

4.4. СИСТЕМИ УПРАВЛІННЯ БАЗАМИ ДАНИХ

Системою управління базами даних (СУБД) називають програмну систему, призначену для створення на ЕОМ загальної бази даних для безлічі додатків; підтримка її в актуальному стані і забезпечення ефективного доступу користувачів до даних, що містяться в ній, в рамках наданих їм повноважень. СУБД призначена, таким чином, для централізованого управління базою даних як соціальним ресурсом на користь всієї сукупності користувачів.

Практично неможливо уявити інформаційну підтримку сучасної установи без застосування професійних СУБД. Проте існуючий сьогодні рівень можливостей програмних продуктів даного напрямку був досягнутий не відразу: еволюція СУБД пройшла шлях від систем, що спиралися на ієрархічну і мережну модель даних, до систем, так званого, третього покоління, для яких характерні ідеї об'єктно-орієнтованого підходу.

СУБД першого покоління мали ряд істотних недоліків: відсутність стандарту зовнішніх інтерфейсів і можливості перенесення прикладних програм. Проте ці СУБД виявилися вельми довговічними: розроблене на їх основі програмне забезпечення використовується і сьогодні, при цьому великі ЕОМ (mainframe) містять величезні масиви актуальної інформації.

Розробка Е.Коддом реляційної теорії сприяла створенню наступного класу СУБД. Особливостями другого покоління є застосування реляційної моделі даних і розвинена мова запитів SQL. Простота і гнучкість моделі даних дозволили стати їй домінуючою і зайняти лідируючі позиції на відповідному секторі ринку.

Багато розробників сьогодні визначають ряд негативних моментів в реляційній моделі, серед яких можна виділити неможливість уявлення і маніпулювання даними складної структури (тексти, просторові дані). Це примушує вести роботи щодо вдосконалення систем другого покоління або створення нової моделі даних. Для СУБД третього покоління характерні використання пропозицій, що стосуються управління об'єктами і правилами, управління розподіленими даними, мов програмування четвертого покоління (4GL), технології тиражування даних і інших досягнень у області обробки даних. Сьогодні СУБД цього покоління застосовуються в діловій сфері достатньо активно не тільки як незавершені технічні рішення, а й як готові продукти, що дають можливість розробникам активно використовувати могутні засоби управління даними.

Системи управління базами даних можна класифікувати:

- щодо мови спілкування, котра використовується:
 - а) замкнуті власні самостійні мови спілкування користувачів з БД; вони забезпечують безпосереднє спілкування з системою в режимі діалогу, дозволяють працювати без програмістів;
 - б) відкриті, в яких для спілкування з БД використовується мова програмування, «розширена» операторами мови маніпулювання даними (МД); в цьому випадку необхідна присутність кваліфікованого програміста;
 - за числом рівнів моделей даних, що підтримуються СУБД:
 - а) одно-, дво-, трьохрівневі системи. Теоретично обґрунтований вибір трьохрівневої архітектури даних, проте на практиці СУБД для персональних ЕОМ часто об'єднують концептуальний і внутрішній рівні уявлення;
 - щодо функцій, які виконуються:
 - а) операційні, припускаючи інші види обробки щодо отримання інформації, котра не зберігається в певному вигляді в БД;
 - б) інформаційні, дозволяючи організувати зберігання даних, пошук та видачу потрібних даних з БД, підтримувати їх доцільність і актуальність;
 - щодо сфери застосування:
 - а) універсальні, які настроюються на будь-яку предметну область через створення відповідної БД і прикладних програм;
 - б) проблемно-орієнтовані на певні процедури обробки даних, властивих конкретній області застосування;
 - щодо допустимих режимів роботи:
 - а) пакетні;
 - б) з використанням телеобробки.
- Серед основних функцій системи управління базами даних, виділяються наступні:
1. Управління даними в зовнішній пам'яті. Функція управління даними в зовнішній пам'яті включає забезпечення необхідних структур як для зберігання безпосередніх даних, так і для службових цілей, наприклад, для прискорення доступу до даних (звичайно використовуються індекси). Існує безліч способів організації зовнішньої пам'яті баз даних. Як і всі рішення, що приймаються під час створення баз даних, конкретні методи організації зовнішньої пам'яті необхідно вибирати разом з ухваленням решти рішень.

2. Управління буферами оперативної пам'яті. СУБД звичайно працюють з базами даних значних розмірів; принаймні цей розмір перевищує доступний обсяг оперативної пам'яті. Зрозуміло, що в разі звернення до будь-якого елемента даних проводитиметься обмін із зовнішньою пам'яттю, то вся система працюватиме зі швидкістю зовнішньої пам'яті. Єдиним способом реального збільшення швидкості є буферизація даних в оперативній пам'яті. І навіть, якщо операційна система проводить загальносистемну буферизацію, цього недостатньо для цілей СУБД, яка має в своєму розпорядженні набагато більшу інформацію про корисність буферизації тієї або іншої частини бази даних. У розвинених СУБД підтримується свій набір буферів оперативної пам'яті з власною дисципліною заміни буферів. При управлінні буферами необхідно розробляти і застосовувати узгоджені алгоритми буферизації, журналізації і синхронізації. Відмітимо, що існує власний напрям СУБД, який орієнтований на постійну присутність всієї БД в ОП. Цей напрям ґрунтується на припущенні, що в подальшому об'єму оперативної пам'яті може бути настільки великий, що дозволить не турбуватися про буферизацію.
3. Управління транзакціями. Транзакція - це послідовність операцій з БД даних СУБД як єдине ціле. А саме транзакція успішно виконується, і СУБД фіксує зміни БД, проведені нею, в зовнішній пам'яті жодна з цих змін ніяк не відображається в стані БД. Транзакція необхідна для підтримки логічної цілісності БД (наприклад, об'єднання елементарних операцій над файлами). Підтримка механізму транзакцій - необхідна умова навіть розрахованих на одного користувача СУБД. Але поняття транзакції набагато важливіше в розрахованих на велику кількість користувачів СУБД. Та властивість, що кожна транзакція починається при цілісному стані БД і залишає цей стан цілісним після свого завершення, робить дуже зручним використання транзакції як одиниці активності користувача по відношенню до БД. При відповідному механізмі управління транзакціями користувач може відчувати себе єдиним користувачем СУБД.
4. Журналізація і відновлення БД після збоїв. Одна з основних вимог до СУБД - надійне зберігання даних в зовнішній пам'яті. Під надійністю зберігання маємо на увазі те, що СУБД - в змозі відновити останній узгоджений стан БД після апаратного або програмного збою. Підтримка надійного зберігання даних в базі вимагає збільшення об'єму пам'яті для зберігання даних, причому та їх частина, яка використовується для відновлення, має зберігатися надійно. Найбільш поширений метод такої підтримки - це ведення журналу змін бази даних. У всіх випадках необхідно дотримання "попереджаючих" записів в журналі (так званий протокол Write Ahead Log). Стратегія полягає в тому, що запис про зміну будь-якого об'єкта БД повинен потрапити в зовнішню пам'ять журналу раніше, ніж вона потрапить в зовнішню пам'ять основної частини БД. Відомо, що коли в СУБД коректно прописаний протокол WAL, то за допомогою журналу можна розв'язати всі проблеми відновлення БД після будь-якого збою.
5. Підтримка мов БД. Для роботи з БД використовуються спеціальні мови, так звані, мови баз даних. У ранніх СУБД підтримувалося декілька спеціалізованих за своїми функціями мов. У сучасних СУБД звичайно підтримується єдина інтегрована мова, що містить всі необхідні засоби для роботи з БД, починаючи від її створення, що забезпечує призначений для користувача інтерфейс з базами даних.