Федеральное государственное автономное

образовательное учреждение

высшего образования

«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт Космических и информационных технологий

институт

Кафедра «Информатика»

кафедра

Отчёт о научно-исследовательской работе

Сравнение скорости CRUD операций (Clickhouse c PostgreSQL)

тема

|  |  |  |
| --- | --- | --- |
| Руководитель |  | А. Н. Пупков |
|  | подпись, дата | инициалы, фамилия |
| Студент КИ18/17-1б 031831229 |  | В. А. Прекель |
| номер группы, зачетной книжки | подпись, дата | инициалы, фамилия |

Красноярск 2020

Содержание

[Содержание 2](#_Toc49799774)

[Введение 3](#_Toc49799775)

[1 История создания 4](#_Toc49799776)

[1.1 Разработка IntelliJ IDEA 4](#_Toc49799777)

[Заключение 5](#_Toc49799778)

[Список использованных источников 6](#_Toc49799779)

Индивидуальное задание

Необходимо провести сравнение скорости и анализ CRUD операций для систем управления баз данных Clickhouse и PostgreSQL.

Теоретическая часть:

1. Описать что такое CRUD операции;
2. Описать как работают хранилища данных, ссылаясь на соответствующую документацию (например если вы пишете про ClickHouse, нужно сослаться на документ https://clickhouse.yandex/docs/ru/);
3. Найти информацию о том, как и почему скорость CRUD операций хранилищ отличается, провести сравнительный анализ для каждой операции с детальным и обоснованным объяснением (со ссылками на источники);
4. Сделать выводы о том, почему в данных хранилищах имеются различия в выполнении CRUD операций, чем это вызвано и как дизайн системы влияет на данный параметр.

Экспериментальная часть:

* Установить docker toolbox (или более свежее решение);
* Скачать контейнеры с соответствующими базами данных;
* Написать два простых скрипта выполняющих CRUD операции для каждой из пары баз данных и измеряющих время выполнения;
* Каждый эксперимент провести несколько раз, при этом:
* Нужно указать параметры (виртуальной) машины, на которой проводились исследования (кол-во RAM, CPU, потоков);
* Указать количество итераций для каждого эксперимента;
* Привести значения математического ожидания и дисперсии для каждого результата;
* Сделать графики с пояснениями;
* Сделать выводы о том, почему в данных хранилищах имеются различия в выполнении CRUD операций, чем это вызвано и как дизайн системы влияет на данный параметр.

Введение

Система управления базами данных – набор программного обеспечения, позволяющий определять, обрабатывать, получать и управлять данными в базе данных. [1] Соответственно, основные операции – создания, чтения, изменения и удаления называются CRUD-операциями. В случае с SQL-СУБД, за CRUD операции отвечают соответственно конструкции INSERT, SELECT, UPDATE, DELETE соответственно. [2] В разных системах, с разным дизайном, на идентичной схеме данных, время выполнения CRUD-операций может различаться.

# Теоретическая часть

## Описание СУБД

Существуют строковые и столбцовые (колоночные, column-based) СУБД. Они отличаются тем, что данные, принадлежащие к одной строке в строковых СУБД, хранятся рядом, а в столбцовых СУБД хранятся рядом данные, принадлежащие к одному столбцу. Это позволяет экономить время в запросах на чтение, которые не затрагивают все столбцы таблицы и проводить прочие оптимизации, в том числе по сжатию данных, которые невозможны в столбцовых СУБД. [3] Примеры строковых СУБД: MySQL, PostgreSQL, и MS SQL Server. Примеры столбцовых СУБД: ClickHouse, Vertica, Paraccel, Sybase IQ, Exasol, Infobright, InfiniDB, MonetDB, LucidDB, SAP HANA, Google Dremel, Google PowerDrill, Druid, and kdb+. [4]

ClickHouse – строковая СУБД, предназначенная для анализа данных и OLAP-запросов. Поддерживает SQL, хоть с во многих случаях не совпадающий со стандартом. Представляется как по-настоящему столбцовая СУБД, поддерживающая сжатие данных, хранящая данные на диске, параллельно использующая процессорные ядра, и позволяющая работать на нескольких серверах в кластере. [5] Была разработана в Яндексе и выпущена под открытой лицензией в 2016 году. [6]

PostgreSQL – строковая объектно-реляционная СУБД. Поддерживает большую часть SQL стандарта, а также множество современных функций, такие как сложные запросы, внешние ключи, триггеры, изменяемые представления, транзакционная целостность, многоверсионность. Начинает свою историю из 80-тых годов с проекта POSTGES в Беркли. PostgreSQL основано на последней версии POSTGRES 4.2, выпущенной в 1994 году. [8]

## Анализ работы СУБД

Исходя из того, что PostgreSQL строковая СУБД, то запросы в узких таблицах должны работать быстрее. Так же, ClickHouse предназначен для извлечения и вставки большого количества данных одним запросом, поэтому выполнение множества аналогичных запросов должно быть медленнее, чем в PostgreSQL. [9]

Так же ClickHouse поддерживает операции изменения и удаления через механизм мутаций нестандартным синтаксисом ALTER TABLE … UPDATE и ALTER TABLE … DELETE соответственно. [10] Механизм мутаций не был реализован на релизе Clickhouse и логично предположить, изменение/удаление данных будет работать хуже, чем в строковых СУБД. [11]

# История создания

## Разработка IntelliJ IDEA

История IntelliJ NetBeans

NetBeans, распространяется по свободной лицензии Apache 2.0. Но имеет ряд недостатков таких как худшая производительность, худший тектовый редактор и т.д. [15]

Заключение

В заключение можно отметить, что благодаря грамотной политики, маркетингу и умелой разработке [16] компания JetBrains удачно конкурирует среди производителей IDE, даже таких как Microsoft, выпустив IDE для .NET Rider.

Список использованных источников

1. <https://www.techopedia.com/definition/24361/database-management-systems-dbms>

1. <https://en.wikipedia.org/wiki/Create,_read,_update_and_delete>

1. <https://en.wikipedia.org/wiki/Column-oriented_DBMS>

1. <https://clickhouse.tech/docs/en/>

1. <https://clickhouse.tech/docs/en/introduction/distinctive-features/>

1. <https://habr.com/en/company/yandex/blog/303282/>

1. <https://www.postgresql.org/docs/12/intro-whatis.html>

1. <https://www.postgresql.org/docs/12/history.html>

1. <https://clickhouse.tech/docs/en/introduction/performance/>

1. <https://clickhouse.tech/docs/en/sql-reference/statements/alter/>

1. <https://clickhouse.tech/blog/en/2016/how-to-update-data-in-clickhouse/>