

Оптимізація баз даних на прикладі PostgreSQL 9

Дипломна робота
Студента групи ДА-72
Грицука Артема Сергійовича

Завдання

- Дослідження областей оптимізації баз даних
- Огляд існуючих інструментів
- Реалізація практичних заходів з оптимізації БД

Актуальність

Правильно оптимізована база даних дає наступні переваги:

- економія на апаратному забезпеченні
- можливість обслуговувати більшу кількість клієнтів
- прискорення роботи з базою даних
- краща відмовостійкість

Інструменти для оптимізації баз даних

PostgreSQL - популярна СУБД, що базується на відкритому коді та має активну спільноту розробників. Завдяки цьому, її можливості не обмежуються лише стандартом SQL. В кожній версії СУБД додається значна кількість інструментів, що покращують її функціональність та додають нові можливості.

Для проектів, що не увійшли до складу СУБД, створено спеціальний Інтернет-ресурс.

Області оптимізації баз даних

База даних під керівництвом СУБД є складним програмним комплексом, роботу якого можна покращити. Незалежно від СУБД виділяють наступні області, в яких можлива оптимізація:

- Оптимізація налаштувань серверу (взаємодія СУБД з апаратним забезпеченням платформи)
- Оптимізація структури бази даних
- Оптимізація запитів
- Оптимізація прикладного ПЗ, що взаємодіє з базою даних

Оптимізація налаштувань серверу

Ефективність роботи СУБД залежить від параметрів, що визначають кількість виділених для неї системних ресурсів та специфіку їх використання. Ці параметри визначаються у файлі конфігурації та за замовчанням не відповідають можливостям сучасних серверів. Згідно теми, в даній роботі розглянуто налаштування PostgreSQL, проте для інших СУБД існують аналогічні механізми.

Основні параметри, що впливають на швидкодію СУБД

Назва	Опис	Значення за замовчанням
shared_buffers	Пам'ять, що виділяється СУБД при її запуску	32 MB
wal_buffers	Визначає кількість пам'яті, що виділяється для збереження транзакції	64kB
checkpoint_segments	Визначає частоту записів до log-файлу	3
checkpoint_completion_target	Визначає час, необхідний на запис одного сегменту	0.5
work_mem	Пам'ять, що виділяється для операцій сортування підчас виконання запиту (якщо сортування вимагає більше пам'яті, дані записуються на диск)	1MB
maintenance_work_mem	Пам'ять, що виділяється на складніші операції, ніж сортування (чистка, створення індексів або ключів)	16MB

Результати оптимізації налаштувань серверу

Назва	Опис	Значення за замовчанням
shared_buffers	Пам'ять, що виділяється СУБД при її запуску	1GB
wal_buffers	Визначає кількість пам'яті, що виділяється для збереження транзакції	16MB
checkpoint_segments	Визначає частоту записів до log-файлу	32
checkpoint_completion_target	Визначає час, необхідний на запис одного сегменту	0.9
work_mem	Пам'ять, що виділяється для операцій сортування підчас виконання запиту (якщо сортування вимагає більше пам'яті, дані записуються на диск)	4MB
maintenance_work_mem	Пам'ять, що виділяється на складніші операції, ніж сортування (чистка, створення індексів або ключів)	32MB

Індексування

Індексування дозволяє підвищити ефективність вибору даних з таблиці шляхом створення сортованого представлення. Оскільки вибір даних з таблиці лежить в основі більшості запитів, індекси є одним з найефективніших засобів підвищення продуктивності роботи з БД. Ефективність індексів зростає із зростанням розміру таблиці

Розділення даних

Розділення даних - механізм заснований на наслідуванні. В результаті застосування розділення, на основі однієї з таблиць БД створюється група таблиць, що мають таку саму структуру та вміщують деяку частину даних початкової таблиці. Ефективно для великих таблиць, в яких є поле критерію для розподілу.

Оптимізація запитів

Оптимізація запитів - велика область оптимізації БД, яка базується на відомостях про організацію планування виконання запиту. При проведенні заходів з оптимізації запитів слід керуватись доступною статистикою щодо використання бази даних та відомостями про час виконання запитів до неї

Висновки

Оптимізація БД - важлива частина роботи з її адміністрування. Існує декілька областей, в яких можливе покращення.

Отримані в даній роботі практичні результати підтверджують ефективність та необхідність заходів оптимізації.

Невміле застосування заходів оптимізації може призвести до погіршення роботи бази даних.

Значення та можливості оптимізації збільшуються із зростанням розміру БД та потужності апаратного забезпечення системи.