|  |  |
| --- | --- |
|  | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ \_\_\_\_\_ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА \_\_\_\_\_СИСТЕМЫ ОБРАБОТКИ ИНФОРМАЦИИ И УПРАВЛЕНИЯ

**РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

***К НАУЧНО-ИССЛЕДОВАТЕЛЬСКОЙ РАБОТЕ***

***НА ТЕМУ:***

***\_\_\_\_Оценка времени выполнения SQL запросов***

***\_\_\_в озерах данных\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

***\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

***\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

***\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

Студент \_\_ИУ5-32М\_\_\_\_\_\_ **\_\_\_\_\_\_\_\_\_\_\_ Позняк А.А.**

(Группа) (Подпись, дата) (И.О.Фамилия)

Руководитель **\_\_\_\_\_\_\_\_\_\_\_\_\_** **\_****Ю.Е. Гапанюк**

(Подпись, дата) (И.О.Фамилия)

*2023 г.***Министерство науки и высшего образования Российской Федерации**

**Федеральное государственное бюджетное образовательное учреждение**

**высшего образования**

**«Московский государственный технический университет имени Н.Э. Баумана**

**(национальный исследовательский университет)»**

**(МГТУ им. Н.Э. Баумана)**

УТВЕРЖДАЮ

Заведующий кафедрой \_\_\_ИУ5\_\_\_\_

(Индекс)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_В.И. Терехов\_\_

(И.О.Фамилия)

«\_04\_» \_\_\_\_сентября\_\_\_\_\_\_\_ 2023 г.

**ЗАДАНИЕ**

**на выполнение научно-исследовательской работы**

по теме \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Студент группы \_ИУ5-32М\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Позняк Анастасия Александровна\_\_\_\_\_\_\_\_\_\_\_\_

(Фамилия, имя, отчество)

Направленность НИР (учебная, исследовательская, практическая, производственная, др.)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ИССЛЕДОВАТЕЛЬСКАЯ\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Источник тематики (кафедра, предприятие, НИР) \_\_\_\_\_КАФЕДРА\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

График выполнения НИР: 25% к \_\_\_\_ нед., 50% к \_\_\_\_ нед., 75% к \_\_\_ нед., 100% к \_\_\_\_ нед.

***Техническое задание*** \_\_\_подготовить математическую модель для оценки времени выполнения sql запросов в озерах данных

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

***Оформление научно-исследовательской работы:***

Расчетно-пояснительная записка на \_\_18\_ листах формата А4.

Перечень графического (иллюстративного) материала (чертежи, плакаты, слайды и т.п.)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата выдачи задания «\_04\_» \_\_\_\_сентября\_\_\_\_\_\_\_ 2023 г.

**Руководитель НИР**  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_Ю.Е. Гапанюк\_

(Подпись, дата) (И.О.Фамилия)

**Студент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_**А.А. Позняк\_\_\_\_

(Подпись, дата) (И.О.Фамилия)

Примечание: Задание оформляется в двух экземплярах: один выдается студенту, второй хранится на кафедре.

Оглавление

[Подсчет вероятности выполнения SLA 2](#_Toc154524791)

[Постановка проблемы 2](#_Toc154524792)

[Обзор способов решения 5](#_Toc154524793)

[Подготовка к проведению расчетов 9](#_Toc154524794)

[Подготовка экспериментальных значений 11](#_Toc154524795)

[Словарь терминов 12](#_Toc154524796)

[Источники 14](#_Toc154524797)

Постановка проблемы

Озера данных имеют многоуровневую структуру. В озере данных могут быть технические таблицы, которые используются для оптимизации хранения данных и вычислений. Например, озера данных архитектур “zone” или “pond”. Витрины верхнего уровня используются в процессах пользователей озера данных. Эти витрины обновляются либо в режиме реального времени (стриминговая или микробатчевая загрузка), либо единожды за определенный промежуток времени (батчевая загрузка). Во втором случае чаще всего используется обновление таблиц каждой ночью, так как нагрузка на вычислительные мощности от пользователей минимальна и их можно использовать для загрузки свежих данных.

Концепция озера данных предполагает, что пользователи самостоятельно работают с необходимыми им данными, создают на их основе отчеты, строят аналитику (см “self-service’). Очевидно, существует пользователи, которые ожидают обновление витрин не позднее определенного времени, так как от этого зависят их отчетность. В случае, если по какой-либо причине новые данные поступили в витрины с большой задержкой, может быть заторможена или остановлена работа целого отдела, возможны убытки для компании.

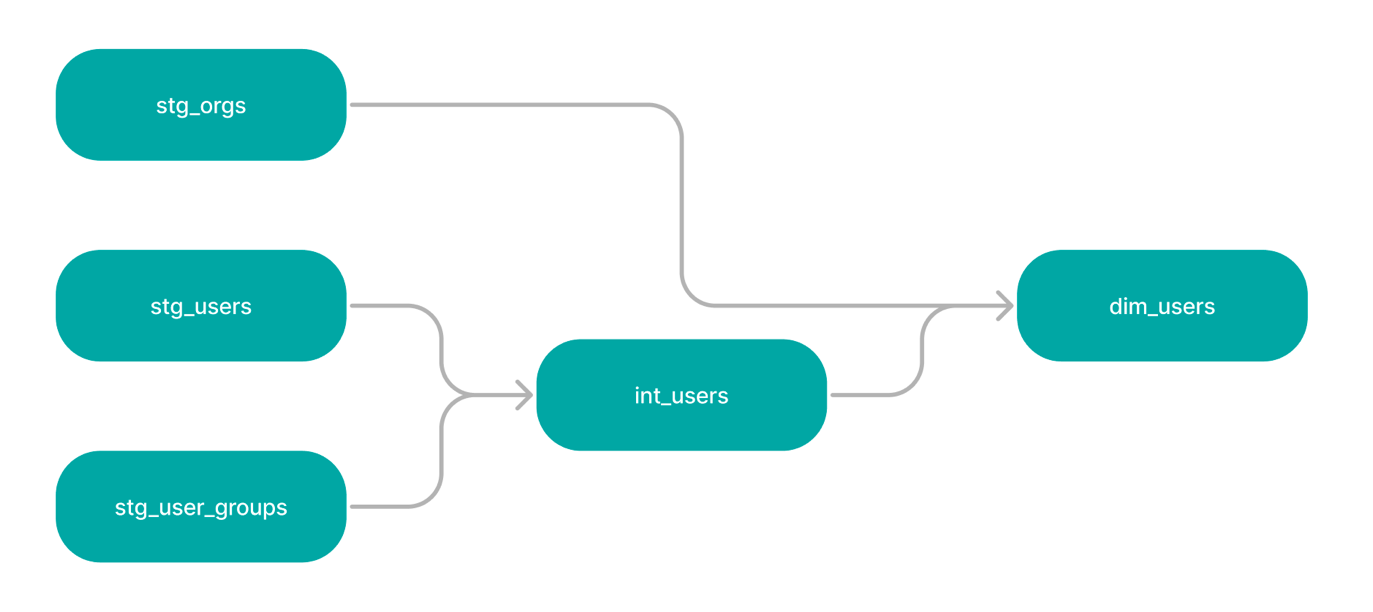
Ожидания пользователей по времени получения новых данных, согласованные с инженерами озера данных, закрепляются в виде договора об уровне оказываемых услуг (см. SLA). SLA может формулироваться следующим образом “80% таблиц должно быть обновлено к 8 утра”.

В данной работе проводится исследование способов оценки времени построения витрины при увеличении инкремента одной из таблиц-источников внутри озера данных.

Оценку времени построения витрины необходимо проводить в случае, если для этой витрины существует соглашение по уровню оказываемых услуг в виде максимального времени, за которое в таблице должны появиться новые данные, либо в случае, если аналитик считает, что увеличение времени построения витрины увеличит время обновления всего озера данных.

Ситуация, когда значительно увеличивается инкремент в одной из таблиц возможна при исправлении ошибок в данных. Например (см картинку):

1. Была доработка алгоритма дага таблицы stg\_users
2. При доработке была допущена ошибка, в результате часть данных не попадает в int\_users
3. Из-за этого и в int\_users, и в dim\_users некорректные данные.
4. Прошла неделя, прежде чем это было обнаружено
5. Ошибка в алгоритме была исправлена
6. Для исправления некорректных данных нужно пропустить через даги этих таблиц данные за неделю, то есть инкремент увеличивается приблизительно в 7 раз.



Из этого вытекает потребность в оценки времени выполнения SQL запросов.

Эргономика в данной работе связана с обеспечением эффективного управления временем построения витрины, обеспечением прозрачности и планирования, а также с управлением изменениями и соблюдением соглашений по уровню обслуживания:

1. Эффективность использования данных:
   * Оценка влияния времени построения витрины: Если увеличение инкремента в одной из таблиц ведет к заметному увеличению времени построения витрины, это может оказать влияние на быстроту доступа к актуальной информации. Анализ эффективности использования данных может помочь принять решение о том, насколько критично увеличение времени построения витрины.
2. Планирование и мониторинг:
   * Управление сроками: Оценка времени построения витрины важна для планирования обновлений озера данных и обеспечения соблюдения соглашений по уровню обслуживания. Адекватный мониторинг времени выполнения процесса поможет оперативно реагировать на возможные проблемы. Например, при необходимом, но долгом обновлении должны быть оповещены дежурные аналитики, которые могут исправить
3. Обучение и поддержка**:**
   * Документация и обучение*:* с точки зрения эргономики важно предоставить пользователям (в том числе аналитикам) информацию о влиянии увеличения инкремента на время построения витрины. Доступная и понятная документация, а также обучение пользователей с целью обеспечения прозрачности и понимания последствий увеличения времени построения важны для обеспечения удобства использования.
4. Управление изменениями:
   * Анализ влияния на весь охват данных: Анализ влияния увеличения времени построения витрины на обновление всего озера данных представляет собой важный аспект. Он помогает оценить, как изменения в одной таблице могут сказаться на целостности всей системы.
5. Безопасность:
   * Соблюдение соглашений по уровню обслуживания: Если существует соглашение по уровню обслуживания, связанное с максимальным временем построения витрины, эргономика включает в себя обеспечение соблюдения этих соглашений. Регулярный мониторинг и уведомления об опережении сроков важны для поддержания высокого уровня обслуживания.

Обзор способов решения

В работах [1] и [2] предлагается математический аппарат анализа времени выполнения запросов SELECT к распределённой базе данных. В работе [3] приведен пример использования такой математической модели.

Рассмотрим данную математическую модель. В ней время выполнения SQL-запроса оценивается с помощью преобразования Лапласа-Стилтьеса (ПЛС или преобразование Л.-С).

ПЛС функции распределения вероятностей случайной величины имеет следующий вид:

,

где - функция распределения вероятностей случайной величины .

Свойства ПЛС:

1. ,

где  - n-ая производная  при ,

 - n-ый начальный момент случайной величины ,  - математическое ожидание.

Благодаря этому свойству мы можем получить математическое ожидание времени выполнения SQL-запроса и его дисперсию:

2. Пусть  - сумма независимых случайных величин, тогда имеет место равенство

,

здесь  - преобразование Л.-С. случайной величины ,

 - преобразование Л.-С. случайной величины , .

Согласно преобразование Лапласа-Стилтьеса времени выполнения исходного запроса имеет следующий вид:

,

где  — это преобразование Лапласа-Стилтьеса времени выполнения подзапроса ,

 - преобразование Лапласа-Стилтьеса времени -го соединения промежуточных таблиц.

 - число подзапросов.

Для базы данных, не использующей индексы время выполнения подзапроса Qi определяется следующим выражением:

,

Где:

 - преобразование Л.-С. времени чтения одного блока таблицы ,

 - производящая функция читаемых блоков таблицы .

В общем случае, если записи, удовлетворяющие условию поиска , не располагаются в блоках таблицы  последовательно (в соседних строках), то производящую функцию читаемых блоков таблицы  можно найти с помощью следующей формулы:

,

где  - число участков записей; каждый участок включает записи, удовлетворяющие условию поиска  и располагающиеся в блоках таблицы  в соседних строках,

 - это функция, которая совпадает с (1), только здесь следует использовать производящую функцию  числа записей, удовлетворяющих условию поиска  и принадлежащих -ому участку, .

, (1)

здесь  определяется выражением (2),

- максимальное число записей в блоке таблицы 

Величину  в (28) можно оценить по формуле:

 (30)

Найдём производящую функцию числа записей таблицы , удовлетворяющих условию поиска :

, (2)

здесь  - вероятность, которая определяется с помощью описанной выше рекуррентной процедуры (табл. 3),

 - производящая функция числа записей в таблице .

Подготовка к проведению расчетов

Таким образом, необходимо получить следующие входные данные для проведения расчетов:

Для расчета времени выполнения подзапроса Qi:

* - преобразование Л.-С. времени чтения одного блока таблицы Ri
* Размер блока файла – необходимо для вычисления величины ri
* Средняя длина записи таблицы – необходимо для вычисления величины ri
*  - число участков записей. Каждый участок включает записи, удовлетворяющие условию поиска  и располагающиеся в блоках таблицы  в соседних строках
* n – число подзапросов
* - вероятность, что атрибут aij кортежа таблицы Ri принимает значение dijm (это элемент домена значений атрибута aij)
* Производящая функция числа строк в таблице

Для получения преобразование Л.-С. времени чтения одного блока таблицы  () необходимо проанализировать как реализуется чтение блока данных с диска SSD. Возможны два случая:

* Если это время пропорционально размеру блока, то время чтения блока определённой длины – это константа ti. И преобразование Л.-С. (ПЛС) определяется выражением.
* Если зависимость более сложная, то надо вывести функцию распределения вероятностей времени чтения блока F(t) и взять интеграл, в таком случае:

Размер блока файла можно узнать в документации СУБД, либо у администраторов СУБД, так как многие системы позволяют настраивать размер блока файла самостоятельно.

Данные о размерах блока (страницы) данных в различных СУБД представлены ниже (Таблица 1) [4], [5], [6].

Таблица 1

|  |  |
| --- | --- |
| СУБД | Размер блока |
| Postgres | 8 Кб |
| Oracle | 8 Кб |
| Grenplum | 32 Кб |

Среднюю длину записи таблицы можно найти как частное размера таблицы и количества записей таблицы. Данные о размере таблицы и количестве строк зачастую собираются утилитами сбора статистик СУБД. В противном случае можно выполнить SQL-запрос для получения этих данных. Такие запросы не являются тяжелыми и выполняются быстро.

Пример для СУБД Postgres и Grenplum:

Select count(\*) from <table\_name>;

SELECT pg\_size\_pretty(pg\_total\_relation\_size('<table\_name>;'));

Вероятность, что атрибут aij кортежа таблицы Ri принимает значение dijm можно получить из утилиты сбора статистик СУБД, либо собрать самостоятельно и задать априорно. Для сбора этих характеристик вручную можно использовать следующи SQL-запрос:

Select

<attribute\_name1>

,count(\*)

,’attribute\_name1’ as attribute

From <table\_name>

Group by 1

Union all

Select

<attribute\_name2>

,count(\*)

,’attribute\_name2’ as attribute

From <table\_name>

В приведенном запросе количество объединяемых с помощью оператора “union all” частей должно быть равно количеству атрибутов в таблице, либо количеству задействованных в таблице атрибутах.

Для вычисления производящей функции числа записей в таблице необходимо знать вероятности всех возможных значений количества строк в таблице. Посчитать вероятность, что на момент вычислений, количество строк в таблице соответствует определенному значению сложно, так как:

1. Количество записей ежедневно увеличивается, соответственно функция строго возрастает
2. Значения не повторяются
3. Значение n+1 зависит от значения n

Однако, так как мы рассматриваем оценку времени работы SQL запроса в контексте загрузки данных в озере данные, нужно учитывать механизмы инкрементальной загрузки, которые “обрезают” входную таблицу и фактически создают новую, с меньшим количеством данных.

Однако, аналитик может оценить диапазон возможных значений, которые поступят на вход для работы sql-запроса. Аналитик при этом учитывает:

1. Особенности механизма инкрементальной загрузки
2. Количество данных в первоначальной таблице
3. Количество данных, которые обычно загружаются в таблицу
4. И другие обстоятельства

Таким образом, мы можем посчитать производящую функцию числа строк в таблице, используя диапазон возможных значений. Для этого предположим, что значения в этом диапазоне равновероятны, X – число строк в таблице, значения a и b заданы аналитиком как минимальное и максимальное значения X:

1. Пусть у нас есть случайная величина *X*, принимающая значения в интервале [*a*,*b*], и случайная величина *R*, равномерно распределенная на отрезке [0,1].
2. Рассмотрим выражение *X*=*a*+*R*(*b*−*a*). Это линейное преобразование случайной величины *R* с коэффициентами (*b*−*a*) и *a*.  
   Заметим, что:

* Когда *R*=0, X=*a*.
* Когда *R*=1, *X*=*b*.
* Когда 0<*R*<1, *X* принимает значения в интервале между *a* и *b*.

1. Таким образом, X охватывает весь интервал [a,b] с равной вероятностью, и это представление подразумевает, что мы можем получить любое значение *X* в этом интервале с использованием равномерно распределенной случайной величины *R*.
2. Теперь, когда мы выражаем X через R, мы можем использовать это представление для вычисления производящей функции:
3. Подставим выражение для X в G(s):
4. Найдем математическое ожидание для . Это равно интегралу от 0 до 1 для sx по переменной x, умноженному на (b−a):
5. Таким образом, производящая функция:

Словарь терминов

Озеро данных – хранилище больших данных, в котором есть как структурированные, так и не структурированные данные.

Apach Airflow – инструмент для разработки ETL процессов. Он позволяет создавать, отслеживать и ставить на расписание процессы. В терминах системы эти процессы называются DAG (Directed Acyclic Graphs).

Даг (dag, directed acyclic graph) – ориентированный ациклический граф, в программе Apach Airflow это концептуальное представление серии действий с данными, конвейер данных (data pipeline). Иными словами – sql код построения таблицы.

Блок (страница) – пронумерованная часть области памяти хранения объекта БД фиксированного размера. [7]

SLA (англ. Service Level Agreement) – соглашение об уровне оказываемых услуг.

Витрина – таблица, которая используется для построения бизнес-отчетности. Противоположность – технические таблицы, к которым у пользователей доступа нет.

Инкремент – данные, которые поступают на вход дага. В большинстве систем реализована система инкрементальной загрузки данных, это означает, что на вход дагу поступают не все данные, а только те, новые и обновленные данные. После обработки этих данных алгоритмом дага, они будут вставлены в конечную таблицу.

"Self-service" – в области озер данных означает возможность пользователя самостоятельно проводить анализ данных, запрашивать информацию и выполнять необходимые действия без прямого участия IT-специалистов или других экспертов. Это может включать в себя создание отчетов, формирование запросов к данным, настройку визуализаций и другие операции, которые ранее могли потребовать специальных навыков или доступа к инструментам аналитики данных.

Источники

|  |  |
| --- | --- |
| [1] | Ю. А. Григорьев и А. Д. Плутенко, «Метод оценки времени выполнения запросов sql к реляционным базам данных,» *Радиоэлектроника. Информатика. Управление.,* т. 2, pp. 62-69, 2002. |
| [2] | Г. Ю. Александрович, «Оценка времени выполнения SQL-запросов к базам данных.,» *Наука и образование.,* pp. 1-19, 2012. |
| [3] | Ю. А. Григорьев и С. П. Остриков, «Оценка времени выполнения запросов с коррелированными подзапросами и операциями агрегирования,» *Вестник МГТУ им. Н.Э. Баумана. Сер. “Приборостроение”,* т. 1, pp. 99-110, 2006. |
| [4] | «Estimating Storage Capacity,» [В Интернете]. Available: https://docs.vmware.com/en/VMware-Greenplum/7/greenplum-database/install\_guide-capacity\_planning.html. [Дата обращения: 23 Декабрь 2023]. |
| [5] | «Настройка сервера. Изменение параметров,» [В Интернете]. Available: https://postgrespro.ru/docs/postgresql/9.4/config-setting. [Дата обращения: 23 Декабрь 2023]. |
| [6] | «Администрирование баз данных Oracle,» [В Интернете]. Available: https://oracle-dba.ru/docs/architecture/tablespaces/segments-extents-and-blocks/#:~:text=%D0%A0%D0%B0%D0%B7%D0%BC%D0%B5%D1%80%20%D0%B1%D0%BB%D0%BE%D0%BA%D0%B0%20%D1%81%D0%BB%D0%B5%D0%B4%D1%83%D0%B5%D1%82%20%D0%B2%D0%BE%D1%81%D0%BF%D1%80%D0%B8%D0%BD%D0%B8%. [Дата обращения: 23 Декабрь 2023]. |
| [7] | И. П. Карпова, Базы данных. Учебное пособие., Москва: Московский государственный институт электроники и математики, 2009. |
| [8] | W. Wu, Y. Chi, S. Zhu, J. Tatemura, H. Hacıgumus и J. F. Naughton, «Predicting Query Execution Time: Are Optimizer Cost Models Really Unusable?,» 2013. |