

СРАВНИТЕЛЬНАЯ ОЦЕНКА ПРОИЗВОДИТЕЛЬНОСТИ ОБЛАЧНОЙ И ТРАДИЦИОННОЙ БАЗ ДАННЫХ ПО КРИТЕРИЮ ВРЕМЕНИ ОТКЛИКА

COMPARATIVE EVALUATION OF THE PERFORMANCE OF CLOUD AND TRADITIONAL DATABASES BY THE RESPONSE TIME CRITERION

V. Voronin
O. Romashkova

Summary. This paper attempts to evaluate and compare the performance of traditional and cloud databases. The size of databases has increased exponentially in recent years, and the same growth is expected in the future. The cost of data storage has declined rapidly, and this has been followed by a rapid increase in storage capacity. The recent emergence of cloud databases has brought about many changes. Database performance plays an important role in computing.

Keywords: database, cloud computing, performance.

Воронин Владимир Алексеевич

Аспирант
Московский городской педагогический
Университет (МГПУ)
г. Москва
mgvoron@gmail.com

Ромашкова Оксана Николаевна

Доктор технических наук, профессор
Российская академия народного хозяйства
и государственной службы при Президенте РФ
г. Москва
ox-rom@yandex.ru

Аннотация. В этой статье был проведён эксперимент по оценке и сравнению производительности традиционных и облачных баз данных. Размер баз данных в последние годы вырос экспоненциально, и, как ожидается, будет расти и в будущем. Стоимость хранения данных значительно снизилась, а ёмкость хранилищ быстро увеличивается. Появление облачных баз данных в последнее время принесло много изменений в технологии хранения информации. Производительность базы данных играет жизненно важную роль в вычислениях, и следовательно, существенным образом влияет на выбор в прикладных решениях.

Ключевые слова: база данных, облачные вычисления, производительность.

Введение

Облако можно определить как параллельную и распределённую информационную систему, в которую входят несколько виртуализированных и взаимосвязанных компьютеров. Облака представляются в виде отдельных или различных объединённых вычислительных ресурсов. Облачные технологии могут использоваться в виде трёх различных архитектур: инфраструктура как услуга (Infrastructure as a Service, IaaS), платформа как услуга (Platform as a Service, PaaS) и программное обеспечение как услуга (Software as a Service, SaaS). Эти архитектуры включают распределённые операционные системы, распределённые базы данных и другие сервисы [1, 2].

Облачные базы данных позволяют производить различные вычисления быстро, а также они экономят время на настройку маршрутизации. Облачная база данных представляет собой совокупность нескольких сайтов. Сайты также называются узлами, и они соединя-

ются с помощью телекоммуникационной сети. Отдельные узлы являются классами базы данных.

База данных — место для хранения набора записей данных, файлов и других объектов. Система управления базами данных (СУБД) — программное обеспечение, с помощью которого можно создавать, изменять, получать информацию из баз данных. Такое программное обеспечение позволяет разрабатывать базы данных для различных целей. СУБД позволяет различным пользовательским приложениям одновременно обращаться к одной и той же базе данных. СУБД может легко описывать и поддерживать приложения, использующие различные модели баз данных, такие как реляционная модель или объектная модель. Термин «база данных» применяется к данным и структурам данных, которые их поддерживают. Сочетание базы данных и СУБД называется системой баз данных [1, 3].

База данных — это совокупность данных, которая облегчает доступ, обновление и управление данными.

Облачная база данных — это база данных, которая работает на платформе облачных вычислений, таких как Windows Azure или Amazon EC2. Две распространённые модели развёртывания — это запуск пользователями собственных баз данных в облаке с помощью образов виртуальных машин, или приобретение доступа к услугам баз данных, поддерживаемым поставщиками облачных баз данных. Облачные базы данных в основном основаны либо на SQL-модели, либо на NoSQL-модели данных.

Системы управления базами данных

Как упоминалось выше, система управления базами данных — это компьютерное программное обеспечение, с помощью которого пользователи создают и используют базы данных. Используя базы данных, можно эффективно хранить и извлекать необходимые данные. Цель пользовательского интерфейса — создание новых баз данных в системе или обновление существующих. Согласно глоссарию IBM Computing Dictionary, словарь данных — это централизованное хранилище информации о данных, включая собственно значение, связь с другими данными, источник, использование и формат данных. Данный документ определяет структуру и описывает базы данных. СУБД может облегчить доступ к нескольким базам данных одновременно через пользовательский интерфейс.

Система управления базами данных служит платформой для администрирования баз данных. Администраторы могут запускать определенные приложения в СУБД для доступа, изменения и обновления данных.

Существуют различные типы баз данных: сетевые, иерархические и реляционные. Реляционные базы данных являются лучшим выбором для хранения данных по сравнению с другими моделями [1, 3].

Оптимизация баз данных

В современном мире компании сосредоточены на использовании большого количества данных. Эти данные, хранятся в базе данных и эффективно управляются. Поскольку компании постоянно анализируют эти базы данных и принимают обоснованные решения на основе анализа, поэтому производительность базы данных играет решающую важную роль в общей функциональности корпоративных и управленческих информационных систем. В годы создания базы данных размер метаданных, связанных с ней, был небольшим. Однако с увеличением размера базы данных производительность постепенно снижается. В ответ на такое снижение производительности исследователи ищут

способы повышения производительности путём оптимизации структуры базы данных. Сегодня наиболее распространённым способом оптимизации базы данных является индексирование [4]. Индексы внутренне организованы в виде древовидной структуры. Индексирование базы данных позволяет СУБД поддерживать отдельный объект базы данных для хранения метаданных, связанных с базой данных.

Несмотря на эти преимущества, индексирование баз данных имеет и определённые недостатки. Индексирование ускоряет запросы и поиск данных, но каждый раз, когда индекс добавляется в индексную таблицу, это ещё больше замедляет работу. Поскольку каждый запрос обрабатывается только после обновления всех соответствующих индексов, для синхронизации индексов и таблиц требуются дополнительные циклы центрального процессора (ЦП) и время. Это также потребует дополнительного места в базе данных [3].

Облачные вычисления

Трудно определить, что такое облачные вычисления, поскольку определения облачных вычислений у разных авторов различные. Облачные вычисления — это модель, которая обеспечивает повсеместный, удобный, сетевой доступ по требованию к общему пулу конфигурируемых вычислительных ресурсов (сети, серверы, хранилища, приложения и др.), с минимальными административными усилиями или взаимодействием с поставщиками услуг.

Облачные вычисления имеют пять основных характеристик:

1. Самообслуживание по запросу.
2. Широкий доступ к сети.
3. Объединение ресурсов.
4. Быстрая эластичность.
5. Измеряемое обслуживание.

Также есть три модели обслуживания облачных вычислений:

1. Программное обеспечение как услуга (SaaS).
2. Платформа как услуга (PaaS).
3. Инфраструктура как услуга (IaaS).

Для облачных вычислений используются четыре модели развёртывания:

1. Частное облако.
2. Облако сообщества.
3. Общедоступное облако.
4. Гибридное облако.

Производительность традиционных и облачных баз данных

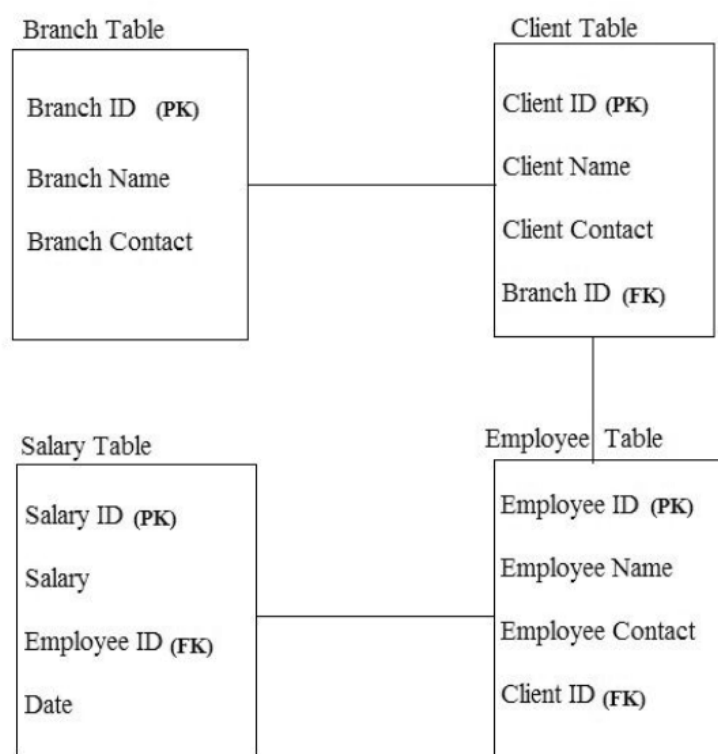


Рис. 1. Диаграмма отношений сущностей базы данных

Производительность баз данных может быть оценена как за счёт времени отклика, так и за счёт пропускной способности, стоимости транзакций и использования ресурсов. Длительное время выполнения запросов негативно влияет на время отклика и в конечном итоге приводит к низкой производительности баз данных. Из этого следует что время отклика на запрос является параметром для измерения производительности баз данных. Оптимальное время отклика системы не должно превышать предела заданного времени отклика.

Для эксперимента в качестве традиционной базы данных выбрана база Microsoft SQL 2016 SP2. Данные формируются с помощью онлайн генератора данных. Для занесения данных в реляционную базу данных используется оператор "Insert" (Вставить). В качестве построения облачной базы данных на основе SQL-модели была выбрана облачная система Microsoft Azure.

Эксперимент посвящен проверке изменения производительности традиционной и облачной баз данных при увеличении количества записей данных. Сначала были внесены первые 30 000 записей, и с этими первыми записями выполнялись стандартные запросы. Далее были добавлены ещё 30 000 записей. И такие шаги выполнялись вплоть до момента, когда количество записей достигло количества 240 000. На рисунке 1 изо-

бражена диаграмма отношений сущностей для базы данных.

Представленная диаграмма отношений сущностей оптимально подходит для тестирования производительности. Истёкшее время отклика использовалось в качестве критерия оценки производительности как в традиционной, так и в облачной базах данных с помощью оператора SQL языка "SELECT".

Действия, которые воспроизводились во время эксперимента, следующие:

1. Были выбраны несколько строк из множества строк в таблице с использованием простых и сложных операций как в традиционной базе данных, так и в облачной базе.
2. Первое действие было повторено 30 раз, и было определено среднее значение времени отклика.

Эксперимент

Как говорилось ранее, для проверки времени отклика традиционной и облачной баз данных использовались запросы SELECT с различными условиями. Каждая выборка повторялась 30 раз, и для каждой попытки отмечалось время ответа и определялось среднее значение времени отклика.

Таблица 1. Среднее время выполнения запроса 1

Количество строк	Полученные результаты	Время отклика традиционной БД, мс	Время отклика облачной БД, мс
30000	15	6	11
60000	15	7	9
120000	38	16	40
240000	89	20	74
480000	184	62	178

Таблица 2. Значение А для запроса 1

Количество записей	30000	60000	120000	240000	480000
А	22500	45000	110000	220000	440000

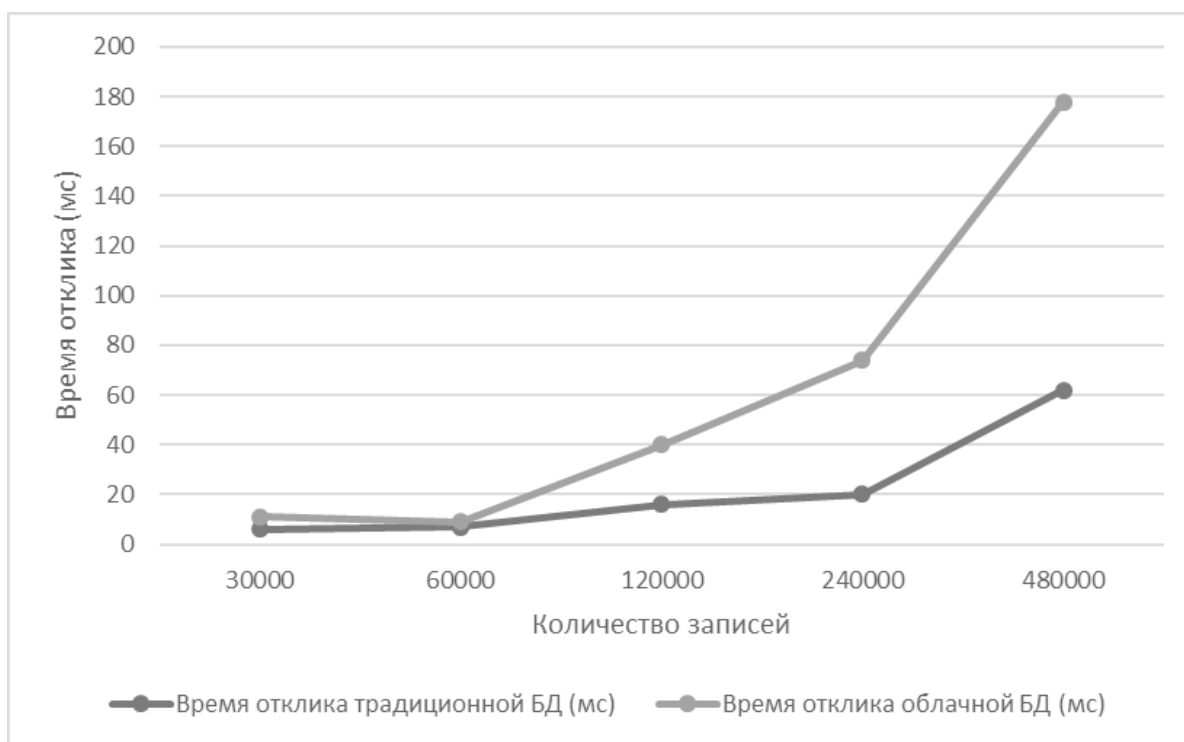


Рис. 2. Результат первого запроса

Также был построен график кривой “замедления” для полученных данных. Указанный график построен путем деления всех времён отклика для всех размеров данных на первоначальное значение времени отклика.

Запрос 1 “SELECT EmployeeID, Date, Salary from Salary where Date = ‘01/02/2019’ and EmployeeID>0 and EmployeeID<A;”

Этот запрос позволяет просканировать всю таблицу, чтобы узнать, сколько времени потребуется для извлечения небольшого количества строк из таблицы.

Этот запрос извлекает данные из таблицы Salary столбцов EmployeeID, Date и Salary для даты “01.02.2019” с диапазоном EmployeeID “<0” и значением А. В таблице 1 и на рисунке 2 показано среднее время выполнения

Таблица 3. Среднее время выполнения запроса 2

Количество строк	Полученные результат	Время отклика традиционной БД, мс	Время отклика облачной БД, мс
30000	24,499	324	1373
60000	48,999	465	1928
120000	97,999	885	4359
240000	195,999	1690	7777
480000	391,999	3235	15587

Таблица 4. Значение А для запроса 2

Количество записей	30000	60000	120000	240000	480000
А	25000	50000	100000	20000	40000

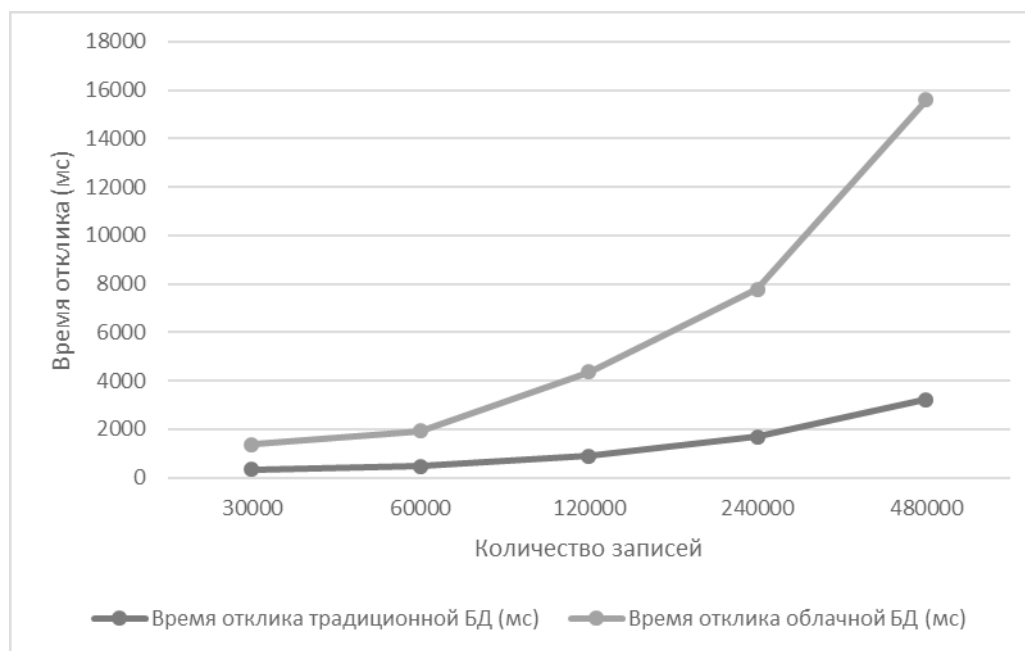


Рис. 3. Результаты запроса 2

запроса для традиционной и облачной баз данных. В таблице 2 представлены значения А для Запроса 1.

Исходя из полученных результатов, можно сделать вывод о том, что традиционная база данных справляется с этим запросом лучше, чем облачная база данных. При количестве записей 30000 время отклика облачной БД превышает почти в 2 раза время отклика в традиционной базе данных. При количестве записей 60000 время отклика одинаковое. При количестве записей 120000 записей время отклика в облаке выше в 2,5 раза. При количестве записей 240 000 время отклика в облачной базе данных выше в 3,5 раза, а при количестве записей 480

000 время отклика облачной базы данных выше в 2,9 раза по сравнению с традиционной базой данных.

Запрос 2

Данный запрос проводится на основе таблиц “Сотрудник” (Employee) и “Salary” (Зарплата). Используя простой запрос SELECT и запрос на соединение JOIN, были выведены данные EmployeeName, EmployeeID и Date.

```
“Set statistics time on select e.EmployeeID, e.EmployeeName, s.Salary, S. Date from Employee e inner
```

join Salary s on e.EmployeeID = S.EmployeeID where EmployeeName > 'a%' and s.SalaryID > 0 and S. SalaryID < A;”

Данный запрос извлекает идентификатор сотрудника, имя сотрудника, зарплату и дату в пределах диапазона SalaryID “0” и значением А. В таблице 3 и на рисунке 3 показано среднее время выполнения запроса для традиционной базы данных и для облачной. В таблице 4 представлены значения А для Запроса 2.

Получив результаты, можно сделать вывод о том, что ситуация повторяется, как и для результатов первого запроса.

Традиционная база данных справляется быстрее, чем облачная база данных.

При количестве записей 30 000 время отклика облачной БД превышает почти в 3 раза время отклика в традиционной базе. При количестве записей 60 000 время отклика у облачной базы данных выше в 4 раза. При количестве записей 120 000 время отклика в облаке выше в 5 раз. При количестве записей 240 000 вре-

мя отклика в облачной базе данных превышает в 4,6 раза. А при количестве записей 480 000 время отклика облачной базы данных выше в 4,8 раза по сравнению с традиционной базой данных.

Заключение

Проведя эксперимент с двумя запросами, используемыми как в традиционной, так и в облачной базах данных, и проанализировав графики зависимостей средних показателей времени отклика, можно сделать вывод о том, что производительность облачной базы данных значительно ниже, чем традиционной базы данных.

Особенно существенное падение производительности облачной базы данных по сравнению с традиционной базой данных можно отметить при увеличении количества записей.

Выполненные исследования позволяют дать конкретные рекомендации для формирования и эксплуатации баз данных в информационных системах различных сфер применения [5, 6, 7].

ЛИТЕРАТУРА

1. Агальцов В.П. Базы данных. В 2-х кн. Книга 2. Распределённые и удалённые базы данных: учебник. — М.: ИД «ФОРУМ»: ИНФРА-М, 2018. 272 с.
2. Ромашкова О.Н., Ломовцев Р.С., Пономарева Л.А. Компьютерная поддержка принятия управленческих решений для образовательной системы регионального уровня // Вестник Рязанского государственного радиотехнического университета. 2019. № 67. С. 50–58.
3. Кэмпбелл Л., Мейджорс Ч. Базы данных. Инжиниринг надёжности. — М.: Издательство Питер, 2020. — 304с.
4. Ромашкова О.Н., Пономарева Л.А., Василюк И.П. Применение инфокоммуникационных технологий для анализа показателей рейтинговой оценки вуза // В книге: Информационно-телекоммуникационные технологии и математическое моделирование высокотехнологичных систем. Материалы Всероссийской конференции с международным участием. 2018. С. 65–68.
5. Заболотникова В.С., Ромашкова О.Н. Информационная управленческая система для налоговой службы // Современная наука: актуальные проблемы теории и практики. Серия: Естественные и технические науки. 2017. № 6. С. 27–32.
6. Gaidamaka, Y.V., Romashkova, O.N., Ponomareva, L.A., Vasilyuk, I.P. Application of information technology for the analysis of the rating of university // В сборнике: CEUR Workshop Proceedings 8. Сер. “ITMM 2018 — Proceedings of the Selected Papers of the 8th International Conference “Information and Telecommunication Technologies and Mathematical Modeling of High-Tech Systems”” 2018. С. 46–53.
7. Ромашкова О.Н., Фролов П.А. Технология расчета показателей прибыли и рентабельности в коммерческой организации // Фундаментальные исследования. 2016. № 4–1. С. 102–106.