

IV. Економіко-математичні методи

УДК 004.65

А. О. Примостка, аспірант кафедри інформаційного менеджменту, ДВНЗ «КНЕУ імені Вадима Гетьмана»

АРХІТЕКТУРА ІНФОРМАЦІЙНО-АНАЛІТИЧНИХ СИСТЕМ: ЕВОЛЮЦІЯ ТА ТИПОЛОГІЯ

Анотація. Досліджено основні типи архітектури інформаційно-аналітичних систем: централізована, розподілення файлів, архітектура клієнт—сервер; виявлено їх переваги та недоліки, обґрунтовано пріоритети вибору інтеграційного базису для побудови прикладних систем.

Ключові слова: інформаційно-аналітична система, архітектура, клієнт, сервер.

Аннотация. Исследованы основные типы архитектуры информационно-аналитических систем: централизованная, разделения файлов, архитектура клиент—сервер; выявлены их преимущества и недостатки, обоснованы приоритеты выбора интеграционного базиса для построения информационно-аналитических систем.

Ключевые слова: информационно-аналитическая система, архитектура, клиент, сервер.

Annotation. The main types of the architecture of information-analytical systems have been researched: centralized, files distribution, client-server architecture; their advantages and disadvantages have been shown, the priorities of integration basis for building information-analytical systems have been proved.

Key words: information-analytical system, architecture, client, server.

Постановка проблеми. У ХХІ столітті світове господарство перейшло до якісно нової стадії свого розвитку, яка ґрунтується на інформаційних технологіях і комп'ютерних мережах. Відтак виникає потреба швидко обробляти все більше інформації протягом все коротшого часу, оперативно і адекватно реагувати на появу нових даних, зберігати великі інформаційні масиви. Все це зумовлює якісні зміни в системах управління економічними процесами на всіх ієрархічних рівнях. Для України, яка перебуває в процесі економічних і політичних реформ, питання модернізації інформаційно-комунікаційних технологій у сфері управління є

особливо актуальним. Вирішення цієї проблеми об'єктивно вимагає всебічного використання сучасних комп'ютерних технологій і засобів телекомунікації на всіх рівнях управління. Проте, впровадження нових технологій ще не гарантує швидкої та зручної роботи з ними, адже різні програмні продукти потребують різних методів підключення до них. Також не існує єдиного методу представлення, зберігання та доступу до даних. Тому виникає проблема формування інтеграційного базису для побудови цілісної інформаційно-аналітичної системи, яка дає можливість користувачам працювати через Internet з будь-якої апаратної платформи. Таким чином, актуальність теми дослідження визначається необхідністю підвищення ефективності розробки та функціонування інформаційно-аналітичних систем з метою забезпечення якості та своєчасності прийняття управлінських рішень на всіх ієрархічних рівнях.

Аналіз останніх джерел та публікацій. Розвиток підходів до побудови універсальної архітектури інформаційно-аналітичних систем відбувається протягом останніх п'ятдесяти років. Дослідженню цієї проблематики присвятили свої праці багато вітчизняних науковців [1—4], не залишаються осторонь і зарубіжні учені та практики [5—10]. В процесі розвитку інформаційних технологій сформувалися різні концепції та підходи до побудови архітектури інформаційно-аналітичних систем, які відрізняються за можливостями доступу до сховищ і баз даних, логікою представлення, рівнем автоматизації, кількістю користувачів [1—3]. Зараз поширення набула триланкова архітектура, яка має різноманітні модифікації і дозволяє підтримувати інтеграцію з різними спеціалізованими програмними продуктами [3, 4, 9]. Однак проблема формування інтеграційного базису для цілісної інформаційно-аналітичної системи потребує подальших досліджень і практичних розробок, а пошуки у напрямі створення універсальної архітектури інформаційно-аналітичних систем тривають.

Виклад основного матеріалу дослідження. Дослідження процесу еволюції архітектури інформаційно-аналітичних систем дозволяє виокремити такі їх основні типи: централізована, розподілення файлів, архітектура клієнт—сервер, яка, в свою чергу, може бути дволанковою, триланковою і багатоланковою. З погляду історії виникнення найдавнішою є централізована архітектура, яка забезпечує значно масштабніше рішення, ніж дві наступних — розподілення файлів і дволанкова. Прошло три

покоління, перш ніж ці розподілені архітектури змогли ефективно конкурувати з рішенням, орієнтованим на універсальний комп'ютер. Це пояснюється, насамперед, тим, що побудувати розподілену прикладну систему набагато складніше, ніж централизовану. Слід зауважити, що усі типи архітектури, крім централізованої, є архітекторами розподілених систем. Розглянемо докладніше кожен тип архітектури.

Централізована архітектура. За такого підходу одна могутня універсальна ЕОМ слугує єдиною платформою, що виконує всі алгоритми логіки додатка (рис. 1). Централізована архітектура має багато переваг: проста розробка додатків, легкість обслуговування і керування. Саме ці переваги і забезпечили досить тривале функціонування цих систем. Зникнення універсальних ЕОМ неодноразово прогнозувалося після появи чотирьох- і восьмирядних персональних комп'ютерів (ПК) на початку 1980-х років, таких як комп'ютери PET і Vi-20 компанії Commodore, TRS-80 компанії Radio Shack, комп'ютери на базі процесорів Z-80, а також 6502 і 6800 виробництва Motorola. Однак універсальні ЕОМ продовжували працювати, здійснюючи щоденно десятки мільйонів транзакцій і пристосовуючись до постійно зростаючого навантаження.

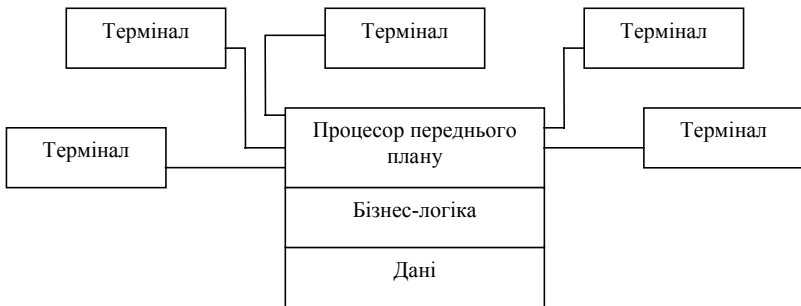


Рис. 1. Централізована архітектура. Блок «Дані» включає алгоритми доступу до даних, СУБД і власне базу даних

Архітектура розподілення файлів. Поява ПК супроводжувалася очікуванням того, що велика кількість малих комп'ютерів може замінити, а в окремих випадках і перевершити за продуктивністю, універсальну ЕОМ. Першим кроком до реалізації цього проекту стала архітектура розподілення файлів, що включає фай-

ловий сервер і багато настільних ПК, які зв'язані мережею (рис. 2). Файловий сервер завантажує файли з відокремленого місця розташування ПК, а прикладні програми виконуються цілком на ПК. Архітектура розділення файлів набула популярності при використанні програмних продуктів на зразок dBASE, FoxPro і Clipper. Спочатку мережі ПК були засновані на ідеї спільного використання файлів, тому що це найпростіший варіант. Однак така мережа добре працювала лише в окремих випадках, адже мала низку обмежень.

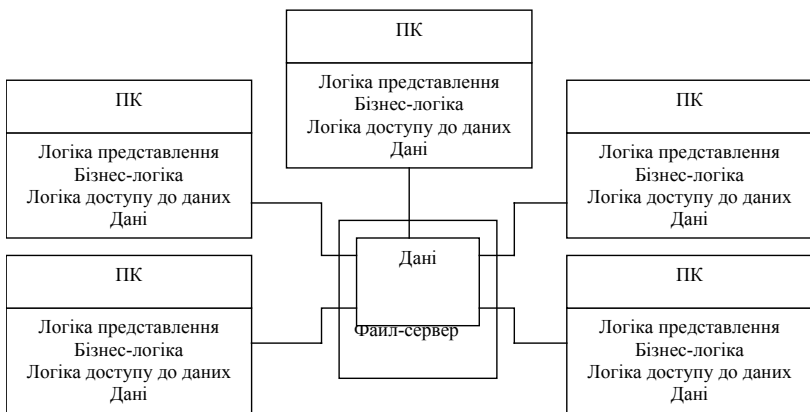


Рис. 2. Архітектура розділення файлів

По-перше, за такого підходу усі додатки повинні вписатися в єдиний ПК. По-друге, спільне використання файлового серверу багатьма користувачами, а також виникнення конфліктних ситуацій відновлення надзвичайно знижують продуктивність. По-третє, пропускна здатність такої мережі незначна, а отже й обсяг даних, що можуть передаватися, також невеликий. Усі ці недоліки суттєво обмежують кількість рівноправних користувачів, що їх здатна підтримувати архітектура розподілення файлів.

Архітектура клієнт—сервер. Прагнення до удосконалення архітектури розподілення файлів привело до заміни файлового сервера сервером баз даних. Замість передачі файлів у цілісній формі сервер баз даних пересилає тільки відповіді на запити клієнтів, зменшуючи в такий спосіб навантаження на мережу. Реалізація цього підходу привела до появи у 1980-х роках архітектури

клієнт—сервер (рис. 3). Розробка цієї архітектури супроводжувалася введенням поняття «клієнта» як сторони, котра запитує функції/обслуговування, та поняття «сервера» як сторони, яка надає функції/обслуговування. На рівні програмного забезпечення поділ «клієнт — сервер» був цілком логічним, адже процеси клієнта і сервера можуть фізично розміщуватися як на одному, так і на різних комп'ютерах.

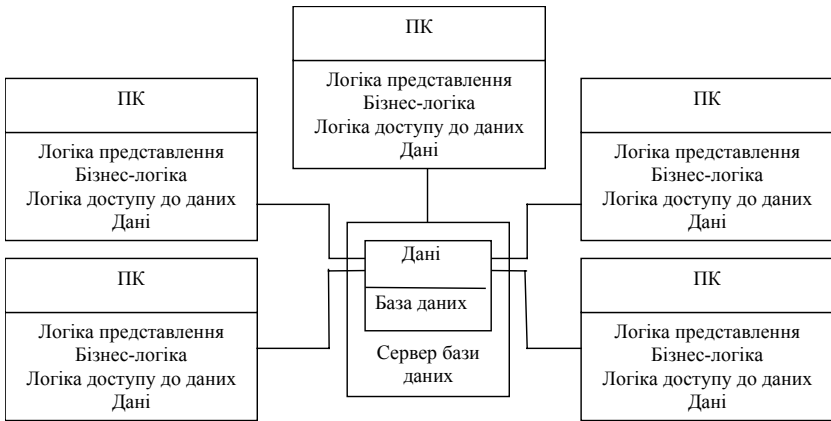


Рис. 3. Ранні архітектури клієнт—сервер

Слід підкреслити, що загальна назва архітектури клієнт—сервер включає три типи архітектури: дволанкову, триланкову і багатоланкову. Серед цих типів найперше була розроблена дволанкова архітектура.

Дволанкова архітектура розділяє додаток на дві частини, клієнтську і серверну (рис. 4). Клієнтська частина містить логіку представлення, а логіка доступу до даних, система управління базою даних і сама база даних розміщується у серверній частині. Алгоритми бізнес-логіки можуть розміщуватися як на комп'ютері клієнта разом з логікою представлення (конфігурація «товстий клієнт»), так і на стороні сервера (конфігурація «тонкий клієнт»), а також можуть бути розподілені між ними. На практиці більш поширена конфігурація «товстий клієнт», оскільки сумарна обчислювальна потужність клієнтів, принаймні теоретично, перевищує потужність єдиного сервера.

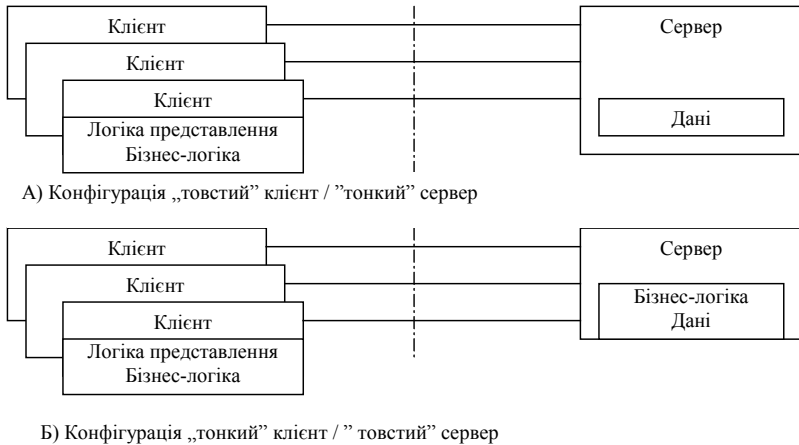


Рис. 4. Дволанкова архітектура

Разом з тим, дволанкова архітектура (особливо конфігурація «товстий клієнт») має низку недоліків. Як і архітектура розділення файлів, дволанкова архітектура має обмеження, які впливають з потужності окремих обчислювальних машин клієнтів. Фундаментальний недолік стосується обмеження кількості одночасних з'єднань із сервером, який підтримує відкрите з'єднання з усіма активними клієнтами, навіть якщо жодної діяльності з їх боку немає. Це необхідно, щоб сервер одержував сигнали тактового імпульсу. Якщо клієнтів менше ста, цей недолік не дуже явний. Однак, якщо кількість клієнтів перевищує сотню, продуктивність системи починає швидко деградувати до неприпустимо низького рівня. У деяких прикладних системах бізнес-логіку реалізують за допомогою збережених процедур, хоча ці механізми від системи до системи можуть відрізнятися. Розглянуте дозволяє зробити висновок, що дволанкова архітектура добре працює в маленьких робочих групах.

Триланкова архітектура. Із середини 1990-х років визнання фахівців отримала триланкова архітектура, яка, як і дволанкова, підтримує концепцію клієнт-сервер, але за рахунок розподілу системи за функціональним призначенням між трьома ланками: логікою представлення, бізнес-логікою і логікою доступу до даних (рис. 5).

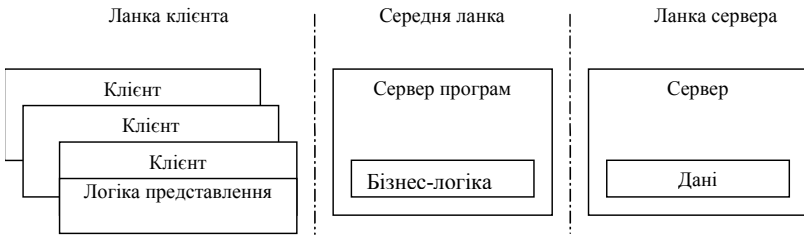


Рис. 5. Триланкова архітектура

За такого підходу з'являється додаткова ланка — «сервер додатків», яка цілковито призначена для реалізації бізнес-логіки. Саме виокремлення бізнес-логіки в окрему ланку дозволяє подолати фундаментальне обмеження дволанкової архітектури. За такого підходу клієнти не підтримують постійного з'єднання з базою даних, а обмінюються інформацією із середньою ланкою тільки тоді, коли це необхідно. Разом з тим, процесор середньої ланки підтримує усього кілька активних з'єднань з базою даних, але використовує їх багаторазово, тому теоретично процеси в середній ланці можуть обслуговувати необмежену кількість клієнтів. Порівняно з іншими підходами триланкова архітектура має багато переваг. Однак, як відомо, переваги не даються задарма: розробка прикладних програм, заснованих на триланковій архітектурі, суттєво складніша порівняно з дволанковою чи централізованою. Подолати виникаючі проблеми допомагає програмне забезпечення проміжного рівня.

Виділяють чотири варіанти побудови інформаційно-аналітичних систем, заснованих на триланковій архітектурі: із сервером додатків, з монітором обробки транзакцій, із сервером передачі повідомлень, із брокером об'єктних запитів. Усі варіанти базуються на перенесенні частини функціоналу на проміжні програмні комплекси, тому для прикладу розглянемо тільки сервер додатків. У триланковій архітектурі з сервером додатків ланка клієнта містить тільки логіку представлення, тоді як алгоритми бізнес-логіки і логіки доступу до даних переміщені в середню ланку (рис. 6). За такого підходу сервер програм є сховищем бізнес-правил і процедур, а клієнти з'єднуються із сервером програм і надають йому дані для обробки.



Рис. 6. Триланкова архітектура з сервером програм

Слід підкреслити, що спільне використання алгоритмів бізнес-логіки, загальних для всієї системи, має важливі переваги. Насамперед, ця стратегія дозволяє створити систему, в якій майбутні відновлення прикладного програмного забезпечення будуть здійснюватися на сервері програм, що спрощує процес модифікацій. Зазвичай сервер програм підтримує пул обмеженої кількості відкритих підключень до баз даних, тому замість того, щоб робити кожне підключення до бази даних виділеним для визначеного клієнта (як у дволанковій архітектурі), підключення багаторазово використовуються для виконання запитів різних клієнтів.

Триланкова архітектура має й низку інших переваг. По-перше, завдяки тому, що вся важлива частина прикладної логіки зосереджена в середній ланці, зникає необхідність підтримувати складні механізми аутентифікації на стороні клієнтів. По-друге, апаратна платформа, на якій виконується сервер програм, за такого підходу може бути досить могутньою, що забезпечує додатковий ступінь масштабованості всієї прикладної системи. По-третє, централізований доступ до даних у серверах програм робить усю прикладну систему менш залежною від конкретної системи управління базою даних. По-четверте, сервер програм забезпечує ефективну стратегію для інтеграції, адже дотримання однакового з клієнтом протоколу комунікації забезпечує взаємодію зовнішньої програми з чужим сервером програм. Надзвичайно важливою перевагою є те, що така конфігурація допускає інтеграцію

програм не тільки на рівні даних, але й на рівні правил бізнес-логіки. В загальному випадку спільне використання даних різними прикладними програмами може вести до логічних протиріч у базі даних. Типове рішення полягає в тому, щоб копіювати ті самі правила й алгоритми в кілька прикладних програм. Однак у такому разі значно ускладнюється їхня підтримка і відновлення, адже будь-яка зміна коду повинна проводитися у всіх прикладних програмах, що його використовують. Якщо ж бізнес-правила зосереджені на сервері програм і використовуються спільно, то такої проблеми не виникає.

Інші варіанти триланкової архітектури засновані на комбінуванні методів пріоритетності запитів клієнтів, виокремлення функцій балансування навантаження в окремий програмний комплекс та створення кількох синхронізованих серверів додатків, що комбінуються як у послідовному, так і в паралельному режимах. За результатами вивчення особливостей функціонування та характеристик різних типів архітектури інформаційно-аналітичних прикладних систем визначено їхні переваги та недоліки (табл. 1).

Таблиця 1

**ПЕРЕВАГИ ТА НЕДОЛІКИ РІЗНИХ ТИПІВ АРХІТЕКТУРИ
ПРИКЛАДНИХ СИСТЕМ**

Тип архітектури	Максимальна кількість користувачів	Переваги	Недоліки
Централізована	Тисячі	Простота, достатньо висока масштабованість	Орієнтація на певну апаратну платформу; недостатня гнучкість
Розподілення файлів	Десятки	Простота	Надмірна обмеженість
Дволанкова	Сотні	Достатня простота; дуже потужні інструменти розробки	Дуже мала кількість користувачів
Триланкова	Необмежена	Висока масштабованість; висока продуктивність і надійність; основа для багатократного використання існуючих компонентів; гнучкість	Велика складність з погляду розробки програмного забезпечення

Висновки з проведеного дослідження. Вивчення та аналіз підходів до побудови архітектури інформаційно-аналітичних систем показав, що кожен з існуючих типів архітектури має свої переваги та недоліки. За результатами проведеного дослідження зроблено висновок, що практично незалежно від форми побудови, програмного забезпечення, апаратного забезпечення, що використовуються, найбільше визнання отримала триланкова архітектура побудови інформаційно-аналітичної системи. З огляду на останні тенденції розвитку інформаційних технологій необхідно будувати системи з кількома внутрішніми серверами для формування інтеграційного базису для цілісної інформаційно-аналітичної системи та програмним забезпеченням, що дає користувачам системи можливість працювати з такими системами через Internet з будь-якої апаратної платформи.

Література

1. Інформаційні системи в економіці: монографія // за заг. ред. д-ра екон. наук, проф. С.В. Устенка. — К.: КНЕУ, 2012. — 425 с.
2. Козак І.А. Інформаційні технології віртуальних організацій // Навчальний посібник. — К.: КНЕУ, 2005. — 380 с.
3. Ситник Н.В. Проектування сховищ і баз даних // Навчальний посібник. — К.: КНЕУ, 2004. — 348 с.
4. Шаховська Н.Б., Пасічник В.В. Сховища та простори даних // Монографія. — Л.: Видавництво Львівської політехніки, 2009. 244 с.
5. Журкин И. Г., Шайтура С. В. Геоинформационные системы. — М.: КУДИЦ-ПРЕСС, 2009. — 272 с.
6. Коголовский М.Р. Перспективные технологии информационных систем. — М.: ДМК Пресс; Компания АйТи, 2003. — 288 с.
7. Codd E.F., Codd S.B., Salley C.T. Providing OLAP (On-Line Analytical Processing) to User-Analysts: An IT Mandate. // E.F.Codd & Associates, 1993.
8. Leon, Alexis. Enterprise Resource Planning. — 2nd. — New Dehli: McGraw-Hill, 2008. — С. 224. — 500 с.
9. Meer, Kamran H. Best Practices in ERP Software Applications. — Lincoln, NE: iUniverse, 2005. — 232 с.
10. Hamilton, Scott. Maximizing your ERP system: a practical guide for managers. — McGraw-Hill, 2003. — 392 с.

Статтю подано до редакції 22.04.12 р.