

## АНАЛИЗ ПРОИЗВОДИТЕЛЬНОСТИ БАЗ ДАННЫХ POSTGRESQL/POSTGIS И MONGODB ДЛЯ ГЕОПРОСТРАНСТВЕННЫХ ЗАПРОСОВ

**О. А. Ляшенко, С. Н. Литвинов, Н. А. Солодка**

ГБУЗ «Украинский государственный химико-технологический университет»

просп. Гагарина, 8, г. Днепр 49005, Украина.

E-mail: oksana\_liashenko@udhtu.edu.ua, litvinovwebdev@gmail.com, solodka\_n\_o@ukr.net

С ростом количества интернет-пользователей все больше возникает потребность в веб-сервисах с высокой скоростью отклика при большом количестве одновременных подключений. Данные программные системы классифицируются как Highload-системы. Оптимизация таких систем для ускорения отклика является важным критерием разработки современных веб-ресурсов. В связи с этим многими специалистами разрабатываются подходы, методологии и предлагаются архитектуры для таких систем. В работе рассматриваются информационные системы, оперирующие пространственными данными (они же геоданные), которые относятся к специальной области геоинформационных и развлекательных технологий – веб-картографии. Одной из особенностей таких систем является большое количество операций расчета объектов в геопространстве относительно друг друга, что может сказаться на скорости отклика системы. Следовательно, выбор правильного решения для проектирования и разработки подобного рода задач является актуальным. Одним из возможных решений является выбор такой системы управления базами данных для хранения пространственных данных, которая позволит не только хранить, редактировать и удалять данные, но и производить обработку геоданных. Предложено рассмотреть реляционную модель данных на примере PostgreSQL/PostGIS и нереляционную – на примере MongoDB, которые распространяются по лицензии BSD, имеют возможность хранить и выполнять операции на геоданных. Данная работа была выполнена на синтаксических наборах данных для двух типов базовых операций в геоинформационном анализе: пересечения линий в пространстве и вхождения точек в плоскость. В результате исследований было выявлено, что MongoDB по мере увеличения данных все так же надежно и максимально быстро справляется с поставленными задачами в отличии от PostgreSQL (с плагином PostGIS), где с увеличением данных значительно увеличивалось время обработки операций, а также вероятность неуспешного завершения программы. Данные исследования можно применить при проектировании архитектуры Highload-сервиса, предоставляющего операции обработки геоданных. И тем самым выбрать для сервиса правильный слой данных, который будет справляться с поставленными задачами сервиса.

**Ключевые слова:** Highload, веб-ресурс, веб-картография, база данных, пространственные данные, геоинформационная система, геопространственный запрос, PostgreSQL, PostGIS, MongoDB, сферическая геометрия.

## АНАЛІЗ ПРОДУКТИВНОСТІ БАЗ ДАНИХ POSTGRESQL/POSTGIS І MONGODB ДЛЯ ГЕОПРОСТОРОВИХ ЗАПИТІВ

**О. А. Ляшенко, С. М. Литвинов, Н. О. Солодка**

ДВНЗ «Український державний хіміко-технологічний університет»

просп. Гагаріна, 8, м. Дніпро 49005, Україна.

E-mail: oksana\_liashenko@udhtu.edu.ua, litvinovwebdev@gmail.com, solodka\_n\_o@ukr.net

Із зростанням кількості інтернет-користувачів все більше виникає потреба в веб-сервісах з високою швидкістю відгуку при великій кількості одночасних підключень. Дані програмні системи класифікуються як Highload-системи. Оптимізація таких систем для прискорення відгуку є важливим критерієм розробки сучасних веб-ресурсів. У зв'язку з цим фахівцями розробляються підходи, методології та пропонуються архітектури для таких систем. В роботі розглядаються інформаційні системи, які оперують просторовими даними (вони ж геодані), які відносяться до спеціальної області геоінформаційних та розважальних технологій – веб-картографії. Однією з особливостей таких систем є велика кількість операцій розрахунку об'єктів в геопросторі відносно один одного, що може позначитися на швидкості відгуку системи. Отже, вибір правильного рішення для проектування і розробки подібного роду завдань є актуальним. Одним з можливих рішень є вибір такої системи управління базами даних для зберігання просторових даних, яка дозволяє не тільки зберігати, редагувати і видаляти дані, але і проводити обробку геоданих. Запропоновано розглянути реляційну модель даних на прикладі PostgreSQL / PostGIS і нереляційну – на прикладі MongoDB, які поширюються за ліцензією BSD, мають можливість зберігати і виконувати операції на геоданих. Дана робота була виконана на синтаксичних наборах даних для двох типів базових операцій в геоінформаційному аналізі: перетинання ліній в просторі і входження точок в площину. В результаті досліджень було виявлено, що MongoDB в міру збільшення даних продовжує так само надійно і максимально швидко справлятися з поставленими завданнями на відміну від PostgreSQL (з плагіном PostGIS), де зі збільшенням даних значно збільшувалася час обробки операцій, а також ймовірність невдалого завершення програми. Дані дослідження можна застосувати при проектуванні архітектури Highload-сервісу, що надає операції обробки геоданих. І за допомогою цього вибрати для сервісу правильний шар даних, який буде справлятися з поставленими завданнями сервісу.

**Ключові слова:** Highload, веб-ресурс, веб-картографія, база даних, просторові дані, геоінформаційна система, геопросторові запити, PostgreSQL, PostGIS, MongoDB, сферична геометрія.

**АКТУАЛЬНОСТЬ РАБОТЫ.** С каждым годом количество пользователей интернет и подключенных к нему устройств растет, а вместе с ними и интерес к веб-системам, которые способны обслуживать большое количество подключений и запросов к ним. В сфере разработки программного обеспечения данные программные системы классифицируются как Highload-системы и имеют собственные практики, применяемые при разработке высоконагруженных и устойчивых систем, способных обслуживать множество пользователей в единицу времени [1]. Как правило, архитектура систем такого рода состоит из множества подсистем и решений, таких как: база данных (БД), сервера кэширования, сервера обработки пользовательских данных, сервера маршрутизации и т.д. [2].

Highload-системы классифицируются по решению двух типов задач на:

- высоконагруженные данными системы: в реальном времени могут хранить и обрабатывать такие объемы данных, которые превышают возможности одного компьютера;
- высоконагруженные вычислениями системы: объем и сложность вычислений превышает возможности одного компьютера [2].

Разработка программных систем с высокой нагрузкой на сегодняшний день достаточно актуальная тема. Количество высоконагруженных Интернет-сервисов постоянно растет – интернет-СМИ, новостные или развлекательные порталы, интернет-магазины, сайты-сервисы и т.д. При этом зачастую, в зависимости от вида информации, которую необходимо предоставить пользователю, возникает необходимость использования различных видов отображения пространственных данных [3, 4]. Хранение таких данных в базе данных является разумным решением, т.к. с базой данных можно эффективно предоставлять доступ к большим объемам информации. Кроме того, благодаря использованию базы данных также возможно организовать работу с данными одновременно для множества пользователей.

Одними из ярких примеров пространственных данных являются спутниковые изображения, карты. Информационные системы, оперирующие такими данными относятся к геоинформационным системам (ГИС) [5–8], а соответствующие картографические веб-сервисы, созданные для пользователей Интернета, относятся к специальной области геоинформационных и развлекательных технологий – веб-картографии.

Данного рода приложения активно используются разными государственными учреждениями и промышленными отраслями, такими как сельское [9], лесное хозяйство, управление земельными ресурсами [10], кадастровые системы учета, управление различными видами транспорта (воздушным движением, железнодорожным и автомобильным) [11], мониторинг окружающей среды [12], строительный бизнес, медицина и т.д. Также широкое распространение имеют различные развлекательные ГИС веб-сервисы для обслуживания рядовых пользователей

интернета: системы навигации (2GIS), игровые платформы (Pokemon GO, Minecraft EARTH), службы такси (Uber), службы доставки (Glovo).

Одной из черт таких ГИС систем является большое количество операций расчета объектов в геопространстве относительно друг друга, что может сказаться на скорости отклика системы. Следовательно, выбор правильного решения для проектирования и разработки подобного рода задач является основополагающим.

Зачастую, узким местом таких приложений является база данных. Для хранения пространственных данных необходимо выбрать такую систему управления базами данных (СУБД), которая позволит не только хранить, редактировать и удалять данные, но и производить обработку геоданных [13, 14].

Традиционно для хранения данных и управления географической информацией используют реляционные базы данных. Однако в последнее время в связи с постоянно ускоряющимся ростом объема и сложности данных постепенно набирают популярность решения, основанные на использовании так называемых нереляционных технологий (NoSQL) [15, 16], в частности системы хранения и обработки данных MongoDB [16].

Целью данной работы являются исследования по определению возможностей и ограничений реляционной и нереляционной СУБД при работе с геоинформацией на основе open-source решений общего назначения, PostgreSQL/PostGIS и MongoDB. Также предпринята попытка оценить производительность баз данных, созданных на основе двух указанных подходов в запросах на геопространственных данных, а конкретно на задаче определения пересечения линий и задаче вхождения (локализации) точек в плоскость, которые являются одними из базовых в ГИС-анализе при выполнении основных процедур.

**МАТЕРИАЛ И РЕЗУЛЬТАТЫ ИССЛЕДОВАНИЯ.** При создании геоинформационных систем, одним из важнейших компонентов которых являются базы данных для хранения пространственных данных, необходимо выбрать такую СУБД, которая позволит не только хранить, редактировать и удалять данные, но и производить обработку геоданных.

На текущий момент связка Postgres/PostGIS получила такое широкое распространение, что даже крупные производители проприетарных ГИС включили ее поддержку в свои продукты.

PostgreSQL – это мощная объектно-реляционная система управления базами данных (ORDBMS). Она выпускается под лицензией BSD и, таким образом, является свободным и открытым программным обеспечением. Языком интерфейса базы данных PostgreSQL является стандартный язык интерфейса SQL, который позволяет вставлять данные, обновлять и запрашивать данные, хранящихся в реляционных таблицах.

PostGIS – открытое программное обеспечение, добавляющее дополнительную функциональность к СУБД PostgreSQL, а именно поддержку географических объектов. PostGIS расширяет возможности

PostgreSQL с точки зрения хранения пространственных данных, запросов к ним и управления ими. Основным достоинством PostGIS является возможность использования языка SQL совместно с пространственными операторами и функциями. Кроме простого хранения данных, PostGIS позволяет осуществлять любые виды операций над ними.

Функции PostGIS можно разделить на следующие пять категорий:

- менеджмент: функции управления информацией о пространственных таблицах и PostGIS администрирование;
- преобразование: функции, которые преобразуют геометрии и внешние форматы данных;
- извлечение: функции, восстанавливающие свойства и измерения геометрии;
- сравнение: функции, которые сравнивают две геометрии относительно их пространственного положения;
- поколение: функции, которые генерируют новые геометрии от других пользователей [17].

В качестве СУБД, основанной на нереляционном подходе, рассмотрим MongoDB. MongoDB – СУБД с открытым исходным кодом, не требующая описания схемы таблиц и ориентированная на документы. Она классифицирована как NoSQL, использует JSON-подобные документы для описания схемы БД [16]. Данная СУБД написана на языке C++ и ее основной фокус заключается в высокой производительности и сохранении некоторых дружественных свойств SQL, таких как реляционные отношения, ACID транзакции и т.д.

В настоящее время для хранения пространственных геометрий MongoDB использует объекты GeoJSON. GeoJSON – спецификация с открытым исходным кодом для форматирования фигур в пространстве координат с помощью JSON. Спецификация GeoJSON и его библиотечная поддержка растет в самых популярных языках [2].

Каждый документ GeoJSON (или поддокумент) обычно состоит из двух полей:

1. Тип – отображаемая форма, которая информирует читателя GeoJSON о том, как интерпретировать поле «координаты».
2. Координаты – массив точек, структура которого зависит от поля «Тип».

Каждый из вариантов реализации пространственных баз данных имеет свой подход к работе с геометриями.

В настоящее время Postgres/PostGIS поддерживает геометрические операции с 2D геометрией и сферической геометрией. В то время как в MongoDB поддерживаются те же операции нахождения пересечения линий для кривых и поверхностей, но только в сферической геометрии. Также поддерживается операция сортировки по расстоянию от заданных координат.

Хотя для работы с пространственной геометрией существуют библиотеки с открытым исходным кодом, такие как JTS Topology Suite, GEOS, ArcGIS, MongoDB заточена под задачи сферической геометрии,

так как большинство приложений работают с данными GPS, которые отлично вписываются в сферическую модель.

MongoDB поддерживает операции чтения и записи сферических фигур в GeoJSON. PostGIS поддерживает только операции вхождения точки в плоскость, операцию пересечения линий, которые также поддерживаются MongoDB. Операция определения расстояния между двумя объектами также поддерживается PostGIS.

Для проведения сравнительного анализа двух СУБД – PostgreSQL и MongoDB – набор исходных данных был сгенерирован вручную из данных улиц и домов города Нью-Йорк.

Информация для создания цифровой карты в большинстве случаев собирается и представляется послойно, отражая иерархический процесс ручного картографирования. Большинство ГИС-технологий анализа и визуализации таких слоев базируется на операции определения пересечения линий. Базовой частью алгоритма для этой операции является определение существования точки пересечения двух линейных отрезков, заданных координатами их концов, и вычисление ее координат.

Задача определения пересечения линий работает для любой геометрии в пространстве и показывает, пересекается ли эта геометрия с какой-либо другой геометрией или нет. Эта проблема очень важна, например, в сценариях реального времени, например, когда пользователь хочет знать, на какой стороне дороги он находится.

Поэтому было создано несколько синтетических наборов данных с учетом наихудших сценариев развития событий. Первый набор данных – для операции поиска пересечения линий в пространстве. Он представляет из себя набор плотно пересекающихся улиц жилого района (рис. 1). Каждая прямая пересекает как минимум одну линию в другом слое, при этом некоторые линии пересекают несколько линий слоя. Приведенный выше набор данных эмулирует большинство случаев в режиме реального времени с наихудшими сценариями развития событий.

Еще одной операцией, которая относится к базовым в ГИС-анализе является задача вхождения или локализации точек в плоскости. Как и предыдущая задача, проблема локализации точки также работает для любой геометрии, и показывает, находится ли данная геометрия полностью внутри другой геометрии или нет, какой из полигонов содержит каждую из точек и т.п. Это очень важная и популярная проблема в пространственных базах данных, которая полезна в следующих задачах: генерации карт, моделирования, анализа пространственных данных и т.д. К примеру, такого рода задачей может быть задача проверки, как изменилось количество домов в городе за определенный период времени, проанализировав пространственные данные этого города. Возможно подсчитать количество точек (которые представляют каждый дом в отдельности) в полигоне (городе).



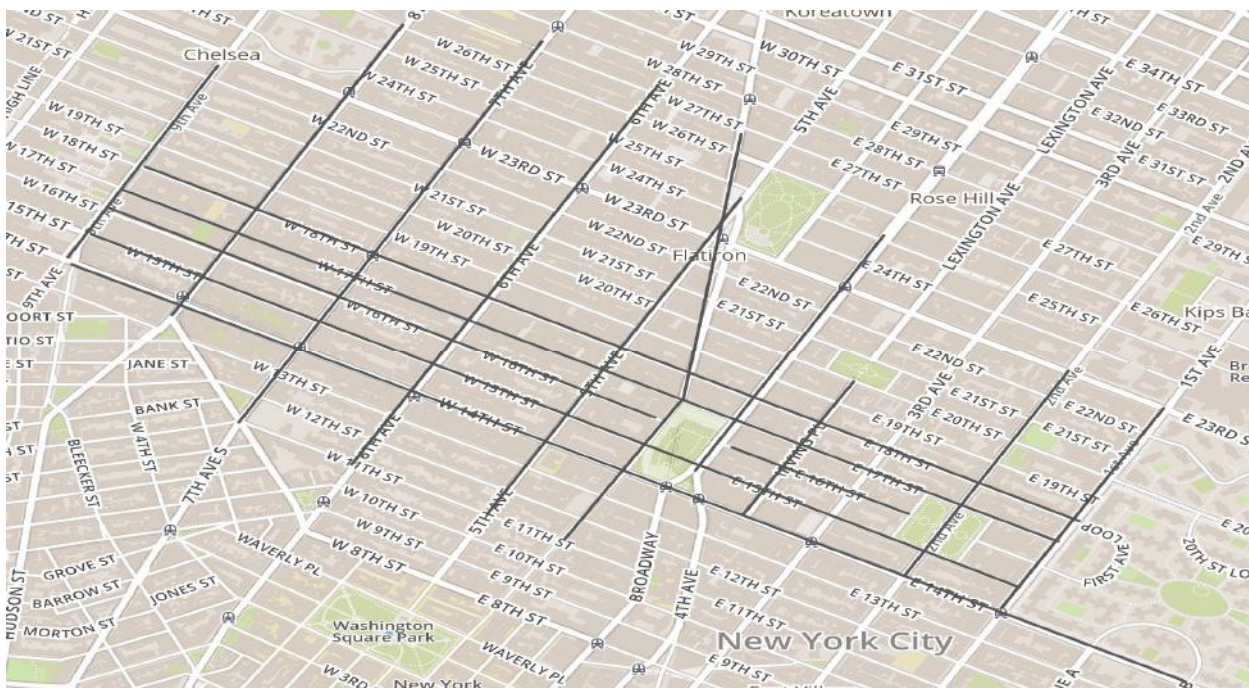


Рисунок 1 – Проблема пересечения линий

Общий подход к решению этого типа задач состоит из двух этапов: провести вертикальную линию от исследуемой точки вверх («до бесконечности»); подсчитать, сколько раз эта линия пересечет границу полигона: если число пересечений нечетное, точка находится внутри полигона; если четное – вне.

Анализ задачи вхождения точек в плоскость проводился на исходном наборе данных, состоящем из двух независимых слоев. Один слой представлял

собой линию инкрементальной длины. Размер слоя варьировался от 10 линий до 10 миллионов линий с точками на них, которые отображали некоторые происшествия на карте. Другой слой состоял из квадратных блоков разного периметра, разбросанных последовательно и случайно, через пробел с несколькими линиями внутри коробки, несколько пересекающихся линий и несколько линий внутри коробки, совершенно случайно (рис. 2).

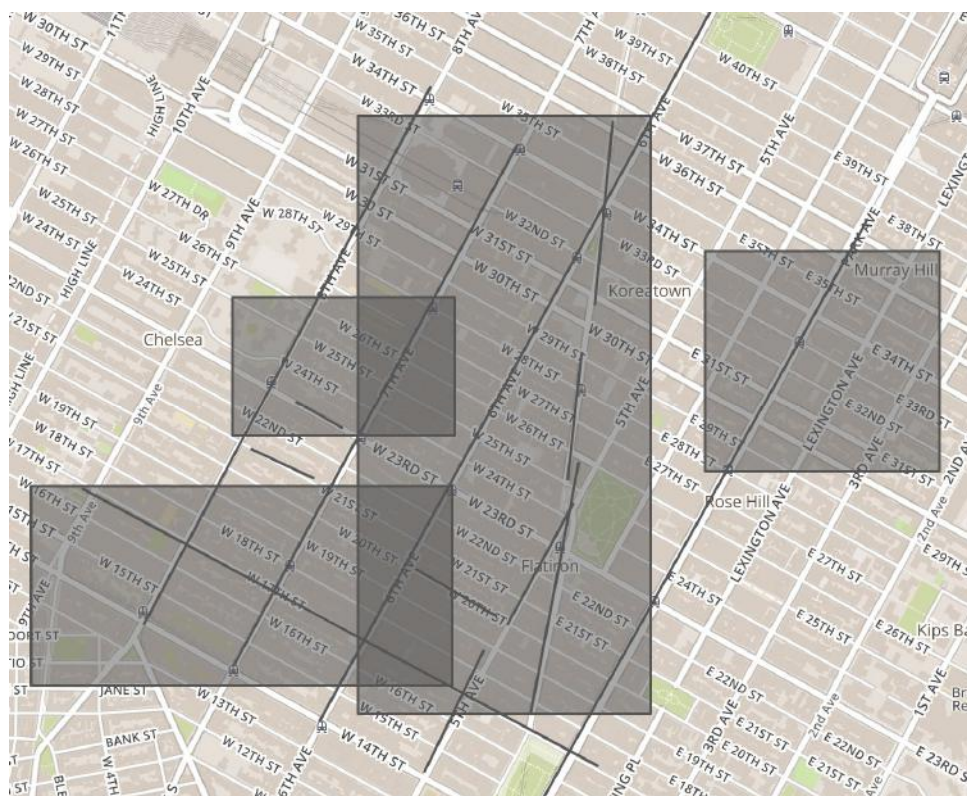


Рисунок 2 – Задача вхождения точки в плоскость

Далее приведен сравнительный анализ производительности СУБД PostgreSQL 9.1/PostGIS 2.0 и MongoDB 3.6.

Компьютер, на котором запускались тесты, оснащен процессором Intel Core i7 (2,2 ГГц) с ОЗУ 16 ГБ и жестким диском объемом 250 ГБ. Операционная система – MacOS High Sierra.

Все данные для анализа обрабатывались с использованием оперативной памяти и не использовалась вторичная память.

Испытания проводились на обеих СУБД с одними и теми же наборами данных, с индексацией по данным в случае с MongoDB использовался индекс 2dsphere и без индексации. В табл. 1 и на рис. 3 приведены результаты тестирования для задачи определения пересечения линий.

В табл. 2 и на рис. 4 приведены результаты тестирования для задачи вхождения точек в плоскость.

Таблица 1 – Проблема пересечения линий

Количество пересечений	PostgreSQL/PostGIS		MongoDB	
	Без индексации, с	С индексацией, с	Без индексации, с	С индексацией, с
21	1,546	9,721	1	1
1875	88	48	18	20
195691	13364	1963	185	190
< (1000 * 1000)	> 21600	> 21600	4093	3140

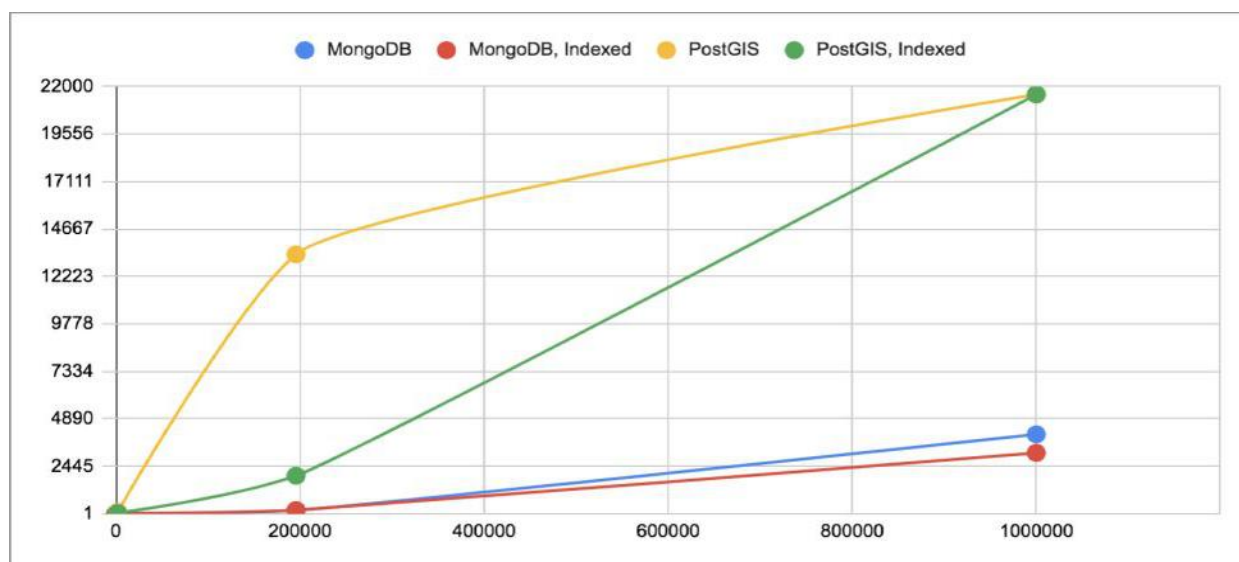


Рисунок 3 – График роста времени при расчете пересечения линий

Таблица 2 – Проблема локализации точек (вхождение точек в плоскость)

Количество точек	PostgreSQL/PostGIS		MongoDB	
	Без индексации, с	С индексацией, с	Без индексации, с	С индексацией, с
5000	53	55	6,75	7,75
5471	85	77	12	12
17004	142	159	21	24
46000	675	631	50	51
122006	1339	1110	51	53
204024	3991	3121	99	106



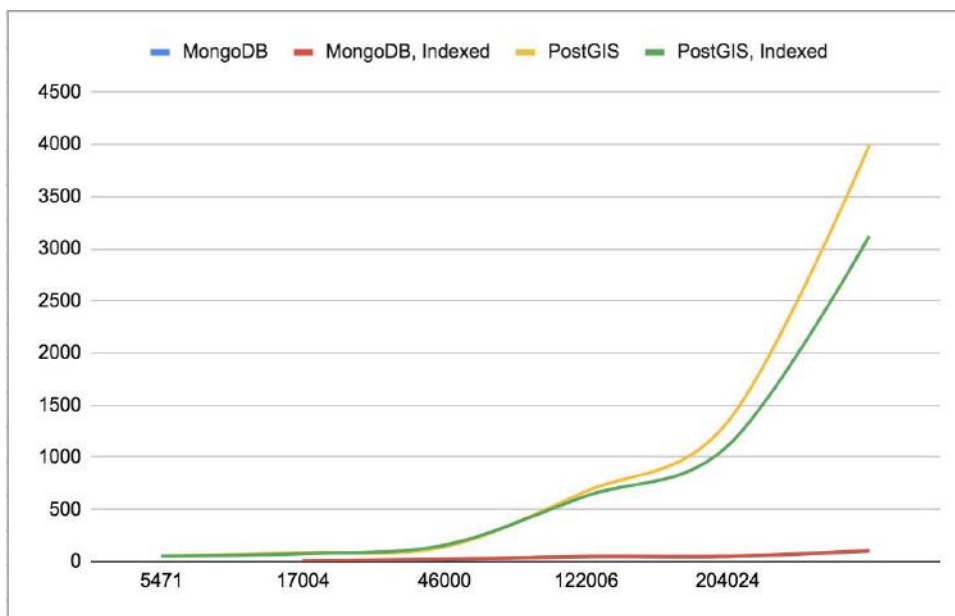


Рисунок 4 – Графік росту часу при входу точок в площину

**ВЫВОДЫ.** Стремительный рост различных картографических веб-сервисов и появление огромного множества пользовательских карт в сети Интернет дали достаточно широкое распространение области веб-картографии. В работе представлено исследование по сравнению возможностей и ограничений двух различных СУБД при работе с геоинформацией. Материалом для исследования являются: реляционная СУБД на примере PostgreSQL 9.1/PostGIS 2.0 и нереляционная – MongoDB 3.6, которые исследовались с целью установления, какая из платформ может быть более подходящей для высоконагруженных геоинформационных приложений. Для освоения этих двух технологий набор исходных данных был сгенерирован вручную из данных улиц и домов города Нью-Йорк.

Была проведена оценка производительности выбранных баз данных, созданных на основе двух указанных подходов в запросах на геопространственных данных, а конкретно на задаче определения пересечения линий и задаче вхождения (локализации) точек в плоскость, которые являются одними из базовых в ГИС-анализе.

Результат показал, что применение NoSQL решения в виде MongoDB, справляется быстро и надежно как на малых, так и на больших наборах данных. Этот вывод может быть основополагающим фактором для использования технологии MongoDB в подобного рода задачах.

#### ЛИТЕРАТУРА

1. Плотников Д. Ю., Малеваный Е. Ф., Аноприенко А. Я. *Інформаційні управляючі системи та комп'ютерний моніторинг*: зб. матеріалів III всеукр. наук.-техн. конф. студентів, аспірантів та молодих вчених (Донецьк, 16–18 квіт. 2012 р.). Донецьк : ДонНТУ, 2012. С. 431–435.
2. Клеппман М. Высоконагруженные прило-

жения. Программирование, масштабирование, поддержка. Питер : 2019. 740 с.

3. John E. Harmon, Steven J. Anderson. The design and implementation of geographic information systems. Hoboken, New Jersey : John Wiley & Sons, Inc., 2003. 272 p.

4. Бурачек В. Г., Железняк О. О., Зацерковний В. І. Геоінформаційний аналіз просторових даних. Ніжин : ТОВ Видавництво «Аспект-Поліграф», 2011. 440 с.

5. Steiniger S., Hunter A. Free and open source GIS software for building a spatial data infrastructure. *Geospatial Free and Open Source Software in the 21st Century*. Lecture Notes in Geoinformation and Cartography. Springer-Verlag Berlin Heidelberg, 2012. P. 247–261. [https://doi.org/10.1007/978-3-642-10595-1\\_15](https://doi.org/10.1007/978-3-642-10595-1_15)

6. Karpik A. P., Musikhin I. A. Research and practical trends in geospatial sciences. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences : XXIII ISPRS Congress* (12–19 July 2016). Prague, Czech Republic, 2016. P. 177–184. <https://doi.org/10.5194/isprsarchives-XLI-B6-177-2016>

7. Шипулін В. Д. Основні принципи геоінформаційних систем. Харків : ХНАМГ, 2012. 312 с.

8. Ткаченко В. П., Овраменко М. І., Губа В. Д., Зелений О. П. Інструментальні засоби створення міських геоінформаційних систем на базі програмних продуктів з відкритим кодом. *Проблеми безперервної географічної освіти і картографії*. Харків, 2013. Вип. 17. С. 60–64.

9. Зацерковний В. І., Кривоберець С. В. Аналіз можливості підвищення ефективності сільськогосподарського виробництва при застосуванні ГІТ у задачах управління. *Вісник Чернігівського державного технологічного університету. Серія «Технічні науки»*. Чернігів, 2013. № 3(67). С. 174–183.2225-7551

10. Dubrovsky A. V., Antipov I. T., Kalenitsky A. I., Guk A. P. Elements of Geoinformation Support of Natural Resource Management System. *International Journal of Advanced Biotechnology and Research*, 2018. Vol. 9, Issue 1. P. 1185–1202.

11. Касім А. М., Касім М. М., Ясенев С. О. Специфіка використання геопросторових даних для задач аеронавігації. *Наукоємні технології*. Київ, 2016. № 1 (29). С. 16–22. <https://doi.org/10.18372/2310-5461.29.10087>

12. Лазоренко-Гевель Н. Ю. Створення геоінформаційних моделей даних моніторингу природних комплексів. *Містобудування та територіальне планування*. Київ, 2014. Вип. 51. С. 275–283.

13. Zeiler Michael Моделирование нашего мира. Пособие ESRI по проектированию баз геоданных. Киев : ECOMM, 2003. 182 с.

14. Сокольников А. М. Сравнительный анализ подходов к разработке архитектуры и систем управ-

ления базами данных для высоконагруженных WEB-сервисов. *Кибернетика и программирование*. 2014. № 4. С. 1–13. <https://doi.org/10.7256/2306-4196.2014.4.12800>

15. Ляшенко О. А., Конашков О. О. Сравнение реляционной и нереляционной СУБД на примере MYSQL и MONGODB. *Комп'ютерне моделювання та оптимізація складних систем: матеріали V Міжнародної наук.-техн. конф. (Дніпро, 5-7 лист. 2019 р.)*. Дніпро : ДВНЗ УДХТУ, 2019. С. 125–126. <https://doi.org/10.32434/CMOCS-2019>

16. Солодка Н. О., Поліщук Є. О., Ляшенко О. А. Використання графової та реляційної моделей даних при розробці експертних систем. *Вісник Херсонського національного технічного університету*. Херсон, 2018. № 4(67). С. 246–251.

17. Regina O. Obe, Leo S. Hsu. PostGIS in Action. USA : Manning Publications, 2011. 520 p.

## PERFORMANCE ANALYSIS FOR GEOSPATIAL QUERIES POSTGRES/POSTGIS DATABASE VS. MONGODB

O. Liashenko, S. Lytvynov, N. Solodka

SHEI Ukrainian State University of Chemical Technology

prosp. Gagarina, 8, Dnipro, 49005, Ukraine.

E-mail: oksana\_liashenko@udhtu.edu.ua, litvinovwebdev@gmail.com, solodka\_n\_o@ukr.net

**Purpose.** Explore two types of basic operations in geo information analysis: the operation of intersecting lines in space and the operation of entering points on a plane using two general-purpose database technologies: relational (PostgreSQL) and non-relational (MongoDB). Besides, these approaches should provide the ability to store and process geo information data for a highly loaded web resource. **Methodology.** Two data sets were created for this study. The first data set is for the operation of searching for the intersection of lines in space. It is a collection of tightly intersecting streets of a residential area. The second data set is for the operation of searching for the occurrence of points in the plane in the form of a dense number of points on the map, which represented some incidents on the map. For each data set, the operation of searching for intersections in an arbitrary radius was used. **Results.** After studying the search operations in the given data and tasks, it was found that as the amount of data grows, MongoDB continues to quickly and without fail cope with the tasks in optimal time. While PostgreSQL/PostGIS crashed periodically with data growth, the speed of operations decreased in proportion to the volume of data. For both solutions and data sets, data indexing was also performed, as a result of which the performance in both solutions was improved with an increase in the amount of data. **Originality.** This study has been conducted due to the lack of information on a comparative analysis of the performance of two solutions based on PostgreSQL/PostGIS and MongoDB. For these systems, it is possible to perform operations of searching for intersections of lines in space and the entry of points in the plane, which relate to operations carried out on geo information data. Testing results are given. The test examples show higher database performance values based on NoSQL solutions compared to relational one. **Practical value.** This study may be useful to developers as well as software architects at the stage of designing the application architecture and choosing the type of data model. In particular, the results of the study can be applied in the design of highly loaded geo information web resources with numerous users.

**Key words:** Highload, web resource, web mapping, database, spatial data, geo information system, geospatial query, PostgreSQL, PostGIS, MongoDB, spherical geometry.

## REFERENCES

1. Plotnikov, D. Ju., Malevanyj, E. F. Anoprienko, A. Ja. (2012), "Razrabotka vysokonagruzhenogo veb-prilozhenija" [Highly loaded web application development], *Informacijni upravljajuchi systemy ta komp'juternyj monitoring: zb. materialiv III vseukr. nauk.-tehn. konf. studentiv, aspirantiv ta molodyh vchenyh* [Information management systems and computer monitoring: Material of the third nation. scien.-techn. conference of students and young researches], Donec'k, DonNTU, April 16–18, 2012, pp. 431–435.

2. Kleppman, M. (2019), *Vysokonagruzhenye prilozhenija Programirovanie, masshtabirovanie, podderzhka*. [Heavily loaded applications. Programming, Scaling, Support], Piter, Russia.

3. Harmon, J. E., Anderson, S. J. (2003), *The design and implementation of geographic information systems*. John Wiley & Sons, Inc. Hoboken, New Jersey.

4. Burachek, V. G., Zheleznyak, O. O., Zacerkovnyj, V. I. (2011), *Geoinformacijnyj analiz prostorovyh danyh* [Geoinformation analysis of spatial data], Nizhyn, Russia.

5. Steiniger, S., Hunter, A. (2012), "Free and open

source GIS software for building a spatial data infrastructure”, *Geospatial Free and Open Source Software in the 21st Century*. Lecture Notes in Geoinformation and Cartography. Springer-Verlag Berlin Heidelberg, pp. 247–261. [https://doi.org/10.1007/978-3-642-10595-1\\_15](https://doi.org/10.1007/978-3-642-10595-1_15)

6. Karpik, A. P., Musikhin, I. A. (2016), “Research and practical trends in geospatial sciences”, *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* : XXIII ISPRS Congress, Prague, Czech Republic, July 12–19 2016. pp. 177–184. <https://doi.org/10.5194/isprsarchives-XLI-B6-177-2016>

7. Shypulin, V. A. (2012), *Osnovni pryntsyipy heoinformatsijnykh system* [Basic principles of geographic information systems], BNUUEK, Kharkiv.

8. Tkachenko, V. P., Ovramenko, M. I., Guba, V. D., Zelenyj, O. P. (2013), “Instrumental'ni zasoby stvorennja mis'kyh geoinformacijnyh system na bazi programnyh produktiv z vidkrytym kodom” [Tools for creating urban geo-information systems based on open source software *Problemy bezperervnoi' geografichnoi' osvity i kartografii'* [Problems of continuous geographical education and cartography], vol. 17. pp. 60–64.

9. Zacerkovnyj, V. I., Kryvoberec', S. V. (2013), “Analiz mozhylyvosti pidvyshhennja efektyvnosti sil'skogospodars'kogo vyrobnytva pry zastosuvanni GIT u zadachah upravlinnja” [Opportunity analysis of increasing the efficiency of agricultural production in the application of GIT in management tasks], *Visnyk Chernigivs'kogo derzhavnogo tehnologichnogo universytetu. Serija «Tehnichni nauky»* [Bulletin of Chernihiv State Technological University. Technical Sciences Series], no. 3(67). pp. 174–183.

10. Dubrovsky, A. V., Antipov, I. T., Kalenitsky, A. I., Guk, A. P. (2018), “Elements of Geoinformation Support of Natural Resource Management System”. *International Journal of Advanced Biotechnology and Research*, vol. 9, Issue 1, pp. 1185–1202.

11. Kasim, A.M., Kasim, M.M., Jasenev, S.O. (2016), Specyfika vykorystannja geoprostorovyh danyh dlja zadach aeronavigacii' [Specificity of using geospatial data for aeronautical tasks], *Naukojemni tehnologii'* [Technology-intensive], no. 1 (29). pp. 16–22. <https://doi.org/10.18372/2310-5461.29.10087>

12. Lazorenko-Gevel', N. Ju. (2014), “Stvorennja geoinformacijnyh modelej danyh monitoryngu pryrodnyh kompleksiv” [Creation of geoinformation models for monitoring of natural complexes], *Mistobuduvannja ta terytorial'ne planuvannja* [Urban planning and territorial planning], vol. 51, pp. 275–283.

13. Zeiler, M. (2003), *Modelirovanie nashego mira. Posobie ESRI po proektirovaniju baz geodannyh* [Modeling of our world. ESRI Geodatabase Design Toolkit], Kiev, Ukrain.

14. Sokol'nikov, A. M. (2014), “Sravnitel'nyj analiz podhodov k razrabotke arhitektury i sistem upravlenija bazami danyh dlja vysokonagruzhenykh WEB-servisov” [Comparative analysis of approaches to the development of architecture and database management systems for highly loaded WEB services], *Kibernetika i programmirovannje* [Cybernetics and programming], no. 4. pp. 1–13. <https://doi.org/10.7256/2306-4196.2014.4.12800>

15. Liashenko, O. A., Konashkov, O. O. (2019), “Sravnenie reljacionnoj i nereljacionnoj SUBD na primere MYSQL i MONGODB [Comparison of relational and non-relational database on the example of MYSQL and MONGODB], *Materiali V Mizhnarodnoi naukovo-tehnichnoi konferencii “Komp'juterne modeljuvannja ta optimizacija skladnih sistem”* [5th Int. Sci.-Techn. Conf. proceedings “Computer modeling and optimization of complex systems”], Dnipro, November 5–7, 2019, pp. 125–126. <https://doi.org/10.32434/CMOCS-2019>

16. Solodka, N. O., Polishhuk, Eu. O., Liashenko, O. A. (2018), “Vikorisannja grafovoi ta reljacionnoi modelej danih pri rozrobci ekspertnih sistem” [Using the graph database model and the relation model while developing expert systems], *VISNIK HNT [VISNYK of KNTU]*, no. 4(67), pp. 246–251.

17. Obe, R., Hsu, L. (2011), *PostGIS in Action*. Manning Publications, USA.

Стаття надійшла 09.12.2019.