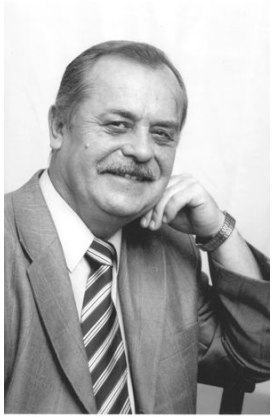
УДК 004.65

Основні компоненти інформаційного сховища



Георгій Асєєв,
доктор технічних наук, професор, завідувач кафедри інформаційних технологій ХДАК

Розглянуті основні компоненти інформаційних сховищ: програмне забезпечення проміжного шару, транзакційні БД і зовнішні джерела інформації, рівень доступу до даних, завантаження й попереднє опрацювання, рівень інформаційного доступу й опрацювання запитів і подання даних, рівень управління (адміністрування), інтегрованість, історичність і стабільність, складні багатомірні дані, що уможливлюють розуміння участі людини в цьому процесі.

У роботах [1—5] порушені різні питання методології розроблення архівів і сховищ даних електронних документів. У пропонованій статті будуть розглянуті такі компоненти цих інформаційних сховищ:

Програмне забезпечення (ПЗ) проміжного шару. Забезпечує мережний доступ і доступ до баз даних. До нього належать мережні й комунікаційні протоколи, драйвери, системи обміну повідомленнями тощо [6].

Транзакційні БД і зовнішні джерела інформації. Бази даних OLTP-систем історично призначалися для ефективного опрацювання структур даних у відносно невеликій кількості чітко визначених транзакцій. Через обмежену цільову спрямованість "облікових" систем застосовувані в них структури даних погано підходять для систем підтримки прийняття рішень. Крім того, вік багатьох установлених OLTP-систем (системи оперативного опрацювання транзакцій) сягає 10—15 років. Нині джерелами даних сховища є оперативні транзакційні системи, які обслуговують повсякденну облікову діяльність компаній. Необхідність включення тієї або іншої транзакційної системи як джерела визначається бізнес-вимогами до системи підтримки прийняття рішень (СППР). Виходячи із цих самих вимог, як джерела даних можуть бути розглянуті зовнішні системи, у тому числі й Інтернет. Детальні дані із джерел або прямо надходять у сховище, або попередньо агрегуються до необхідного рівня узагальнення.

Рівень доступу до даних. ПЗ цього рівня забезпечує спілкування кінцевих користувачів з інформаційним сховищем і завантаження необхідних даних із транзакційних систем. Універсальною мовою спілкування нині є мова структурованих запитів (SQL).

Завантаження й попереднє опрацювання. Цей рівень містить у собі набір засобів для завантаження даних з OLTP-систем і зовнішніх джерел. Виконується, як правило, у поєднанні з додатковим опрацюванням: перевіркою даних на чистоту, консолідацією, форматуванням, фільтрацією тощо. Ця компонента є ПЗ, що відповідно до певного регламенту витягає дані із джерел і приводить їх до єдиного формату, визначеного для сховища, й відповідає за формалізовану логічну узгодженість, якість та інтеграцію даних, які завантажуються із джерел в оперативний склад даних. Кожне джерело даних вимагає розробляння власного завантажувального модуля. Кожен модуль має вирішувати два класи завдань:

- початкового завантаження ретроспективних даних;
- регламентного поповнення сховища даними із джерел.

Ця компонента також витягає детальні дані за регламентом з оперативного складу[1], здійснює їхнє агрегування, консолідацію, трансформацію й поміщає дані в сховище та вітрини даних. Саме в цій підсистемі потрібно визначити всі бізнес-моделі консолідації даних за ієрархічних вимірів і здійснити обрахунок залежних бізнес-показників за незалежними вихідними даними.

Інформаційне сховище. Є предметно-орієнтованою базою або сукупністю БД, які витягаються із джерел, організованих по сегментах, що відбивають конкретну предметну галузь бізнесу: виробництво, правила, детальні слабко агреговані дані.

Метадані. Метадані (депозитарій, "дані про дані"). Відіграють роль довідника, що містить відомості про джерела первинних даних, алгоритми опрацювання, яким вихідні дані були піддані тощо. Метадані — це будь-які дані про дані. Вони відіграють важливу роль у побудові СППР. Одночасно це один з найскладніших і недостатньо практично пророблених об’єктів. Можна виділити, принаймні, три аспекти метаданих, що мають бути присутні у системі.

1. З погляду користувачів: метадані для бізнес-аналітиків; для адміністраторів; для розроблювачів.

2. З погляду предметних областей: структури даних сховища; моделі бізнес-процесів; опису користувачів; технологічні та ін.

[1] У літературі існують різні визначення цього класу даних. Зокрема, оперативним складом даних можна вважати технологічний елемент зберігання даних у СППР, що служить буфером між транзакційними джерелами даних і сховищем. Як було відзначено раніше, дані, перш ніж вони потраплять у сховище, потрібно перетворити в єдині формати, очищені, об'єднані й синхронізовані. Наприклад, дані, необхідні для підтримки ухвалення рішення, можуть існувати в транзакційній системі коротший час (години, дні), ніж період поповнення даних сховища (дні, тижні). Або семантично однорідні дані надходять із транзакційних систем у різний час. У цьому випадку оперативний склад даних служить акумулятором даних, що надходять від джерел, перед їхнім завантаженням у сховище. На відміну від сховища даних, інформація у складі даних може змінюватися згодом відповідно до перетворень, що відбуваються в джерелах даних. Оперативний склад даних створюється як проміжний буфер між оперативними системами й сховищем даних. Ця конструкція аналогічна конструкції сховища даних. Ідентичність оперативного складу й сховища даних полягає в їхній предметній орієнтованості й зберіганні детальних даних. Відмінність від сховища даних полягає в тому, що оперативний склад даних має змінюваний зміст, містить тільки детальні дані та їхні поточні значення.
Детальні дані — це дані з оперативних і зовнішніх систем, не узагальнені й не підсумовані, тобто дані, що не змінили своєї семантики. З оперативних систем і зовнішніх джерел дані надходять в оперативний склад, проходячи процеси трансформації.
Дані оперативного складу регулярно оновлюються. Щоразу, коли дані змінюються в оперативних системах і зовнішніх джерелах, відповідні їм дані з оперативного складу також повинні бути змінені. Частота відновлення оперативного складу залежить як від частоти відновлення джерел, так і від регламенту завантаження даних у склад (http://www.usm.md).

© *Асєєв Г., 2010*

3. З погляду функціональності системи: метадані про процеси трансформації; з адміністрування системи; про додатки; про подання даних користувачам.

Присутність трьох перерахованих аспектів метаданих має на увазі, що, наприклад, у прикладних користувачів й розроблювачів системи буде різне бачення технологічних аспектів трансформації даних із джерел: прикладні користувачі — семантику, склад і періодичність поповнення сховища даними із джерела, розроблювачі — ER-діаграми, правила трансформації й інтерфейс доступу до даних джерела.

Рівень інформаційного доступу. Забезпечує безпосереднє спілкування користувача з таким сховищем даних за допомогою стандартних систем маніпулювання, аналізу й подання даних типу MS Excel, MS Access, Lotus тощо.

Рівень опрацювання запитів і подання даних. Оперативний склад, сховище й вітрини даних є інфраструктурою, що забезпечує зберігання й адміністрування даних. Їхньому добуванню, аналітичному опрацюванню й поданню кінцевим користувачам слугує спеціальне ПЗ. Як правило, можна виділити три типи таких ПЗ.

Програмне забезпечення регламентованої звітності, що характеризується заздалегідь визначеними запитами даних та їхнім поданням бізнес-користувачам. Від цього ПЗ не вимагається швидкої реакції. З міркувань економії витрат на підвищення ефективності ПЗ найбільше підходить технологія ROLAP (реляційна OLAP).

Програмне забезпечення нерегламентованих запитів користувачів — основний спосіб спілкування бізнес-аналітиків зі сховищем, при якому кожен наступний запит до даних і вид їхнього подання визначається, як правило, результатами попереднього запиту. Для додатків цього типу потрібна висока швидкість опрацювання запитів (одиниці секунд). Аналізоване ПЗ реалізується технологією MOLAP (багатомірна OLAP) і спеціальними інструментами побудови складних нерегламентованих запитів з інтуїтивно зрозумілим для бізнес-аналітиків графічним інтерфейсом.

Програмне забезпечення добування знань, що реалізує складні статистичні алгоритми й алгоритми штучного інтелекту, призначене для пошуку схованих у даних закономірностей, подання цих закономірностей у вигляді моделей і різноманітного прогнозування по них розвитку ситуацій за схемою "Що, якщо...".

Звичайно, такий розподіл має вельми умовний характер, а межі між відповідними додатками можуть бути розмиті [7].

Рівень управління (адміністрування). Відслідковує виконання процедур, необхідних для відновлення інформаційного сховища або підтримання його стану. Тут програмуються процедури підкачування даних, перебудови індексів, виконання підсумкових (підсумовуючих) розрахунків, реплікації даних, побудови звітів, формування повідомлень користувачам, контролю цілісності тощо. Ця підсистема виконує всі завдання з підтримування системи й забезпечення її усталеної роботи та розширення. Можна виділити, принаймні, такі класи завдань, вирішення яких забезпечуватиме ця підсистема:

1) Адміністрування даних, що включає регулярне поповнення даних із джерел, якщо необхідно, ручне уведення, звіряння й корегування даних в оперативному складі. Адміністрування даних ведеться, як правило, бізнес-користувачами, а відповідальність розподіляється за предметно-орієнтованими сегментами.

2) Адміністрування сховища даних. У завдання адміністрування сховища входять усі питання, пов'язані з підтриманням архітектури сховища, забезпеченням його ефективної та безперебійної роботи, захистом і відновленням даних після збоїв.

3) Адміністрування доступу до даних забезпечує супровід профілів користувачів, розмежування доступу до конфіденційних даних, захист інформації від несанкціонованого доступу.

Адміністрування метаданих системи. Сховище даних відіграє, у першу чергу, роль інтегратора й акумулятора історичних даних. Структура організації сховища орієнтована на предметні області. Предметно-орієнтоване сховище містить дані, що надходять із різних оперативних БД і зовнішніх джерел. Сховищем є сукупність даних, що має такі характеристики: орієнтованість на предметну область або низку предметних областей; незалежність; інтегрованість; історичність і стабільність; залежність від часу (підтримання хронології); сталість.

Орієнтованість на предметну область. Перша особливість сховища даних полягає в його орієнтованості на предметний аспект. Предметна спрямованість контрастує із класичною орієнтованістю прикладних додатків на функціональність і процеси. Додатки завжди оперують функціями, такими, наприклад, як укладання угоди, кредитування, виписування накладної, зарахування на рахунок тощо. Сховище даних організоване навколо фактів і предметів, таких, як угода, сума кредиту, покупець, постачальник, продукт тощо.

Незалежність. Виділене інформаційне сховище істотно зменшує навантаження на OLTP-системи з боку аналітичних додатків, тим самим продуктивність існуючих систем не знижується, а на практиці відбувається скорочення часу відгуку й поліпшення доступності систем.

Інтегрованість. Найважливішим аспектом сховища даних є інтегрованість його даних, що проявляється в багатьох аспектах: у погодженості імен, одиниць виміру змінних, структур даних, фізичних атрибутів даних тощо.

Існує контраст між інтеграцією даних у сховищі даних і в прикладному оточенні. Першою причиною можливої неузгодженості додатків є наявність значної кількості засобів розроблення. Кожен з них диктує певні правила, частина з яких властива винятково певному засобу. Не секрет, що кожен розроблювач віддає перевагу одним засобам розроблення над іншими. Якщо розроблювачами застосовуються різні засоби розроблення, то, як правило, використовуються індивідуальні особливості засобів, а значить, виникає ймовірність неузгодженості між створюваними системами.

Друга причина можливої неузгодженості додатків полягає в існуванні значної кількості способів їхньої побудови. Спосіб побудови конкретного додатка залежить від стилю розроблювача, від часу, коли цей додаток був розроблений, а також від низки факторів, що характеризують конкретні умови розроблення додатка. Все це відбивається на використовуваних способах виконання завдання ключових структур, способах кодування, позначення даних, фізичних характеристиках даних тощо. Таким чином, якщо два розроблювачі створюють різні способи побудови додатків, велика ймовірність того, що повної узгодженості між системами не буде.

Інтеграція даних за одиницями виміру атрибутів полягає в тому, що розроблювачі додатків до питання про спосіб виконання завдання параметрів продукції можуть підходити різними шляхами. Параметри можуть задаватися в сантиметрах, дюймах тощо. Яким би не було джерело даних, якщо інформація надійде в сховище, її потрібно наводити в єдиних одиницях виміру, прийнятих як стандарт у сховищі.

Історичність і стабільність: OLTP-системи оперують актуальними даними, строк застосування та зберігання яких, звичайно, не перевищує величини поточного бізнес-періоду (півроку—рік), у той час, як інформаційне сховище даних спрямоване на довгострокове зберігання інформації

протягом 10—15 років. Стабільність означає, що фактична інформація в сховищі даних не оновлюється й не віддаляється, а тільки спеціальним способом адаптується до змін бізнес-атрибутів. Таким чином, з'являється можливість здійснювати історичний аналіз інформації.

Залежність від часу. Всі дані в сховищі в певний момент часу сумісні (несуперечливі). Для оперативних систем ця базова характеристика даних відповідає сумісності даних у момент доступу. Коли в оперативному середовищі здійснюється доступ до даних, очікується, що дані мають сумісні значення тільки в момент доступу до них.

Залежність від часу сховища даних виявляється ось в чому. Дані в сховищі представлені за часовий проміжок від року до десяти. В оперативному середовищі подання даних здійснюється в проміжку від поточного значення до декількох десятків днів. Додатки з високою продуктивністю для забезпечення ефективного процесу транзакцій мають працювати з мінімальною кількістю даних. Отже, оперативні додатки орієнтовані на короткий часовий проміжок.

Друге виявлення залежності сховища даних від часу полягає в його структурі. Кожна структура сховища включає — явно або неявно — елемент часу.

Третє виявлення залежності сховища даних від часу полягає в неухильному виконанні правила — дані, один раз коректно записані в сховищі, не можуть бути оновлені. Сховище даних, з погляду практичного використання, є великою серією моментальних знімків. Природно, якщо моментальний знімок даних був зроблений некоректно, він може бути змінений. Але якщо був отриманий коректний моментальний знімок, то, один раз зроблений, він надалі зміні не підлягає. Оперативні дані, коректні в момент доступу до них, можуть оновлюватися в міру необхідності.

Сталість. Четверта визначальна характеристика сховища даних — це сталість. В оперативному середовищі операції відновлення, додавання, видалення й зміни здійснюються над записами регулярно. Базові маніпуляції з даними сховища обмежені початковим завантаженням даних і доступом до них. У сховищі даних відновлення даних не відбувається. Вихідні (історичні) дані, після того, як вони були узгоджені, верифіковані й внесені в сховище даних, залишаються незмінними й використовуються тільки в режимі читання.

Мають місце важливі наслідки розходження опрацювання даних в оперативному середовищі й опрацювання в сховищі даних. На рівні проектування сховища даних необхідність у підтримці механізмів, що забезпечують коректність відновлень, відпадає — відновлення в сховищі даних не відбувається. Це означає, що на фізичному рівні проектування при вирішенні проблеми нормалізації й фізичної денормалізації доступ до даних може оптимізуватися без будь-яких обмежень. Інший наслідок спрощення роботи з даними сховища стосується технології роботи з ними, яка в оперативному середовищі відрізняється більшою складністю. Вона підтримує функції оперативного резервного копіювання й відновлення, забезпечує цілісність даних, включає механізми вирішення конфліктів і тупикових ситуацій. Для опрацювання інформації в сховищі даних зазначені функції не настільки критичні.

Характеристики сховища даних — орієнтованість на предметну область при проектуванні, інтегрованість даних, залежність від часу й простота управління даними — визначають середовище, що істотно відрізняється від класичного транзакційного середовища.

Джерелом майже всіх даних середовища сховища даних є оперативне середовище. Може виникнути відчуття, що існує величезна надмірність даних в обох середовищах. Однак на практиці надмірність даних у середовищах мінімальна, оскільки:

1. При передачі даних з оперативного середовища в сховище даних вони фільтруються. Багато даних взагалі ніколи не вивантажується з оперативного середовища. У сховище даних передається тільки інформація, використовувана для опрацювання в системі підтримки прийняття рішень.

2. Часовий горизонт у середовищах істотно різниться. Дані в оперативному середовищі завжди є поточними. Дані в сховищі мають хронологію. З погляду часового горизонту перетинання між оперативним середовищем і середовищем сховища даних мінімальне.

3. Сховище даних містить агреговані (підсумкові) дані, які ніколи не включаються в оперативне середовище.

4. Передача даних з оперативного середовища в сховище даних супроводжується фундаментальними перетвореннями. Більшість даних при надходженні в сховище видозмінюється.

Складні багатомірні дані. Традиційні багатомірні моделі даних і методи їхньої реалізації припускають, що: всі факти прямо відображаються на значення вимірів нижчого рівня, причому рівно на одне значення в кожному вимірі; ієрархії вимірів є збалансованими деревами.

Якщо ці припущення не виконуються, то стандартні моделі й системи виявляються неадекватними. Особливо серйозні проблеми викликають комплексні багатомірні дані, оскільки вони не є підсумовуваними (summarizable) — агреговані результати вищого рівня не можна одержати з агрегованих результатів нижчого рівня. Запити за результатами нижчого рівня будуть приносити неправильні дані, або попередні обчислення, збереження й наступне використання їхніх результатів у цьому випадку неможливі. Замість цього агреговані результати потрібно обчислювати безпосередньо з базових даних, що значно збільшує витрати на обчислення.

Підсумовування вимагає застосування розподілених агрегованих функцій і значень ієрархії вимірів. Неформально ієрархія вимірів є "суворою", якщо жодне зі значень вимірів не має більше одного прямого батька, "сюрективною" (onto), якщо ієрархія збалансована, і "покриваюча" (covering), якщо жоден локальний шлях не "перескакує" через рівень. Інтуїтивно це означає, що ієрархії вимірів мають бути збалансованими деревами.

Нерегулярні ієрархії виникають у різних додатках, у тому числі в ієрархії адміністративних структур [10], ієрархії медичних діагнозів та ієрархії концепцій для веб-порталів, подібних Yahoo. Одне з рішень — нормалізувати нерегулярні ієрархії, процес, що передбачає поповнення несюрективних і непокриваючих ієрархій фіктивними значеннями вимірів, і перебудувати набори батьків, для того, щоб вирішити проблеми "несуворих" ієрархій. Це перетворення може виконуватися прозорим для користувача способом.

За 30 років з часу свого виникнення технологія багатомірних баз даних пройшла серйозну еволюцію. Віднедавна вона стала реалізовуватися в рішеннях, призначених для масового ринку, а провідні виробники тепер випускають багатомірні ядра разом зі своїми реляційними базами даних, причому часто без додаткової оплати. Багатомірна технологія стала значно масштабнішою і зрілішою.

Це породжує кілька важливих тенденцій. Дані, які необхідно аналізувати, стають усе більш розподіленими. Приміром, це часто необхідно для виконання аналізу, при якому використовуються дані у форматі XML, одержувані з певних веб-сайтів. Зростаюча розподіленість даних, у свою чергу, вимагає застосування методів, які дають змогу легко додавати нові дані в багатомірні бази даних, тим самим, спрощуючи завдання створення інтегрованого сховища даних. Серед прикладів — автоматична генерація вимірів і кубів з нових джерел даних і методи простого та динамічного очищення даних.

Технологія багатомірних баз даних також застосовується до нових типів даних, які сучасні технології найчастіше не в змозі адекватно аналізувати. Приміром, класичні методики, такі, як передагрегування (preaggregation) не можуть гарантувати оперативність надання відповіді на запити, якщо дані постійно змінюються, як це відбувається, наприклад, коли інформація надходить із датчиків або від об'єктів, що рухаються, таких, як автомобілі, оснащені засобами глобального позиціювання.

Найважливіші методи збільшення продуктивності в багатомірних базах даних — це передобчислення (precomputation). Їхній спеціалізований аналог — передагрегування, що дає змогу скоротити час відповіді на запити, які охоплюють потенційно величезні обсяги даних, у ступені, достатньому для проведення інтерактивного аналізу даних.

Обчислення й збереження або "матеріалізація", зведених обсягів продажів по країнах і місяцях, — приклад передагрегування. Такий підхід уможливлює швидке одержування відповіді на запити, що стосуються загального обсягу продажів, приміром, в одному місяці, в одній країні або у кварталі й конкретній країні одночасно. Ці відповіді можна одержати з попередньо обчислених даних, і немає необхідності звертатися до інформації, розміщеної в сховищі даних.

Сучасні комерційні реляційні бази даних, а також спеціалізовані багатомірні системи, містять засоби оптимізації запитів на ґрунті попередньо обчислених агрегатів (aggregate) і автоматичного переобчислення збережених агрегатів при відновленні базових даних.

Повне передагрегування — матеріалізація всіх сполучень агрегатів — неможливе, оскільки вимагає надто великого дискового простору й значного часу на попередні обчислення. Замість цього в сучасних системах OLAP розробники дотримуються більш практичного підходу до передагрегування, матеріалізуючи тільки вибрані комбінації агрегатів, а потім використовуючи їх для ефективнішого обчислення інших агрегатів. Повторне використання агрегатів вимагає підтримки коректної багатомірної структури даних.

Нарешті, технологія багатомірних баз даних все частіше буде застосовуватися там, де результати аналізу прямо передаються в інші системи, тим самим, *виключаючи участь людини в цьому процесі*. Цей контекст у сукупності з необхідністю постійного відновлення висуває жорсткіші вимоги до продуктивності, яким не відповідає сучасна технологія. Ці питання розглянемо в наступній публікації.

Список використаної літератури

1. *Асєєв Г. Г.* Методологія електронного документообігу: підсистеми автоматизації діловодства і статичні архіви документів / Георгій Асєєв // Вісн. Кн. палати. — 2005. — № 3. — С. 24—27.
2. *Асєєв Г. Г.* Методологія електронного документообігу: динамічні архіви / Георгій Асєєв // Вісн. Кн. палати. — 2005. — № 11. — С. 22—25.
3. *Асєєв Г. Г.* Методи добування та знаходження знань у сховищах даних електронного документообігу / Георгій Асєєв // Вісн. Кн. палати. — 2008. — № 9. — С. 29—31.
4. *Асєєв Г. Г.* Методологія створення сховищ даних: проблема виявлення нового знання методами Knowledge Discovery in Databases / Георгій Асєєв // Вісн. Кн. палати. — 2008. — № 4. — С. 23—26.
5. *Асєєв Г. Г.* Методологія створення сховищ даних: стандарти та моделювання / Георгій Асєєв // Вісн. Кн. палати. — 2009. — № 5. — С. 30—32.
6. *Ковалів С.* Огляд можливостей застосування ведучих СУБД для побудови сховищ даних (DataWarehouse) [Електронний ресурс] / С. Кузнєцов, В. Артем'єв. — Режим доступу : http://www.olap.ru/basic/dbms.asp. — Назва з екрана.
7. *Спирли Э.* Корпоративные хранилища данных. Планирование, разработка, реализация / Э. Спирли. — Т. 1. — М. : Вильямс, 2001. — 400 с.