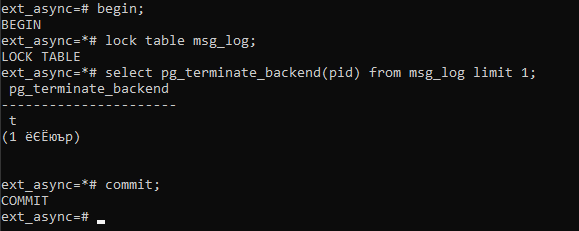


Рисунок 20 - Прерывание обработки очереди в терминале 2

На рисунке 21 представлено наличие зависших из-за блокировки сообщений

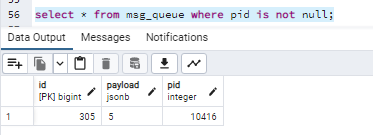


Рисунок 21 - Проверка наличия зависших сообщений

Для обработки зависших сообщений процедура обработки очереди была вызвана заново, после чего была выполнена проверка корректной обработки всех сообщений.

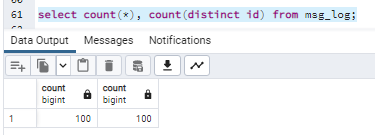


Рисунок 22 - Результат повторного вызова обработки очереди

* 1. **Выполнение задания 3**

Для выполнения 3 задания таблица очереди была заполнена большим кол-вом записей, после чего была запущена процедура обработки очереди в двух сессиях. (Рис. 15 – 16)

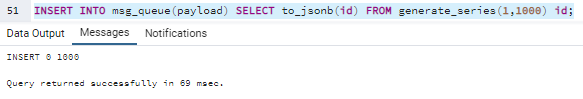


Рисунок 23 - Заполнение очереди большим количеством сообщений

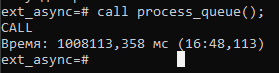


Рисунок 24 - Обработка очереди в первой сессии



Рисунок 25 - Обработка очереди во второй сессии

В результате обработки очереди были единожды обработаны все сообщения (Рис. 26)

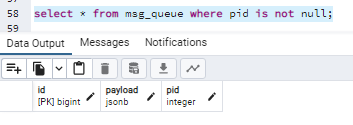


Рисунок 26 - Результат проверки корректной обработки очереди

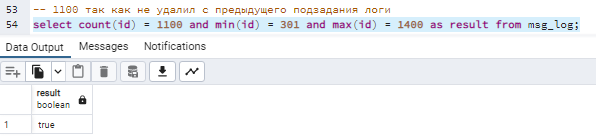


Рисунок 27 - Результат проверки корректной обработки очереди

* 1. **Выполнение задания 4**

Перед выполнением задания очередь вновь была заполнена большим кол-вом данных (Рис. 28)

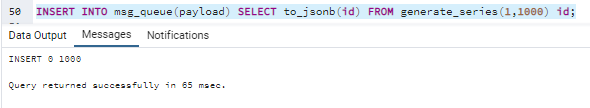


Рисунок 28 - Заполнение таблицы сообщений большим количеством сообщений

После заполнения таблицы был начат процесс обработки в двух сессиях (Рис. 29).

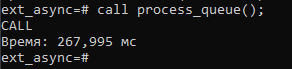


Рисунок 29 – Вызов процедуры обработки очереди

В результате обработки были единожды обработаны все сообщения (Рис. 30 – 31)

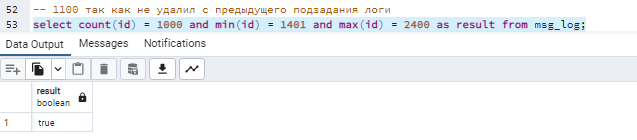


Рисунок 30 - Результат обработки очереди

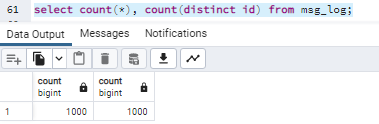


Рисунок 31 - Результат обработки очереди

# ЗАКЛЮЧЕНИЕ

В ходе выполнения практической работы были получены практические навыки по реализации асинхронных механизмов обработки транзакций для повышения эффективности системы, а также знания о проблемах в области транзакционной асинхронной обработки.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. PostgreSQL: Официальная документация [Электронный ресурс] – URL: https://www.postgresql.org/docs/ (дата обращения: 14.02.2024).
2. PostgreSQL Tutorial for Beginners [Электронный ресурс] – URL: https://www.tutorialspoint.com/postgresql/index.htm (дата обращения: 14.02.2024).
3. PostgreSQL: Википедия [Электронный ресурс] – URL: https://en.wikipedia.org/wiki/PostgreSQL (дата обращения: 14.02.2024).
4. Лузанов П.В. и др. Postgres. Первое знакомство. [Электронный ресурс] – URL: https://postgrespro.ru/education/books/introbook (дата обращения: 22.02.2024)
5. Новиков Б. А. Лекции Основы технологий баз данных. [Электронный ресурс] – URL: https://postgrespro.ru/education/university/dbtech (дата обращения: 22.02.2024)