

1. Поділ даних між базами даних

Обрана мною модель даних охоплює широку область геологічних досліджень, включаючи:

- Дослідницькі проекти
- Дослідників
- Лабораторії
- Гірські породи та їх зразки
- Хімічний склад зразків
- Вимірювання властивостей зразків
- Зображення зразків
- Аналізи зразків
- Гірничі майданчики та їх місцезнаходження

Така комплексна модель даних добре підходить для застосування в одній реляційній базі даних PostgreSQL. Розподіл даних між декількома базами даних може ускладнити управління, підтримку цілісності даних та виконання складних запитів, що зв'язують дані з різних частин предметної області.

2. Продуктивність

Результати тестування продуктивності, наведені в файлі **test.py**, показують, що для досить великої кількості записів MongoDB показує себе набагато краще за PostgreSQL. Різниця в часі читання ж не є такою критичною.

```
Порівняння продуктивності баз даних
=====

Тестування MongoDB (кількість зразків: 100000)...
Час запису MongoDB: 64.65 секунд
Час читання MongoDB: 2.76 секунд
Кількість знайдених записів: 100000

Тестування PostgreSQL (кількість зразків: 100000)...
Час запису PostgreSQL: 109.52 секунд
Час читання PostgreSQL: 3.69 секунд
Кількість знайдених записів: 300000

Результати порівняння:
=====
Кількість зразків: 100000

Час запису:
MongoDB: 64.65 секунд
PostgreSQL: 109.52 секунд
Різниця: 44.87 секунд

Час читання:
MongoDB: 2.76 секунд
PostgreSQL: 3.69 секунд
Різниця: 0.93 секунд
```

Слід зазначити, що MongoDB використовує індексацію для оптимізації просторових запитів до локацій зразків, що може бути перевагою при

виконанні геопросторових аналізів. Проте, PostgreSQL також має можливості просторової індексації та обробки, які можуть забезпечити аналогічну продуктивність.

3. Складність архітектури

Використання двох баз даних (реляційної та NoSQL) додатково ускладнює архітектуру системи, вимагає додаткових механізмів синхронізації та узгодження даних між ними. Це може призвести до збільшення складності розробки, технічного обслуговування та управління системою в цілому.

Враховуючи всі ці фактори, можна зробити висновок, що в даному випадку використання однієї реляційної бази даних PostgreSQL буде більш доцільним та обґрунтованим. Така архітектура забезпечить цілісність даних, простоту управління та ефективну продуктивність для більшості сценаріїв використання.