

Сучасні клієнт-серверні технології та їх застосування при вивченні систем управління базами даних

Із розвитком інформаційно-комунікаційних технологій зростає складність інформаційних систем і обсяги даних у них. Кожна прикладна програма є відображенням якоїсь частини реального світу і містить його формалізований опис у вигляді даних. Великі масиви даних розміщують окремо від коду програми, що виконується, і організовують у вигляді баз даних. Для роботи з даними використовують особливі програмні комплекси, системи управління базами даних (СУБД).

На сучасному етапі розвитку засобів опрацювання даних домінує клієнт-серверна архітектура, у рамках якої деяка прикладна програма може взаємодіяти з іншими такими програмами у рамках мережі шляхом обміну даними через сервер баз даних.

Технологія клієнт-сервер означає такий спосіб взаємодії програмних компонентів, при якому вони утворюють єдину систему. Як видно із назви, існує клієнтський процес, що вимагає певних ресурсів, а також серверний процес, що ці ресурси надає. Зовсім необов'язково, щоб вони перебували на одному комп'ютері. Звичайно прийнято розміщувати сервер на одному вузлі локальної мережі, а клієнтів – на інших вузлах.

У контексті бази даних клієнт управляє користувацьким інтерфейсом і логікою роботи, діючи як робоча станція. Клієнт приймає від користувача запит, перевіряє синтаксис і генерує запит до бази даних мовою SQL або іншою мовою бази даних, відповідно до логіки роботи програми-клієнта. Потім передає повідомлення серверу, очікує надходження відповіді й форматує отримані дані для подання їх користувачеві. Сервер приймає й опрацьовує запити до бази даних, після чого відправляє отримані результати назад клієнтові. Таке опрацювання включає перевірку повноважень клієнта, забезпечення вимог цілісності, а також виконання запиту й оновлення даних. Крім цього підтримується управління паралельністю й відновленням.

Архітектура клієнт-сервер має ряд переваг:

- забезпечується ширший доступ до існуючих баз даних;
- підвищується загальна продуктивність системи: оскільки клієнти та сервер перебувають на різних комп'ютерах, їхні процесори можуть виконувати різні завдання паралельно. Налаштування продуктивності комп'ютера із сервером спрощуються, якщо на ньому виконується тільки робота з базою даних;
- знижується вартість апаратного забезпечення; досить потужний комп'ютер з більшим пристроєм зберігання потрібний тільки серверу – для зберігання й управління базою даних;
- скорочуються комунікаційні витрати. Частина операцій виконують клієнтські комп'ютери і посилають через мережу тільки запити до баз даних, що дозволяє значно скоротити обсяг даних, що пересилають мережею;
- підвищується рівень несуперечливості даних, оскільки кожній клієнтській програмі не доведеться виконувати власну перевірку їх цілісності;
- архітектура клієнт-сервер природно відображається на архітектуру відкритих систем.

Подальше розширення дворівневої архітектури клієнт-сервер припускає поділ функціональної частини колишнього, «товстого» (інтелектуального) клієнта на дві частини. У трирівневій архітектурі клієнт-сервер «тонкий» (не інтелектуальний) клієнт на робочій станції управляє тільки користувацьким інтерфейсом, тоді як середній рівень опрацювання даних управляє всією іншою логікою роботи клієнтської програми. Третій рівень – сервер бази даних. Ця трирівнева архітектура виявилася більш придатною для деяких середовищ – наприклад, для мереж Internet та Intranet, де як клієнт може виступати звичайний Web-браузер.

У процесі розвитку технології клієнт-сервер змінювалися і способи її реалізації. Розглянемо основні способи організації доступу до даних за технологією клієнт-сервер та відмітимо деякі особливості кожного варіанту.

Дворівнева архітектура клієнт-сервер. У цій архітектурі на виділеному сервері, що працює, як правило, під управлінням серверної операційної системи, встановлюють спеціальне програмне забезпечення (ПЗ) – сервер БД, наприклад, Microsoft SQL Server або Oracle. СУБД поділяється на дві частини: клієнтську й серверну. Основа роботи сервера БД – використання мови запитів (SQL). Запит мовою SQL, переданий клієнтом (робочою станцією) серверу БД, породжує пошук і вибір даних на сервері. Вибрані дані транспортуються мережею від сервера до клієнта (див. Рис. 1.1). Таким чином, кількість переданих мережею даних зменшується в багато разів.

Такий принцип широко використовується не тільки в базах даних, а й в інших галузях застосування комп'ютерних технологій. Розглянемо деякі з найбільш відомих.

Електронна пошта. Тут у ролі клієнта виступає поштова програма (The Bat, Outlook, тощо), яка з'єднується з поштовим сервером, на якому міститься електронна поштова скринька. Поштова програма «спілкується» з сервером (відправляє запити на отримання окремого листа, запит на вилучення листа тощо) згідно з стандартним протоколом (найпоширеніший POP3 та SMTP).

WWW – найбільш відомий сервіс мережі Internet, завдяки якому вона й набула величезної популярності серед користувачів. У даному випадку клієнт-серверна технологія застосовується при перегляді Web-сторінок. У ролі клієнтської програми виступає Web-браузер, який з допомогою протоколу HTTP відправляє запити серверу, на якому розміщені Web-сторінки. HTTP-протокол також широко використовують у «тонких» клієнтах, які будуть розглянуті нижче.

Практично будь-який сервіс, що надає Internet, базується на даній технології, адже при отриманні довільних даних з мережі користувацька програма відправляє запит віддаленому серверу, в якому містяться певні інструкції. Наприклад, вилучити лист – при управлінні поштовою скринькою, відправити до клієнта вміст певного файлу – при роботі з ftp, запит відомостей про користувача – при

спілкуванні за допомогою ICQ. Подібні приклади можна навести практично для всіх сервісів глобальної мережі.

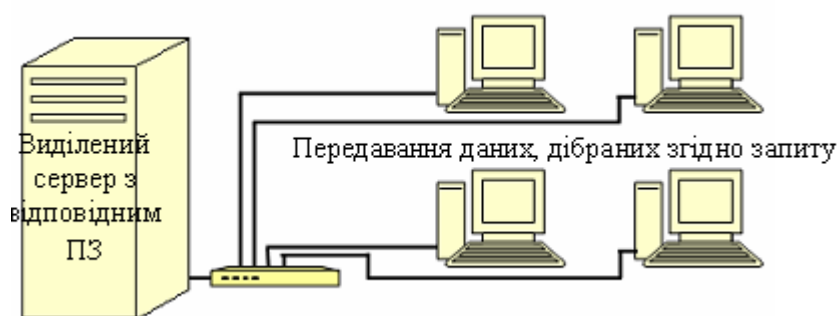


Рис 1.1 Дворівнева архітектура «клієнт-сервер»

Виділяють такі особливості використання дворівневої архітектури:

- можливість використання різноманітних засобів, що їх надає СУБД, завдяки безпосередньому з'єднанню з сервером БД;
- значно знижується навантаження на мережу. Завдяки відправленню запитів і опрацюванню їх на сервері клієнт обмінюється із сервером тільки необхідними даними;
- складніше піддається масштабуванню, ніж трирівнева архітектура, тому що доводиться використовувати різні клієнти для різних операційних систем;
- потребує додаткових затрат для облаштування (встановлення) клієнтської частини;
- при модифікаціях серверної частини (наприклад додаванні нових функцій, зміні базової СУБД тощо) як правило потребує заміни (поновлення) клієнтської частини;

Трирівнева архітектура клієнт-сервер функціонує в Intranet та Internet мережах. Клієнтська частина («тонкий клієнт»), з яким працює користувач, є Web-браузером або прикладною клієнтською програмою, що взаємодіє із Web-сервісами. Уся програмна логіка винесена на сервер застосувань, що забезпечує формування запитів до БД, які передаються серверу баз даних для їх виконання. Сервер застосувань може бути Web-сервером або спеціалізованою серверною програмою (наприклад, Oracle Forms Server). Схема такої структури подана на Рис 1.2

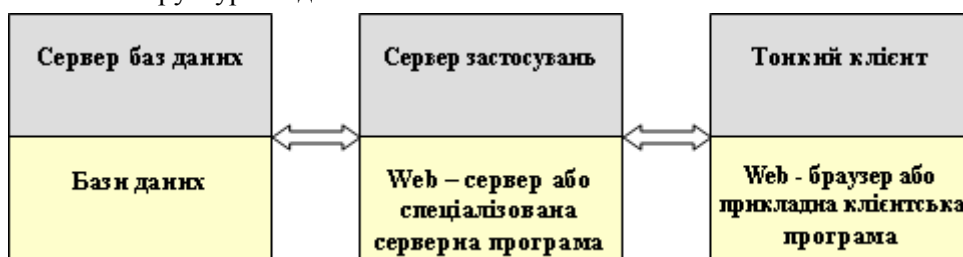


Рис 1.2 Схема трирівневої архітектури «клієнт-сервер»

Саме трирівнева архітектура клієнт-сервер лежить в основі розробки сайтів, порталів, в яких використовуються реляційні БД для зберігання даних. Наприклад, численні Internet-магазини, пошукові й довідкові сервери, системи для Internet-телефонії й обміну повідомленнями в реальному часі, системи передавання відео через Internet тощо.

Для програмних засобів, що функціонують на основі протоколу HTTP, характерним є те, що користувачі працюють з ними за допомогою звичайних браузерів, таких, як Microsoft Internet Explorer. При цьому доступ до них здійснюється як до звичайних статичних HTML-сторінок за їх URL-адресою.

Однак такі програмні засоби зовсім не схожі на статичні HTML-сторінки. За посередництвом Web-сервера вони можуть звертатися до різних об'єктів, служб і систем, таких, наприклад, як бази даних. Так, у відповідь на запит, введений користувачем у вікні браузера, Web-сервер може сформулювати звіт і відобразити його у тому ж вікні. При цьому для одержання результату Web-сервер генерує запит до бази даних [1].

В освітній галузі досить активно застосовують Web-технології для подання навчальних курсів в електронному вигляді та їх подальшого використання при самостійному навчанні. Практично кожен електронний курс, підручник, довідник тощо містить у собі достатньо великий обсяг даних, що спричинює труднощі при пошуку потрібного матеріалу, особливо при недостатній його структурованості. Для поліпшення гнучкості роботи з цими даними: пошуку, вилучення чи додавання окремих структурних частин, зручно оформляти матеріал у вигляді бази даних. При цьому підручник чи будь-яке інше навчальне джерело розміщують на спеціально призначеному комп'ютері (Web-сервері), а працювати з ним можна за допомогою невеликої програми-клієнта, в даному випадку браузера. Описана технологія також найкраще підходить для організації дистанційного навчання.

Важливо відзначити, що програми, які базуються на Web-технологіях, можуть працювати не тільки в Internet, але й у локальних мережах. Застосування браузера на комп'ютері користувача, як основного засобу для доступу до баз даних, значно полегшує супровід великих локальних мереж. При цьому не тільки спрощується процедура установки програмного забезпечення на робочі станції мережі, але й полегшується підтримка бази даних та інших систем, що працюють централізовано на спеціально виділених серверах.

Виділимо основні особливості використання Web-клієнта для доступу до БД навчальних ресурсів:

- незалежність від операційної системи і програмного забезпечення, встановленого у користувача (студента, учня);
- практично не потребує технічної підтримки користувачів, наприклад, поновлення старих версій чи встановлення на клієнтських машинах;
- потребує мінімум часу для освоєння інтерфейсу користувача, оскільки при роботі з Web-клієнтом використовуються ті ж навички, що і при роботі зі звичайним сайтом;
- порівняна простота в розробці;
- використання Internet-технологій, що дозволяє розміщувати ресурс як в глобальній мережі, так і локально, не вносячи змін до програмного коду;
- для продуктивної роботи потрібен виділений комп'ютер (сервер) для розміщення СУБД та сервера застосувань.

Отже, архітектуру клієнт-сервер, широко застосовують у сучасних комп'ютерних мережах взагалі та в Internet зокрема. При підготовці вчителів інформатики детально не вивчається дана технологія, а точніше не акцентується на ній увага, хоча вивчаються служби Internet, мережі та бази даних. Щодо останніх, то як у школах, так і у вищих навчальних закладах, що готують учителів, вивчення баз даних переважно базується на вивченні локальних БД, а конкретніше – програми MS Access з пакету MS Office. За традицією вивчення теми «Бази даних. СУБД» будується на основі MS Access, освоєння якої починається після теми «Електронні таблиці». Формальний підхід при вивченні цієї теми неминуче призводить до нерозуміння фундаментальних відмінностей між системою управління базами даних і електронними таблицями [2].

Оскільки при підготовці вчителя інформатики, як правило, не вивчається архітектура клієнт-сервер взагалі та SQL, як засіб доступу до баз даних за технологією клієнт-сервер зокрема, то це впливає і на подальше глибоке розуміння пов'язаного з цим матеріалу, що вивчається в курсі інформатики.

На нашу думку, доцільно, майбутніх учителів інформатики знайомити із клієнт-серверними базами даних, наприклад MySQL. Дана система управління базами даних має ряд переваг у порівнянні з іншими СУБД. Вона є універсальною і розроблена для роботи під різними операційними системами (є версії для Windows, Linux, Solaris, FreeBSD, Mac OS X та інших операційних систем). MySQL є безкоштовною і розповсюджується за ліцензією GNU GPL (general public license). Не дуже складна мова запитів дозволяє поетапно освоювати тему, починаючи з найпростіших запитів до однієї таблиці та закінчуючи серйозними запитами із складною вибіркою, що містять в собі посилання на три-чотири таблиці.

Нами для студентів спеціальностей «Інформатика», «Математика та інформатика» фізико-математичного факультету педагогічного ВНЗ розроблені методичні рекомендації стосовно вивчення СУБД MySQL. При вивченні даної СУБД у комп'ютерному класі на вчительській машині було встановлено сервер MySQL, де централізовано для кожного студента була створена тестова база даних для виконання лабораторних робіт. Також на цей сервер було експортовано три заповнених бази даних, одна з яких була багатотабличною, на яких студенти відпрацьовували виконання SQL-запитів. Для з'єднання із сервером використовували дві різних програми-клієнти MySQL-Front та PhpMyAdmin. Друга, оскільки є Web-клієнтом, може використовуватися практично під будь-якою операційною системою і не потребує встановлення. MySQL-Front є звичайною прикладною програмою під Windows. Для виконання всіх запропонованих завдань підходять обидва клієнти, і студенти обирали, яким користуватися, на власний розсуд. При вивченні даних програмних продуктів акцент зроблено на використанні SQL – як універсальної мови запитів до більшості баз даних, також розглянуто технологію ODBC, яка теж є доволі універсальною для доступу до багатьох СУБД при створенні власних додатків.

ЛІТЕРАТУРА

1. Аткинсон Л MySQL. Библиотека профессионала — М.: Издательский дом «Вильямс», 2002. — 624 с.
2. Щепакіна Т.Є Роль змісту навчальних задач до теми «Бази даних. СУБД» у формуванні інформаційної культури учня // Комп'ютерно-орієнтовані системи навчання: Зб. наук. Праць / Редкол. – К.:НПУ ім. М.П. Драгоманова. – Випуск 7. – 2003.
3. www.mysql.com.