



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:

Разработка архитектуры для стенда по
тестированию производительности
мультипарадигмального озера данных,
использующего универсальную модель данных

Студент ИУ5-35М
(Группа)

Н.А. Клюкин
(Подпись, дата) (И.О.Фамилия)

Руководитель

Ю.Е. Гапанюк
(Подпись, дата) (И.О.Фамилия)

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 2024 г.

ЗАДАНИЕ
на выполнение научно-исследовательской работы

по теме _____

Студент группы ИУ5-35М

Клюкин Никита Александрович
(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 Разработать архитектуру для стенда по тестированию
производительности мультипарадигмального озера данных, использующего универсальную модель
данных, а также предложить методологию его тестирования

Оформление научно-исследовательской работы:

Расчетно-пояснительная записка на 22 листах формата А4.

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

Дата выдачи задания « 04 » сентября 2024 г.

Руководитель НИР

Ю.Е. Гапанюк
(Подпись, дата) (И.О.Фамилия)

Студент

Н.А. Клюкин
(Подпись, дата) (И.О.Фамилия)

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

Введение

Развитие области баз данных и их применение в реальных проектах требует глубокого понимания особенностей различных типов СУБД. С увеличением объемов данных и появлением новых технологий становится важным проводить анализ производительности и возможностей различных систем управления базами данных. Актуальность данного исследования вытекает из того, что необходимо создать методологию, которая будет максимально приближена к реальным условиям эксплуатации и позволит всесторонне протестировать производительность мультипарадигмального озера данных, использующего универсальную модель данных, а также будет отвечать международным стандартам.

В данной работе предлагается методология для проведения тестирования озера данных, использующего универсальную модель данных. За её основу предлагается взять тест ТРС-Н совета по эффективности обработки транзакций, который относится к классу тестов для систем поддержки принятия решений. Была изложена его основная суть и была представлена на его основе формальная запись тест-кейса. Дан алгоритм, по которому предлагается проводить тестирование, которое будет включать разогревочный этап для захвата системой всех необходимых ей ресурсов и само тестирование производительности, для которого будет произведено по переменное число измерений.

Глава 1. Методология тестирования производительности мультипарадигмального озера данных

В рамках первого эксперимента для оценки скорости было использовано попарно эквивалентные запросы трёх уровней сложности и с их помощью по модели сравнили скорость обработки данных на интегрированной платформе и на специализированной СУБД [1][2].

В дальнейших работах планируется использовать более продуманную методологию, способную всесторонне оценить производительность озера данных. В основу этой методологии планируется воспользоваться бенчмарком TPC-H.

Совет по эффективности обработки транзакций (TPC) — это некоммерческий консорциум, который определяет стандартные тесты для различных систем: транзакционные, системы поддержки принятия решений, системы виртуализации, системы Big Data, искусственный интеллект.

Из всего этого множества был выбран тест для систем поддержки принятия решений TPC-H, поскольку тесты для систем Big Data помимо аналитических запросов включают в себя применение алгоритмов машинного обучения, что не подходит для нас, так как мы стремимся проверить именно скорость выполнения аналитических запросов разной сложности [3][4].

TPC-H — это тест для поддержки принятия решений, который состоит из набора бизнес-ориентированных специальных запросов, модель данных которых и рабочая нагрузка на запросы достаточно сложны, чтобы выполнять разумный набор аналитических задач [5]. Первая версия была выпущена в 1999 году и обновлялась на протяжении многих лет.

Запросы, которые фигурируют в данном бенчмарке отличаются высокой степенью сложности, использованием разными способами доступа и проверяют большой процент доступных данных, что делает их отличным ориентиром при подготовке аналогичных запросов к МАКГ [6][7] (рис. 3).

Query	Avg	Count	Min/Max	Sum	From	Group by	Oder by	Nested queries
Q1	3	1		4	1	•	•	
Q2			1		5		•	•
Q3				1	3	•	•	
Q4		1		2		•	•	•
Q5				1	6	•	•	
Q6				1	1			
Q7				1	5	•	•	•
Q8				3	7	•	•	•
Q9				1	6	•	•	•
Q10				1	4	•	•	
Q11				3	3	•	•	•
Q12				2	2	•	•	
Q13		1			2	•	•	•
Q14				2	2			
Q15			1		2	•	•	•
Q16		1			3	•	•	•
Q17	1			1	2			•
Q18				2	3	•	•	•
Q19				1	2			
Q20				1	4		•	•
Q21		1			4	•	•	•
Q22		1		1	2	•	•	•

Рис. 1. Описание запросов TPC-H [5]

Структурно, описание каждого запроса выглядит следующим образом:

1. Выделяется бизнес-вопрос, на который необходимо найти ответ;
2. Задается функциональный запрос на языке SQL;
3. Описываются параметры подстановки с правилами, как генерировать для них значения;
4. Запрос проверяется на соответствие посредством подстановки в него конкретных значений и сверкой его результата с контрольными значениями из базы тестов.

Стратегически, алгоритм тестирования озера данных, использующего универсальную модель данных выглядит следующим образом:

1. Производим парсинг исходных значений из МАКГ в реляционную базу данных. Она будет выступать в качестве промежуточного стенда из которого будет производиться заполнение тестовых стендов;
2. Инициализируем, настраиваем и подгружаем данные в тестовые стенды: само озеро данных с универсальной моделью, стенды с

реляционной (PostgreSQL), графовой (Neo4j), многомерной (Pentaho BI) моделями данных;

3. Производим тестирование, что включает в себе выполнение "разогревочного этапа" и само тестирование производительности;

3.1. Под "разогревочным этапом" подразумевается запуски, необходимые для системы, чтобы захватить всю нужную информацию;

3.2. Под тестом производительности подразумевается выполнение одного из тестовых запросов;

4. Провести анализ результатов и сделать выводы.

Для каждого тест-кейса будет проведено по переменное число измерений. Поскольку мы работаем с СУБД, а не с системами реального времени, полученные значения могут существенно отличаться. Для работы с такими данными, будем рассматривать результирующее значение как случайную величину.

Тогда, чтобы определить репрезентативное количество измерений, будем проводить измерения итеративно, а в качестве критерия их остановки будем использовать t-критерий Стьюдента [1]. Опишем этот процесс подробнее.

Зададим n_i как число элементов на i -ой итерации. Тогда на каждом шаге итерации будем проверять гипотезу о равенстве средних выборки полученной на i -ой итерации и $(i - 1)$ -ой итерации. Уровень значимости зададим равным 0,05. А число степеней свободы будет, соответственно $n_i + n_{i-1} - 2$. Тогда сравним рассчитанное значение t-критерия по формуле (1) с табличным значением, полученным на основании числа степеней свободы и уровня значимости. Если полученное значение по модулю больше или равно табличного значения критерия, значит в данных имеются сильные колебания, и необходимо дополнительное измерение (следующая итерация), иначе заканчиваем измерения.

$$t = \frac{|\overline{x_{i-1}} - \overline{x_i}|}{\sqrt{\frac{\sigma_{i-1}^2}{n_{i-1}} + \frac{\sigma_i^2}{n_i}}} \quad (1)$$

где $\overline{x_{i-1}}$ – среднее значение выборки, полученной на $(i - 1)$ -ой итерации; $\overline{x_i}$ – среднее значение выборки, полученной на i -ой итерации; σ_{i-1} – среднее квадратичное отклонение выборки, полученной на $(i - 1)$ -ой итерации; σ_i – среднее квадратичное отклонение выборки, полученной на i -ой итерации; n_{i-1} – размер выборки, полученной на $(i - 1)$ -ой итерации; n_i – размер выборки, полученной на i -ой итерации.

Также каждое измерение будет проводиться на “разогретой” системе, т.е. сначала запрос будет запускаться два-три раза без фиксации времени исполнения, для того чтобы система успела захватить все необходимые ресурсы. После чего уже выполняется измерение времени обработки запроса.

С помощью такого подхода будут получены оценки производительности частных моделей данных в архиграфовом озере данных и соответствующих независимых СУБД.

Далее будет проведено сравнение полученных средних значений производительности двух упомянутых систем управления данными. Также будет оценена статистическая значимость результатов с помощью проверки статистической гипотезы о неравенстве полученных средних на основании того же t -критерия Стьюдента (1)[3].

Глава 2. Запросы для тестирования архиграфового озера данных

В результате анализа была составлена оптимальная структура базы данных для тестирования озера данных. Для удобства восприятия были выделены следующие префиксы для таблиц:

- “f_” – таблица фактов
- “m_” – таблица для отображения связей многие ко многим
- “d_” – справочные таблицы

Префикс “f_” вполне мог быть назначен и многим объектам, например для таблицы “авторы”, однако, поскольку ключевым объектом в МАКГ является “статья”, было принято решение использовать этот префикс только для него (рис. 2).

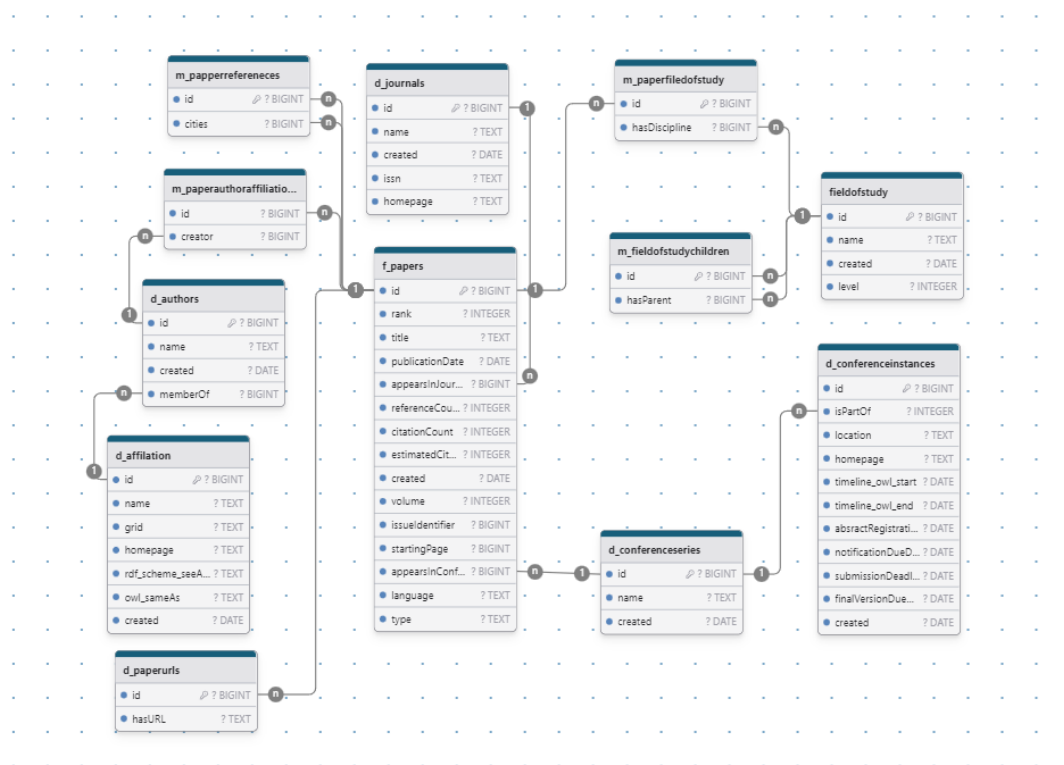


Рис. 2. Схема базы данных [составлено автором по 2]

В итоге было получено 22 бизнес-запроса, которые были разбиты на 3 группы по сложности. К простым относятся 6 запросов, к средним по сложности – 11, к сложным – 5 (рис. 3).

Запрос	AVG	Count	Min/Max	Sum	From	Group By	Order By	Nested queries	Сложность	Группа
Q1	4	1	0	3	4	+	+	-	14	Сложные
Q2					1		+		2	Простые
Q3		3	1		3	+	+	+	11	Сложные
Q4	2				4	+	+		8	Сложные
Q5		1			2	+	+		5	Средние
Q6				1	1				2	Простые
Q7	1				1	+			3	Простые
Q8		4			1				5	Средние
Q9		1			3	+	+		6	Средние
Q10		1			5	+	+		8	Сложные
Q11				1	3	+	+		6	Средние
Q12					1		+		2	Простые
Q13			1		3			+	6	Средние
Q14	1				4			+	7	Средние
Q15		3	1		6	+		+	13	Сложные
Q16		1			4	+	+		7	Средние
Q17					2		+		3	Простые
Q18				1	3	+	+		6	Средние
Q19					1		+		2	Простые
Q20	1				2	+	+		5	Средние
Q21		1			2	+	+		5	Средние
Q22	1				3	+	+		6	Средние

Рис. 3. Сложность запросов

Ниже представлены 22 запроса, оформленные по TPC-H.

Запрос “Суммарный отчёт по статьям” (Q1)

Пункт	Описание
Бизнес вопрос	Этот запрос предоставляет статистику в разрезе типов и языков статей за определенный период.
Функциональный запрос	<pre> SELECT type, language, AVG(rank) AS avg_rank, AVG(referenceCount) AS avg_refCount, AVG(citationCount) AS avg_citCount, AVG(estimatedCitationCount) AS avg_estCitCount, SUM(referenceCount) AS sum_refCount, SUM(citationCount) AS sum_citCount, SUM(estimatedCitationCount) AS sum_estCitCount, COUNT(*) AS count_paper FROM papers WHERE publicationDate >= '1910-06-01'::DATE - ('[DELTA]' ' day')::INTERVAL GROUP BY type, language ORDER BY type, language </pre>

Параметры подстановки	DELTA это значение в диапазоне [60. 120]
Валидация результатов	DELTA = 90

Запрос “Наиболее актуальные статьи” (Q2)

Пункт	Описание
Бизнес вопрос	Найдите наиболее релевантную статью в определенной области исследований для данной серии конференций.
Функциональный запрос	<pre> SELECT f_papers.id AS paper_id, f_papers.title, f_papers.publicationDate, d_conferenceseries."name" AS conference_name, f_papers.citationCount, f_papers.referenceCount FROM f_papers JOIN d_conferenceseries ON f_papers.appearsInConferenceSeries = d_conferenceseries.id JOIN m_paperfiledofstudy ON f_papers.id = m_paperfiledofstudy.id JOIN fieldofstudy ON m_paperfiledofstudy.hasDiscipline = fieldofstudy.id WHERE fieldofstudy."name" = '[FIELD_NAME]' AND d_conferenceseries."name" = '[CONFERENCE_SERIES_NAME]' ORDER BY f_papers.citationCount DESC, f_papers.referenceCount DESC </pre>
Параметры подстановки	FIELD_NAME – это название области исследований CONFERENCE_SERIES_NAME – это серия конференций
Валидация результатов	FIELD_NAME = CONFERENCE_SERIES_NAME =

Запрос “Журналы с максимальным количеством статей в каждом году” (Q3)

Пункт	Описание
Бизнес вопрос	Найти список журналов с максимальным количеством статей, опубликованных в каждом году.
Функциональный запрос	<pre> SELECT j.name AS journal_name, EXTRACT(YEAR FROM p.publicationDate) AS publication_year, COUNT(p.id) AS paper_count </pre>

	<pre> FROM d_journals j JOIN f_papers p ON j.id = p.appearsInJournal WHERE p.publicationDate IS NOT NULL AND p.publicationDate BETWEEN '[START_DATE]' AND '[END_DATE]' GROUP BY j.name, EXTRACT(YEAR FROM p.publicationDate) HAVING COUNT(p.id) = (SELECT MAX(count_per_year) FROM (SELECT COUNT(p_inner.id) AS count_per_year FROM f_papers p_inner WHERE EXTRACT(YEAR FROM p_inner.publicationDate) = EXTRACT(YEAR FROM p.publicationDate) GROUP BY p_inner.appearsInJournal) subquery) ORDER BY publication_year, paper_count DESC; </pre>
Параметры подстановки	<p>START_DATE – 1 января N года</p> <p>END_DATE – 31 декабря Z года, при этом N<=Z</p>
Валидация результатов	<p>START_DATE="01-01-2000"</p> <p>END_DATE="31-12-2005"</p>

Запрос “Популярные авторы по цитированию их статей” (Q4)

Пункт	Описание
Бизнес вопрос	Найдите авторов, у которых среднее число цитирований статей превышает [CITATION_CNT]
Функциональный запрос	<pre> SELECT a.name AS author_name, af.name AS affiliation_name, AVG(p.citationCount) AS avg_citation_count FROM d_authors a JOIN m_paperauthoraffiliations paa ON a.id = paa.creator JOIN f_papers p ON paa.id = p.id JOIN d_affiliation af ON a.memberOf = af.id WHERE </pre>

	<p>p.citationCount IS NOT NULL</p> <p>GROUP BY</p> <p>a.name, af.name</p> <p>HAVING</p> <p>AVG(p.citationCount) > [CITATION_CNT]</p> <p>ORDER BY</p> <p>avg_citation_count DESC;</p>
Параметры подстановки	CITATION_CNT это количество цитирования
Валидация результатов	CITATION_CNT=10

Запрос “Лучшие конференции за определенный год” (Q5)

Пункт	Описание
Бизнес вопрос	Найдите лучшие конференции с наибольшим количеством представленных докладов за определенный год.
Функциональный запрос	<pre> SELECT d_conference series."name" AS conference_name, COUNT(f_papers.id) AS paper_count FROM f_papers JOIN d_conferenceseries ON f_papers.appearsInConferenceSeries = d_conferenceseries.id WHERE EXTRACT(YEAR FROM f_papers.publicationDate) = [YEAR] GROUP BY d_conferenceseries."name" ORDER BY paper_count DESC; </pre>
Параметры подстановки	YEAR – это отчётный год
Валидация результатов	YEAR = 2000

Запрос “Общее количество цитирований статей” (Q6)

Пункт	Описание
Бизнес вопрос	Рассчитайте общее количество цитирований статей, опубликованных за определенный промежуток времени.
Функциональный запрос	<pre> SELECT SUM(f_papers.citationCount) AS total_citations FROM f_papers WHERE f_papers.publicationDate BETWEEN '[START_DATE]' AND '[END_DATE]'; </pre>
Параметры подстановки	<p>START_DATE – дата начала отчётного периода</p> <p>END_DATE – дата окончания отчётного периода, при этом [START_DATE] <= [END_DATE]</p>
Валидация результатов	<p>START_DATE="01-01-2000"</p> <p>END_DATE="01-01-2002"</p>

Запрос “Среднее количество цитирований статей” (Q7)

Пункт	Описание
Бизнес вопрос	Определите среднее количество цитирований статей, сгруппированных по типу публикации (журнал или конференция).
Функциональный запрос	<pre> SELECT CASE WHEN f_papers.appearsInJournal IS NOT NULL THEN 'Journal' WHEN f_papers.appearsInConferenceSeries IS NOT NULL THEN 'Conference' END AS publication_type, AVG(f_papers.citationCount) AS avg_citation_count FROM f_papers GROUP BY publication_type; </pre>
Параметры подстановки	
Валидация результатов	Количество строк в результирующем наборе

Запрос “Доля статей в журналах и сборниках” (Q8)

Пункт	Описание
Бизнес вопрос	Найдите долю статей, опубликованных в журналах, в сравнении с долей опубликованных в сборниках конференций за определенный год.
Функциональный запрос	<pre> SELECT COUNT(CASE WHEN f_papers.appearsInJournal IS NOT NULL THEN 1 END) * 100.0 / COUNT(*) AS journal_share, COUNT(CASE WHEN f_papers.appearsInConferenceSeries IS NOT NULL THEN 1 END) * 100.0 / COUNT(*) AS conference_share FROM f_papers WHERE EXTRACT(YEAR FROM f_papers.publicationDate) = [YEAR]; </pre>
Параметры подстановки	YEAR – это отчётный год
Валидация результатов	YEAR = 2000

Запрос “Области исследований с наибольшим количеством опубликованных работ” (Q9)

Пункт	Описание
Бизнес вопрос	Найдите области исследований с наибольшим количеством опубликованных работ.
Функциональный запрос	SELECT

	fieldofstudy."name" AS field_name, COUNT(m_paperfiledofstudy.id) AS paper_count FROM fieldofstudy JOIN m_paperfiledofstudy ON fieldofstudy.id = m_paperfiledofstudy.hasDiscipline GROUP BY fieldofstudy."name" ORDER BY paper_count DESC
Параметры подстановки	
Валидация результатов	Количество строк в результирующем наборе

Запрос “Топ авторов с наибольшим количеством публикаций в определенной области исследований” (Q10)

Пункт	Описание
Бизнес вопрос	Найдите топ авторов с наибольшим количеством публикаций в определенной области исследований
Функциональный запрос	SELECT d_authors."name" AS author_name, COUNT(m_paperauthoraffiliations.id) AS publication_count FROM d_authors JOIN m_paperauthoraffiliations ON d_authors.id = m_paperauthoraffiliations.creator JOIN f_papers ON m_paperauthoraffiliations.id = f_papers.id JOIN m_paperfiledofstudy ON f_papers.id = m_paperfiledofstudy.id JOIN fieldofstudy ON m_paperfiledofstudy.hasDiscipline = fieldofstudy.id WHERE fieldofstudy."name" = '[FIELD_NAME]' GROUP BY d_authors."name" ORDER BY publication_count DESC
Параметры подстановки	FIELD_NAME – это название области исследований
Валидация результатов	FIELD_NAME=

Запрос “Авторы с наибольшим количеством цитирований” (Q11)

Пункт	Описание
Бизнес вопрос	Найдите авторов с наибольшим количеством цитирований во всех их статьях.

Функциональный запрос	SELECT d_authors."name" AS author_name, SUM(f_papers.citationCount) AS total_citations FROM d_authors JOIN m_paperauthoraffiliations ON d_authors.id = m_paperauthoraffiliations.creator JOIN f_papers ON m_paperauthoraffiliations.id = f_papers.id GROUP BY d_authors."name" ORDER BY total_citations DESC
Параметры подстановки	
Валидация результатов	Количество строк в результирующем наборе

Запрос “Статьи с самым высоким рейтингом за определенный год” (Q12)

Пункт	Описание
Бизнес вопрос	Найдите статьи с самым высоким рейтингом за определенный год.
Функциональный запрос	SELECT f_papers.id AS paper_id, f_papers.title, f_papers.rank FROM f_papers WHERE EXTRACT(YEAR FROM f_papers.publicationDate) = [YEAR] ORDER BY f_papers.rank DESC
Параметры подстановки	YEAR – это отчётный год
Валидация результатов	YEAR = 2000

Запрос “Статьи, опубликованные в новых журналах” (Q13)

Пункт	Описание
Бизнес вопрос	Найти статьи, опубликованные в журналах, которые были созданы позже всех
Функциональный запрос	SELECT p.id AS paper_id, p.title AS paper_title, j."name" AS journal_name, j.created AS journal_created_date FROM f_papers p JOIN d_journals j ON p.appearsInJournal = j.id

	WHERE j.created = (SELECT MAX(created) FROM d_journals);
Параметры подстановки	
Валидация результатов	Количество строк в результирующем наборе

Запрос “Авторы, которые опубликовали статьи с количеством цитирований выше среднего” (Q14)

Пункт	Описание
Бизнес вопрос	Найти авторов, которые опубликовали статьи с количеством цитирований выше среднего
Функциональный запрос	<pre> SELECT a.id AS author_id, a."name" AS author_name, p.title AS paper_title, p.citationCount AS citation_count FROM d_authors a JOIN m_paperauthoraffiliations pa ON a.id = pa.creator JOIN f_papers p ON pa.id = p.id WHERE p.citationCount > (SELECT AVG(citationCount) FROM f_papers); </pre>
Параметры подстановки	
Валидация результатов	Количество строк в результирующем наборе

Запрос “Конференции, с наибольшим количеством статей” (Q15)

Пункт	Описание
Бизнес вопрос	Найти конференции, в которых было опубликовано больше всего статей
Функциональный запрос	<pre> SELECT cs.id AS conference_series_id, cs."name" AS conference_series_name, COUNT(p.id) AS paper_count FROM d_conferenceseries cs JOIN d_conferenceinstances ci ON cs.id = ci.isPartOf JOIN f_papers p ON ci.id = p.appearsInConferenceSeries GROUP BY cs.id, cs."name" HAVING COUNT(p.id) = (SELECT MAX(paper_count) FROM (</pre>

	SELECT COUNT(p.id) AS paper_count FROM d_conferenceseries cs JOIN d_conferenceinstances ci ON cs.id = ci.isPartOf JOIN f_papers p ON ci.id = p.appearsInConferenceSeries GROUP BY cs.id) AS subquery);
Параметры подстановки	
Валидация результатов	Количество строк в результирующем наборе

Запрос “Отчёт по соавторству” (Q16)

Пункт	Описание
Бизнес вопрос	Определите авторов, которые совместно написали больше всего статей.
Функциональный запрос	SELECT a1."name" AS author1, a2."name" AS author2, COUNT(*) AS coauthored_papers FROM m_paperauthoraffiliations AS paa1 JOIN m_paperauthoraffiliations AS paa2 ON paa1.id = paa2.id AND paa1.creator < paa2.creator JOIN d_authors AS a1 ON paa1.creator = a1.id JOIN d_authors AS a2 ON paa2.creator = a2.id GROUP BY a1."name", a2."name" ORDER BY coauthored_papers DESC LIMIT 10;
Параметры подстановки	
Валидация результатов	Количество строк в результирующем наборе

Запрос “Наиболее цитируемый доклад в определенной серии конференций” (Q17)

Пункт	Описание
Бизнес вопрос	Найдите наиболее цитируемые доклады в определенной серии конференций.
Функциональный запрос	SELECT f_papers.id AS paper_id,

	<pre> f_papers.title, f_papers.citationCount FROM f_papers JOIN d_conferenceseries ON f_papers.appearsInConferenceSeries = d_conferenceseries.id WHERE d_conferenceseries."name" = '[CONFERENCE_SERIES_NAME]' ORDER BY f_papers.citationCount DESC LIMIT 1; </pre>
Параметры подстановки	CONFERENCE_SERIES_NAME – это серия конференций
Валидация результатов	CONFERENCE_SERIES_NAME=

Запрос “Области исследования с наибольшим количеством цитирований в целом” (Q18)

Пункт	Описание
Бизнес вопрос	Определите области исследования с наибольшим количеством цитирований в целом.
Функциональный запрос	<pre> SELECT fieldofstudy."name" AS field_name, SUM(f_papers.citationCount) AS total_citations FROM fieldofstudy JOIN m_paperfiledofstudy ON fieldofstudy.id = m_paperfiledofstudy.hasDiscipline JOIN f_papers ON m_paperfiledofstudy.id = f_papers.id GROUP BY fieldofstudy."name" ORDER BY total_citations DESC </pre>
Параметры подстановки	
Валидация результатов	Количество строк в результирующем наборе

Запрос “Статьи с наибольшим количеством ссылок за определенный год” (Q19)

Пункт	Описание
Бизнес вопрос	Найдите статьи с наибольшим количеством ссылок за определенный год.
Функциональный запрос	<pre> SELECT f_papers.id AS paper_id, f_papers.title, f_papers.referenceCount </pre>

	FROM f_papers WHERE EXTRACT(YEAR FROM f_papers.publicationDate) = [YEAR] ORDER BY f_papers.referenceCount DESC LIMIT 5;
Параметры подстановки	YEAR – это отчётный год
Валидация результатов	YEAR = 2000

Запрос “Лучшие конференции с самым высоким средним количеством цитирований на одну статью” (Q20)

Пункт	Описание
Бизнес вопрос	Определите лучшие конференции с самым высоким средним количеством цитирований на одну статью.
Функциональный запрос	SELECT d_conferenceseries."name" AS conference_name, AVG(f_papers.citationCount) AS avg_citations FROM d_conferenceseries JOIN f_papers ON d_conferenceseries.id = f_papers.appearsInConferenceSeries GROUP BY d_conferenceseries."name" ORDER BY avg_citations DESC
Параметры подстановки	
Валидация результатов	Количество строк в результирующем наборе

Запрос “Топ журналов по количеству статей” (Q21)

Пункт	Описание
Бизнес вопрос	Найдите топ журналов, в которых опубликовано больше всего статей.
Функциональный запрос	SELECT d_journals."name" AS journal_name, COUNT(f_papers.id) AS paper_count FROM d_journals JOIN f_papers ON d_journals.id = f_papers.appearsInJournal GROUP BY d_journals."name" ORDER BY paper_count DESC
Параметры подстановки	

Валидация результатов	Количество строк в результирующем наборе
-----------------------	--

Запрос “Топ авторов с самым высоким средним рейтингом” (Q22)

Пункт	Описание
Бизнес вопрос	Определите авторов, чьи статьи имеют самый высокий средний рейтинг.
Функциональный запрос	<pre> SELECT d_authors."name" AS author_name, AVG(f_papers.rank) AS avg_rank FROM d_authors JOIN m_paperauthoraffiliations ON d_authors.id = m_paperauthoraffiliations.creator JOIN f_papers ON m_paperauthoraffiliations.id = f_papers.id GROUP BY d_authors."name" ORDER BY avg_rank ASC </pre>
Параметры подстановки	
Валидация результатов	Количество строк в результирующем наборе

Вывод

Таким образом, в данном исследовании была предложена методология для тестирования озера данных. За её основу был взят тест TPC-H, поскольку тесты TPC для систем Big Data помимо аналитических запросов включают в себя применение алгоритмов машинного обучения, что не подходит для нас, так как мы стремимся проверить именно скорость выполнения аналитических запросов разной сложности. Была представлена формальная запись для тестовых кейсов и разработан алгоритм тестирования, который включает в себя парсинг исходных значений из Microsoft Academic Knowledge Graph с дальнейшей автоматической погрузкой данных в нужном объёме в тестовые стенды, проведение тестового кейса, который включает в себя прогревочный этап, заключающийся в запуске запроса 2-3 раза без фиксации времени для захвата системой всех необходимых ей ресурсов, и само тестирование производительности, для которого будет произведено по переменное число измерений, где для определения репрезентативного количества измерений, будем проводить измерения итеративно, а в качестве критерия их остановки будем использовать t-критерий Стьюдента.

Список литературы

1. Sukhobokov A., Gapanyuk Y., Vetoshkin A., Mironova A., Klyukin N., Afanasev R., Lakhvich D. Universal Data Model as a Way to Build Multi-paradigm Data Lakes // ICBDA. 2024. P. 203–212. DOI 10.1109/ICBDA61153.2024.10607189.
2. Сухобоков А.А., Афанасьев Р.А. Первая стадия эксперимента по оценке производительности мультипарадигмальных озёр данных // Естественные и технические науки. 2023. № 7. С. 124–133. DOI 10.25633/ETN.2023.07.08.
3. TDWI – meet TPCx-BB -- A Benchmark for Assessing Big Data Performance. URL: <https://tdwi.org/articles/2016/06/28/tpcx-bb-big-data-benchmark.aspx> (дата обращения: 15.10.2024).
4. TPCX-BB_v1. URL: https://www.tpc.org/TPC_Documents_Current_Versions/pdf/TPCX-BB_v1.6.2.pdf (дата обращения: 15.10.2024).
5. TPC. URL: https://www.tpc.org/TPC_Documents_Current_Versions/pdf/TPC-H_v3.0.1.pdf (дата обращения: 15.10.2024).
6. Faerber Michael. The Microsoft Academic Knowledge Graph: A Linked Data Source with 8 Billion Triples of Scholarly Data // International Semantic Web Conference (ISWC). 2019. P. 113–129. DOI 10.1007/978-3-030-30796-7_8.
7. Faerber Michael. The Microsoft Academic Knowledge Graph enhanced: Author name disambiguation, publication classification, and embeddings // Quantitative Science Studies. 2022. Vol. 3. № 1. P. 51–98. DOI 10.1162/QSS_A_00183.