

операцій можна працювати з SQL-даними. Крім цього, XML DB дозволяє розкласти XML документи по ієрархічним папкам, встановити для них додаткові засоби контролю доступу (ACL), здійснити пошук потрібних XML документів з контексту і т.д.

Постійно удосконалюються і спрощуються засоби управління СУБД. Багато операцій з адміністрування БД раніше вимагали участі адміністратора БД, тепер виконуються автоматично. А графічний інструмент адміністратора БД – Oracle Enterprise Manager дозволяє, кинувши погляд на всю прикладну систему в цілому, побачити "вузькі місця" і далі допомагає деталізувати проблеми і підказує методи їх усунення. Інтелектуальна експертна система допоможе налаштувати БД.

IV. Аналіз перспективного напрямку розвитку СУБД

Найбільш цікавим напрямом розвитку СУБД є їх розвиток на рівні глобальної мережі, тобто створення СУБД, яка б підтримувала XML-документи, оскільки саме таких документів у мережі стає все більше і більше. Компанія Oracle швидко просунулась у цьому напрямку і створила XML DB. Тепер поряд із звичайними реляційними даними ми можемо зберігати в тій ж БД XML-документи і швидко працювати з ними. При цьому документи зберігаються в СУБД і використовують всі переваги такого зберігання. Спеціальні механізми зберігання, індексування, побудови XML View і т.і. дозволяють не тільки ефективно зберігати, але й запитувати і змінювати ці дані та їх частини. Причому традиційні SQL-операції можуть працювати як з реляційними даними, так і з XML файлами, і, навпаки, за допомогою XML-операцій можна працювати з SQL-даними. До того ж XML DB дозволяє розкласти XML документи по ієрархічним папкам, встановити для них додаткові засоби контролю доступу (ACL), здійснити пошук потрібних XML-документів з контексту.

Список використаної джерел

1. Наместников А.М. Построение баз данных в среде Oracle. Практический курс - УЛГТУ: 2008.
2. Managing Text with Oracle8 ConText Cartridge. Technical White Paper -- Oracle Corporation, 1997.
3. Ю Пуха -- Объектные технологии построения распределенных информационных систем -- Jet Info, 2007, N 16.
4. Oracle Application Server 4.0 White Paper: Product Overview -- Oracle Corporation, 1998.

УДК 681.3.07

ПРОБЛЕМИ РОЗПОДІЛЕНИХ СИСТЕМ УПРАВЛІННЯ БАЗАМИ ДАНИХ

Сивак М.М., Скрипник А.В.

Тернопільський національний економічний університет

I. Постановка проблеми

Основна проблема інформаційних мереж, в крайньому глобальних, заключається у тому, що вони достатньо повільні. В типовій глобальній мережі, інтенсивність обміну даними – біля 5-10 млн. байт в секунду. Внаслідок цього основною вимогою до розподілених систем взагалі, і до розподілених СУБД зокрема, є мінімізація, тобто зменшення до мінімуму об'єму повідомлень в мережі.

II. Основна частина

Прагнення до досягнення поставленої мети – підвищення швидкодії роботи з розподіленою СУБД - приводить, в свою чергу, до необхідності вирішення перерахованих нижче проблем.

- обробка запитів;
- управління каталогом;
- поширення оновлення;
- керування відновленням;
- керування паралельністю.

Розглянемо питання обробки запитів. При мінімізації використання мережі припускається, що сама по собі оптимізація запиту, як і його виконання, повинна бути розподіленою. Іншими словами, загальний процес оптимізації як правило складається з етапу глобальної оптимізації, який супроводжується кількома етапами локальної оптимізації.

Проблема управління каталогом полягає у наступному. Каталог розподіленої системи складається не тільки із звичайних даних, що стосуються базових відношень, представлень, індексів, користувачів та інше, але також і з всієї інформації, необхідної для забезпечення незалежності розміщення, фрагментації і реплікації.

При розгляді питання поширення оновлень основною проблемою реплікації даних є те, що оновлення будь-якого логічного об'єкту повинно розповсюджуватись на всі збережені копії об'єкту. Труднощі виникають через те, що деякий вузол, який містить даний об'єкт, може бути недоступним саме в момент оновлення. В такому випадку очевидна стратегія негайного розповсюдження оновлень на всі копії може виявитися неприйнятною, оскільки передбачається, що оновлення буде провалено, якщо одна з цих копій буде недоступна в даний момент часу. Існують випадки, коли при використанні такої стратегії дані дійсно будуть менш доступними порівняно з їх використанням в нереплікаційному вигляді.

Ще одна проблема – це проблема керування відновленням. Керування відновленням в розподілених системах звичайно побудоване на протоколі двохфазної фіксації. Підтримка двохфазної фіксації необхідна в будь-якому середовищі, в якому одна транзакція може взаємодіяти з декількома автономними адміністраторами ресурсів. Однак вона особливо важлива в розподіленій системі, оскільки адміністратори ресурсів, тобто локальні СУБД, діють на різних вузлах і послідовно, і автономно.

Остання розглянута нами проблема – це керування паралельністю. Управління паралелізмом в більшості розподілених систем, як і в багатьох нерозподілених системах, засновано на блокуванні. Однак в розподіленій системі запити на перевірку, установку і зняття блокувань, є повідомленням, що несе за собою додаткові накладні видатки. Розглянемо, наприклад, транзакцію T , котра потребує оновлення об'єкту, який має реплікації на n віддалених вузлах. Якщо кожен вузол керує блокуваннями для об'єктів, збережених на цьому вузлі, то для простішого способу управління паралелізмом потрібно було б принаймі $5n$ повідомлень.

Отже, з вище наведеного можна зробити висновок, що основною проблемою інформаційних мереж є їх повільність, проте ефективність роботи розподіленої СУБД можна покращити завдяки вирішенню вище перерахованих проблем.

Список використаних джерел

1. Карпова Т. Базы данных. Модели, разработка, реализация. – С-Пб.: Питер, 2004.
2. Дейт К.Дж. Введение в системы баз данных. – К.: Диалектика, 2005.

УДК 004.75

ОРГАНІЗАЦІЯ WEB-ДОСТУПУ ДО РОЗПОДІЛЕНОЇ БАЗИ ДАНИХ ДЛЯ МОДЕЛЮВАННЯ ДИСКРЕТНИХ ДИНАМІЧНИХ СИСТЕМ

Співак І.Я., Брославський В.П.

Тернопільський національний економічний університет

Архітектура "клієнт-сервер" є фактичним стандартом для багатокористувацьких СУБД, прикладні програми, реалізовані в її рамках, мають розподілений характер. Тобто, частина функцій прикладної програми реалізована в програмі-клієнті, інша - в програмі-сервері, причому для їх взаємодії буде визначений деякий протокол.

Основний принцип технології "клієнт-сервер" полягає в розподілі функцій стандартного інтерактивного застосування на чотири групи, що мають різну природу. Перша група - це функції введення і відображення даних. Друга група об'єднує суто прикладні функції, що характерні для цієї предметної області. До третьої групи відносяться фундаментальні функції зберігання і управління інформаційними ресурсами (базами даних, файловими системами і так далі). А функції четвертої групи - службові, що відіграють роль зв'язку між функціями перших трьох груп.

Виділяються чотири підходи, реалізовані в наступних моделях [1]:

- модель файлового сервера (File Server - FS);
- модель доступу до віддалених даних (Remote Data Access - RDA);
- модель сервера бази даних (DataBase Server - DBS);
- модель сервера додатків (Application Server - AS).

FS-модель є базовою для локальних мереж. Відповідно до цієї моделі один з комп'ютерів в мережі вважається файловим сервером і надає послуги з обробки файлів іншим комп'ютерам. На інших комп'ютерах в мережі функціонує додаток, в кодах якого поєднані компонент представлення і