

УДК 004.65

## УНИВЕРСАЛЬНЫЙ АЛГОРИТМ ОБРАБОТКИ ЗАПРОСОВ С ИСПОЛЬЗОВАНИЕМ ТЕХНОЛОГИИ ПАРАЛЛЕЛЬНЫХ ВЫЧИСЛЕНИЙ

Тимофеева Н.Е., Дмитриева К.А.

Саратовский национальный исследовательский государственный университет имени Н.Г. Чернышевского  
(Саратов, Россия)

В настоящее время обработка и хранение большого объема информации является одной из сложных и интересных задач. От того, насколько качественно реализованы производительность и надежность базы данных, зависит быстродействие системы в целом. Одним из сложных моментов данного вопроса является обработка запроса к базе данных и его эффективное выполнение. В данной работе рассмотрены современные методы и модели обработки запросов в базах данных. Предложен алгоритм для обслуживания запроса пользователей, который предполагает использование параллельных технологий при обмене информацией с узлами распределенной базы данных и словарем, а также позволяет увеличить время выполнения запроса, что в свою очередь увеличивает быстродействие системы в целом. Приведены актуальные на данный момент технологии хранения большого объема данных: параллельные и распределенные базы данных, парадигма MapReduce.

**Ключевые слова:** алгоритм, база данных, распределенная база данных, параллельная база данных, MapReduce, обработка запросов.

DOI: 10.22281/2413-9920-2018-04-02-211-217

В современном информационном мире одной из актуальных задач является управление большими потоками данных. Для управления таким объемом данных используют распределенную модель хранения данных – распределенную базу данных (РБД). Управление и получение данных в РБД осуществляется с помощью запросов, от скорости обработки которых зависит быстродействие системы в целом.

Скорость обработки запросов напрямую связана с методом организации связей между узлами РБД, сервером и словарем. В различных работах показаны схемы расположения узлов РБД в системе и методы увеличения эффективности выполнения запроса. Но в литературных источниках не представлено алгоритма организации выполнения запроса в системе и его обработка на различных стадиях развития: от отправки пользователем запроса на выполнение, обработки запроса системой РБД и до получения результата. Возникает вопрос о том, каким образом организовывать обработку запросов системой и как устанавливать связи между всеми ее составляющими.

Решение данного вопроса позволит устранить недостаток в комплексе «НЕКА» [1], который заключается в неоптимальной обработке запросов пользователей к БД. Комплекс «НЕКА» разрабатывался для исследования и построения РБД. Для устранения выявленного недостатка требуется разработ-

ка и внедрение алгоритма, позволяющего оптимально обрабатывать запросы пользователей к БД. Так как современный информационный мир опирается на парадигму параллельных вычислений, то актуальным является разработка алгоритма, который бы включал в себя применение параллельных технологий.

В работе была поставлена цель разработать универсальный алгоритм работы модуля обработки запросов с использованием технологии параллельных вычислений.

**Технологии хранения большого объема данных.** В [2] был проведен анализ актуальных в настоящее время технологий, используемых для управления и хранения больших объемов информации. Выделяют три основных направления:

- параллельные базы данных;
- распределенные базы данных;
- технология MapReduce.

Распределенная база данных (РБД) – это совокупность множества взаимосвязанных баз данных, распределенных в компьютерной сети. Система управления распределенной базой данных организует прозрачный доступ для пользователя, т.е. РБД для пользователя и прикладных программ выглядит как локальная БД. Узлы РБД, в свою очередь, логически представляют собой независимые компьютеры, которые могут иметь различное переменное окружение [3, 4].

Параллельная база данных – это БД, которая находится под управлением СУБД, реализованной для многопроцессорного компьютера. Подходы, применяемые при построении параллельной БД, направлены на более полное использование преимуществ конкретного мультипроцессора. Таким образом, параллельные БД являются аппаратно-программными комплексами [3, 5].

MapReduce – это модель распределённых вычислений, используемая для параллельных вычислений над большими наборами данных. Работа MapReduce состоит из двух основных шагов:

- Map – предварительная обработка и фильтрации исходных данных;
- Reduce – свертка обработанных данных и получение результата.

Преимущество MapReduce заключается в возможности распределенно производить операции предварительной обработки и свертки [6, 7]. Также на данный момент активно разрабатываются модели внедрения технологии MapReduce в традиционные реляционные СУБД [8] для сохранения преимуществ обеих парадигм.

Для увеличения производительности и обеспечения высокой надежности программного комплекса «НЕКА» было решено использовать технологию распределенного хранения данных на узлах и технологию параллельной обработки запроса, опирающуюся на парадигму MapReduce, на центральном узле.

**Методы и модели обработки запросов БД.** Методы и модели обработки запросов БД также, как и сами БД, разделяют на два направления: обработка запросов в реляционной и не реляционной БД. Первые из них представляют собой методы для строчных БД, а последние – для колоночных.

Строчным хранением данных называют физическое хранение кортежа, в котором значения атрибута идут последовательно друг за другом. В колоночных хранилищах значения одного атрибута хранятся последовательно друг за другом [9].

На данный момент оптимизации запросов БД посвящены множество работ. Например, в [9] говорится о том, что реализация планов выполнения запроса для строчных хранилищ базируется на трех базовых парадигмах:

- синхронный конвейер;
- итераторная модель;
- скобочный шаблон.

Эти же модели используются и для реализации планов выполнения запросов и для колоночных хранилищ данных, однако с некоторыми модификациями. Все данные модели используются при разработке СУБД.

В [10] рассматриваются основные алгоритмы лексической оптимизации времени выполнения запроса, которые опираются на тест самого запроса. Лексическая оптимизация запросов подразделяется на три вида:

- перезапись;
- украшения;
- сокращения.

Перезапись направлена на перемещение элементов по дереву запроса, украшение – внедрение в тест запроса дополнительных элементов для ускорения выполнения, сокращение – исключение избыточности в тексте запроса.

Локальный элиминационный алгоритм для обработки запросов в БД исследуется в [11]. Данный алгоритм заключается в построении пути выполнения запроса, где каждый следующий элемент множества сужает предыдущий и так до полного нахождения ответа на заданный запрос. Данная стратегия применяется для дискретных разреженных задач информатики и БД обладающих специальной структурой.

В [12] предложено описание процесса обслуживания запросов к РБД. Данный процесс выражается с помощью сетей массового обслуживания и иллюстрирует полный путь выполнения запроса от его отправки до получения результата. Также в [12] разработана модифицированная модель обработки запросов, в которой учитывается процесс репликации, влияющий на реактивность системы.

Средой для создания новых методов также является концепция MapReduce. В [13] рассматриваются методы обработки запросов к хранению данных по технологии MapReduce. Данные методы различаются количеством фаз Map-Shuffle-Reduce, форматом хранения записей, распределением данных по узлам и использованием ранней или поздней материализации результатов поиска.

После исследования вышеперечисленных работ было выяснено, что существующие

методы обработки запроса направлены либо на изменение кода самой СУБД, либо на оптимальное написание текста запроса. В рассматриваемых работах не было найдено универсального алгоритма, объясняющего функционирование запроса в РБД. Так же не оказалось метода или алгоритма, который бы позволял обрабатывать текст запроса для его дальнейшего выполнения в РБД. Таким образом, необходимо разработать алгоритм, который бы осуществлял все вышеперечисленное.

**Алгоритм обработки запросов пользователей к РБД.** Узким местом комплекса «НЕКА» является обработка запросов пользователей. На данном участке целесообразно применить параллельные технологии и парадигму MapReduce.

В любой реляционной СУБД существуют четыре вида операторов: *Insert*, *Delete*, *Update* и *Select*. Структура в общем случае у разных СУБД одинаковая. Самый интересный и трудоемкий оператор *Select*, так как в нем используются большое количество таблиц. Структура операций во всех реляционных СУБД одинакова и имеет вид:

*Insert into name\_table ...;*

*Delete from name\_table ...;*

*Update name\_table .....*

*Select ... from name\_table1, name\_table2, ..., name\_tableN ...;*

*Select ... from name\_table1 ... join name\_table2 ... join ... join name\_tableN on ...;*

У каждой СУБД есть зарезервированные ключевые слова, которые используются для определения, манипулирования и доступа к базам данных. Они являются частью синтаксиса того или иного подвида языка SQL и не могут быть использованы в качестве идентификаторов. Список ключевых слов можно посмотреть в официальной документации к СУБД [14-17].

Предлагаемый алгоритм обработки запросов пользователей к РБД заключается в обработке запроса пользователя посредством выделения вида операции, фильтрации слов запроса, параллельной обработки данных словаря и параллельной отправки узлам РБД подзапросов с последующим получением объединенного результата.

Работу данного алгоритма можно разделить на четыре этапа:

1. Модуль обработки запросов получает запрос от пользователя.

2. В зависимости от операции, используемой в запросе, выполняется выделение необходимых таблиц и их количества. Если используется операция *Insert*, *Update*, *Delete*, то определение используемой таблицы происходит сразу. Если же используется оператор *Select*, то выполняется промежуточная обработка операции, заключающаяся в выделении таблиц и их количества посредством фильтрации слов.

3. Модуль обработки запросов генерирует и отправляет запрос ко словарю, для выяснения информации о требуемых таблицах, и далее происходит рассылка подзапросов узлам РБД.

4. Получение отклика от узлов РБД и объединение информации с помощью модуля обработки запросов в результирующий ответ. Отправка выборки пользователю.

Подробные шаги алгоритма представлены в виде блок-схемы на рис. 1.

**Заключение.** В данной работе рассмотрены основные технологии хранения большого объема информации: параллельные и распределенные БД, парадигма MapReduce. Проведен анализ методов обработки запросов, в результате которого было выделено два основных направления в оптимизации времени их выполнения: методы, направленные на модификацию кода СУБД, и методы, направленные на сбалансированное написание текста запроса. Также существуют исследования, направленные на изучение методов обработки запросов посредством концепции MapReduce.

Для обслуживания запроса пользователя в РБД был разработан универсальный алгоритм, который позволяет обрабатывать текст запроса, производит обмен со словарем, а также с узлами РБД. Данный алгоритм предполагает использование параллельных технологий при обмене информацией со словарем и узлами РБД, что отвечает современным тенденциям развития информационных технологий, а также способствует улучшению производительности системы в целом.

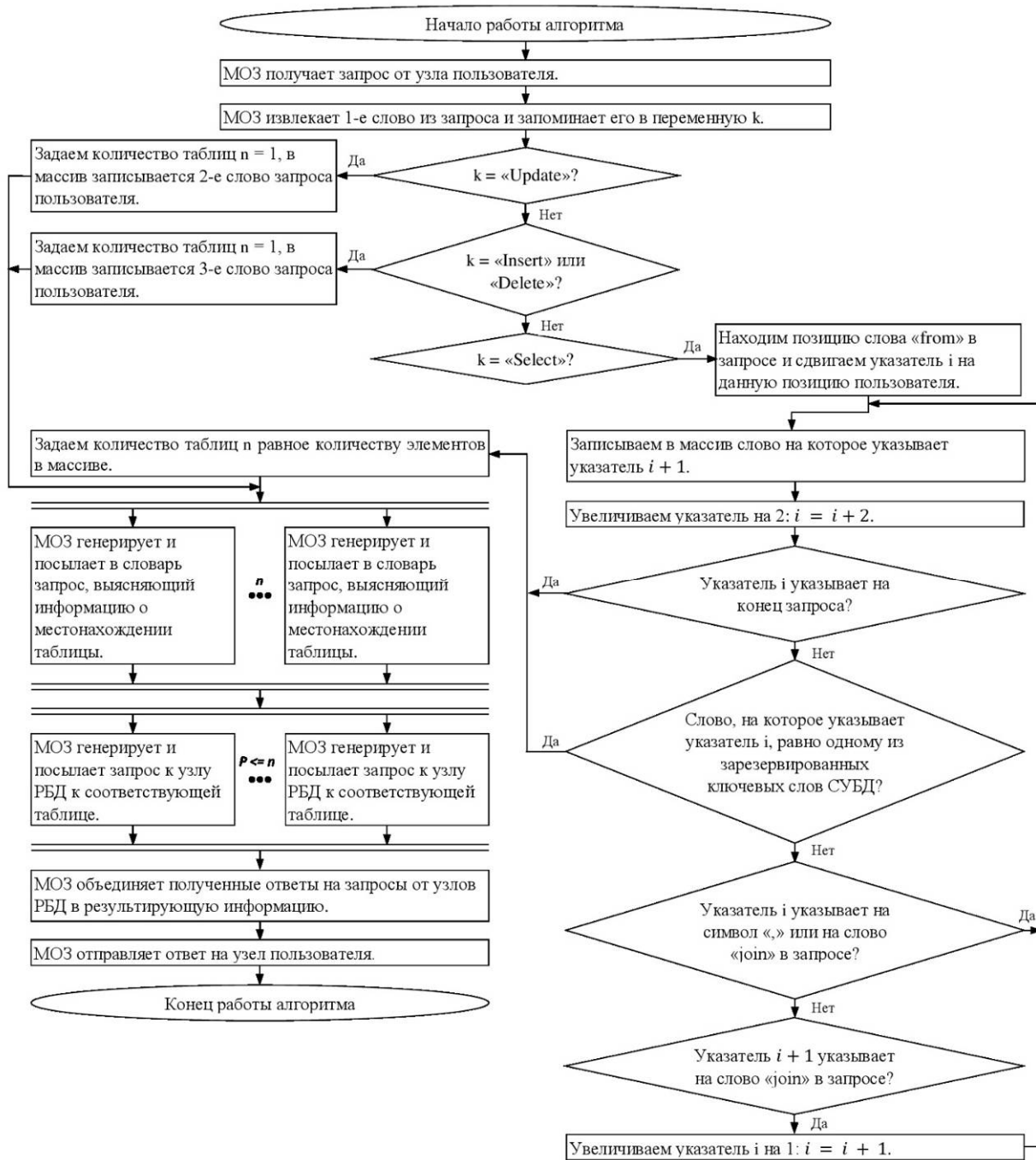


Рис. 1. Алгоритм обработки запросов пользователей к РБД

### Список литературы

1. Тимофеева, Н.Е. Программный комплекс для управления распределенной базой данных / Н.Е. Тимофеева, К.А. Полулях // Программные продукты и системы. – 2017. – Т. 30. – № 4. – С. 663–667.

2. Тимофеева, Н.Е. Анализ современных технологий хранения сверхбольших объемов информации / Н.Е. Тимофеева, К.А. Дмитриева, И.Д. Сагаева // Программные продукты, системы и алгоритмы. – 2018. – №1. – С.

14 – 19. – DOI: 10.15827/2311-6749.18.1.3 ISSN 2311-6749.

3. Тамер Оззу, М. Распределенные и параллельные системы баз данных / М. Тамер Оззу, П. Валдуриз // Системы Управления Базами Данных. – 1996. – №4.

4. Бойченко, А.В. Алгоритм динамического масштабирования реляционных баз данных в облачных средах / А.В. Бойченко, Д.К. Рогожин, Д.Г. Корнеев // Статистика и экономика. – 2014. – №6-2. – С. 461-465.

5. Девитт, Д. Параллельные системы баз данных: будущее высоко эффективных систем баз данных / Д. Девитт, Д. Грэй // Системы Управления Базами Данных. – 1995. – №2.

6. Гладкий, М.В. Модель распределенных вычислений MapReduce / М.В. Гладкий // Труды БГТУ. Серия 6: Физико-математические науки и информатика. – 2016. – №6 (188) – С. 194-198.

7. Янишевская, А.Г. Способы хранения и обработки большого объема данных с использованием MapReduce и Percona Server / А.Г. Янишевская, М.А. Чурсин // Инженерный вестник Дона. – 2015. – №2. - Ч.2.

8. Дергачев, А.А. Анализ данных на основе плат формы sql-mapreduce / А.А. Дергачев // Научно-технический вестник информационных технологий, механики и оптики. – 2014. – №1 (89). – С. 66-71.

9. Ермаков, Е.Ю. Анализ и сравнение процессов обработки запроса к таблице в параллельных колоночных и строчных хранилищах данных / Е.Ю. Ермаков // Современные информационные технологии и ИТ-образование. – 2012. – №8. – С. 545-555.

10. Мендкович, Н.А. Оценка эффективности минимизации ограничений запросов к СУБД / Н.А. Мендкович, С.Д. Кузнецов // Труды ИСП РАН. – 2013. – Т. 25. – С. 113-130.

11. Щербина, О.А. Локальные элиминационные алгоритмы обработки запросов в базах данных / О.А. Щербина // Таврический вестник информатики и математики. – 2009. – №2 (15). – С. 53-62.

12. Тараканов, О.В. Модифицированная модель обработки запросов в распределенных базах данных / О.В. Тараканов, В.А. Ду-

наев // Программные продукты и системы. – 2014. – №1 (105). – С. 70-76.

13. Григорьев, Ю.А. Сравнение методов обработки запросов к хранилищу данных по технологии MapReduce / Ю.А. Григорьев, В.А. Пролетарская // Информатика и системы управления. – 2016. – №1 (47). – С. 3-13.

14. MySQL 5.7 Reference Manual. Режим доступа: <https://dev.mysql.com/doc/refman/5.7/en/preface.html>.

15. Documentation PostgreSQL. Режим доступа: <https://www.postgresql.org/docs/>.

16. Документация по SQL Server. Режим доступа: <https://docs.microsoft.com/ru-ru/sql/sql-server/sql-server-technical-documentation>.

17. Database SQL Reference. Режим доступа: [https://docs.oracle.com/cd/B19306\\_01/server.102/b14200/](https://docs.oracle.com/cd/B19306_01/server.102/b14200/).

#### Об авторах

Тимофеева Надежда Евгеньевна – заведующая лабораторией теоретических проблем информатики и информационных технологий кафедры дискретной математики и информационных технологий Саратовского национального исследовательского государственного университета имени Н.Г. Чернышевского, [timofeevane@yandex.ru](mailto:timofeevane@yandex.ru).

Дмитриева Кристина Андреевна – программист лаборатории теоретических проблем информатики и информационных технологий кафедры дискретной математики и информационных технологий Саратовского национального исследовательского государственного университета имени Н.Г. Чернышевского, [krismail95@gmail.com](mailto:krismail95@gmail.com).

## UNIVERSAL ALGORITHM OF PROCESSING OF REQUESTS WITH USE OF PARALLEL TECHNOLOGY

Timofeeva N.E., Dmitrieva K.A.

Saratov National Research State University named after N.G. Chernyshevsky (Saratov, Russian Federation)

Processing and storage of a large amount of information is one of the difficult and interesting tasks at the moment. The performance of the system as a whole depends on how well the performance and reliability of the database are implemented. One of the most difficult aspects of this issue is the handling of a database query and its efficient execution. In this paper we consider modern methods and models of query processing in databases. We offer an algorithm to service the request of users, which involves the use of parallel technologies in the exchange of information with the nodes of a distributed database and a dictionary, and also allows to increase the query execution time, which in turn will increase the speed of the system as a whole. We bring the current at the moment the technology of storing large amounts of data: distributed and parallel databases, MapReduce.

**Key words:** *algorithm, database, distributed database, parallel database, MapReduce, query processing.*

**DOI:** 10.22281/2413-9920-2018-04-02-211-217

### References

1. Timofeeva N.E., Polulyakh K.A. A software complex to manage and test a distributed database, *Software & Systems*, 2017, Vol. 30, No. 4, pp. 663-667. (In Russian)
2. Timofeeva N.E., Dmitrieva K.A., Sa-gaeva I.D. Analiz sovremennykh tehnologiy khraneniya sverkhbolshikh obemov informatsii, *Programmnye produkty, sistemy i algoritmy*, 2018, No. 1, pp. 14-19. DOI: 10.15827/2311-6749.18.1.3 (In Russian)
3. Özsu M.T., Valduriez P. Distributed and parallel database systems, *Open Systems. DBMS*, 1996, No. 4. (In Russian)
4. Boychenko A.V., Rogozhin D.K., Korn-eev D.G. Algorithm for dynamic scaling relational database in clouds, *Statistics and Economics*, 2014, No. 6-2, pp. 461-465. (In Russian)
5. DeWitt D., Gray J. Parallel database systems: the future of high-performance database systems, *Communications of ACM*, 1992, Vol. 6, No. 35, pp. 85-98. (In Russian)
6. Gladkiy M.V. The model of distributed computing MapReduce, *Proceedings of BSTU. Physics and Mathematics. Informatics*, 2016. No. 6, pp. 194-198. (In Russian)
7. Yanishevskaya A.G., Chursin M.A. Methods for storing and processing large amounts of data using MapReduce and Percona, *Server. Engineering journal of Don*, 2015, No. 2, part 2. (In Russian)
8. Dergachev A. Data analysis by sql-mapreduce platform, *Scientific and Technical Journal of Information Technologies, Mechan-ics and Optics*, 2014, No.1, pp. 66-71. (In Russian)
9. Ermakov E.Yu. Analiz i sravnenie protsessov obrabotki zaprosa k tablitse v parallelnykh kolonochnykh i strochnykh khranilishchakh dannykh, *Modern Information Technologies and IT-Education*, 2012, No. 8, pp. 545-555. (In Russian)
10. Kuznetsov S.D., Mendkovich N.A. Minimization of data base query's conditions: evolution of efficiency, *Proceedings of ISP RAS*, 2013, No. 25, pp. 113-130. (In Russian)
11. Shcherbina O.A. Lokalnye eliminatsionnye algoritmy obrabotki zaprosov v bazakh dannykh. *Tavricheskiy vestnik informatiki i matematiki*, 2009, No.2, pp. 53-62. (In Russian)
12. Tarakanov O.V., Dunaev V.A. The modified query processing model in distributed databases. *Software & Systems*, 2014, No. 1, pp. 70-76. (In Russian)
13. Grigoryev Y.A., Proletarskaya V.A. Omparison of inquiry processings to acces data warehouse using the mapreduce technology, *Information Science and Control Systems*, 2016, No. 1 (47), pp 3-13. (In Russian)
14. MySQL 5.7 Reference Manual Available at: <https://dev.mysql.com/doc/refman/5.7/en/preface.html>.
15. Documentation PostgreSQL Available at: <https://www.postgresql.org/docs/>.
16. Документация по SQL Server Available at: <https://docs.microsoft.com/ru-ru/sql/sql-server/sql-server-technical-documentation>.
17. Database SQL Reference Available at: [https://docs.oracle.com/cd/B19306\\_01/server.102/b14200/](https://docs.oracle.com/cd/B19306_01/server.102/b14200/).

**Authors' information**

Nadezhda E. Timofeeva - Head of Laboratory theoretical problems of computer science and its applications of Department of Discrete Mathematics and Informational Technologies at Saratov National Research State University named after N.G. Chernyshevsky, *timofeevane@yandex.ru*;

Kristina Andreevna Dmitrieva - programmer of Laboratory theoretical problems of computer science and its applications of Department of Discrete Mathematics and Informational Technologies at Saratov National Research State University named after N.G. Chernyshevsky, *krismail95@gmail.com*.

Дата принятия к публикации  
(Date of acceptance for publication)  
17.04.2018

Дата публикации  
(Date of publication):  
25.06.2018

