ТЕМА №2 СХОВИЩА ДАНИХ ТА OLAP – ТЕХНОЛОГІЇ

План

2.1. Концепція сховищ даних
2.2. Технології побудови сховищ даних
2.3. Вітрини та кіоски даних
2.4. OLAP - технологія
2.5. Основні архітектури OLAP - систем
2.6. OLAP - системи та Інтернет - технології

Література:

Основна

Черняк О. І., Захарченко П. В. Інтелектуальний аналіз даних: Підручник. Київ, 2014.

Додаткова

1. Барсегян А. А., Куприянов М. С., Степаненко В. В. Методы и модели анализа данных: OLAP и Data Mining. Санкт-Петербург : БХВ–Петербург, 2004.

2. Дюк В. А., Самойленко А. П. Data Mining: учебный курс. Санкт-Петербург : Питер, 2001.

3. Елманова Н., Федоров А. Введение в OLAP-технологии Microsoft.– М.:Диалог-МИФИ, 2002.

4. Кацко И. А., Паклин Н. Б. Практикум по анализу данных на компьютере. Москва : Колосс, 2009.

5. Zimmerman H. Fuzzy Sets Theory-and Its Applications. Kluwer Academic Publishers, 2001.

2.1. Концепція сховищ даних

Мета концепції сховищ даних – прояснити відмінності в характеристиках даних в операційних і аналітичних системах, визначити вимоги даним, що поміщаються в сховище, визначити загальні принципи і етапи його побудови, основні джерела даних, дати рекомендації по рішенню потенційних проблем, що виникають при їх вивантаженні, очищенні, узгодженні, транспортуванні і завантаженні до цільової базу даних сховища.

Предметом концепції сховищ даних є не аналіз даних, а власне дані, тобто концепція їх підготовки для подальшого аналізу. В той же час концепція сховища даних визначає не просто єдиний логічний погляд на корпоративні дані, а реалізацію єдиного інтегрованого джерела даних.

Розглянемо типові проблеми, що вирішуються за допомогою сховищ даних. До них відносяться, зокрема, аналіз клієнтської бази, аналіз продажів і аналіз доходів, а також управління пасивами і активами та інші.

Аналіз клієнтської бази дозволяє сформувати цільові сегменти клієнтів і використовувати цю інформацію при продажі банківських продуктів і послуг. Цільові сегменти формуються на основі демографічних і фірмографічних відомостей, фінансових показників (наприклад, обороту або прибутку), галузевих ознак і інших параметрів клієнтів.

Аналіз продажів допомагає виявляти тенденції, планувати продажі по продуктах, клієнтах, підрозділам і, виходячи з результатів збуту, будувати механізми стимулювання клієнтських і продуктових підрозділів. Завдяки використанню сховища даних можна отримати інтегроване уявлення про результати продажів і узяти цю інформацію на озброєння при формуванні планів.

Аналіз доходів актуальний для будь-якого банку, причому понад усе затребуваний аналіз в розрізі клієнтів. Дуже важливо також мати уявлення про розподіл доходів по продуктах і послугах, каналах надання послуг і підрозділах банку.

Управління активами і пасивами. За допомогою сховища даних можна проводити ефективний аналіз активів і пасивів і управляти не тільки ними, але і миттєвою ліквідністю банку на основі інструментального і портфельного підходів.

Існують різні типи сховищ даних, які мають свою специфіку.

Фінансові сховища даних. В більшості випадків фінансові сховища даних це сховища, які організації будують в першу чергу.

Сховища даних в області страхування. Сховища даних в області страхування за деякими невеликими виключеннями схожі на інші сховища. Перше виключення (і це особливо справедливо відносно страхування життя) полягає в тому, що тривалість існування наявних сховищ дуже велика. Такі сховища містять дані, які є старими, дуже старими.

Сховища даних для управління людськими ресурсами. Сховища даних для управління людськими ресурсами мають вельми істотні відмінності від інших сховищ. Перша відмінність – число предметних областей. Таке сховище даних неминуче має одну важливу предметну область – це працівник. Практично все інше підпорядковане цій області або займає другорядне положення.

Глобальні сховища даних. Глобальні сховища даних призначені для глобального представлення корпорації. Розрізняють три типи таких сховищ:

– географічно превалююча обробка даних. Наприклад, необхідно інтегрувати бізнес в Гонконзі з бізнесом в Парижі, який у свою чергу слід інтегрувати з Ріо-де-Жанейро, а той – з Нью-Йорком.

– функціонально превалююча обробка даних. Виробнича діяльність повинна бути інтегрована з постачанням, яке необхідно інтегрувати з продажами, а ті – з дослідженнями і так далі.

– галузева превалююча обробка даних. Наприклад, потрібно інтегрувати друкарську справу з консалтингом, який підлягає інтеграції з

бізнесом у сфері медичного устаткування, а той із спеціалізацією в області програмного забезпечення.

Сховища даних з можливостями Data Mining/Data Mining і Exploration. Сховища даних, що підтримують технологію Data Mining і Exploration є гібридом класичних сховищ. Такі сховища використовуються для виконання могутньої статистичної обробки даних. Ці сховища є дуже детальними, глибоко історичними, оптимізованими для статистичного аналізу.

Сховища даних в області телекомунікацій. Відмітна особливість цих сховищ полягає в тому, що вони в значній мірі визначаються даними, що згенеровані в деталях на рівні дзвінка.

2.2. Технології побудови сховищ даних

Ідея, покладена в основу технологій інформаційних сховищ, полягає в тому, що проводити оперативний аналіз безпосередньо на базі інформаційних систем неефективно. Натомість, всі необхідні для аналізу дані витягуються з декількох традиційних баз даних (в основному, реляційних), перетворюються і потім поміщаються в одне джерело даних – сховище даних.

В процесі занурення дані:

Очищаються – усунення непотрібної інформації;

Агрегуються – обчислення сум, середніх;

Трансформуються – перетворення типів даних, реорганізація структур зберігання;

Об'єднуються із зовнішніх і внутрішніх джерел – приведення до єдиних форматів;

Синхронізуються – відповідність одному моменту часу.

Сьогодні, технології побудови сховищ даних є основою для створення повноцінних інтелектуальних система налізу даних, орієнтованих на рішення слабко структурованих задач прийняття рішень, оскільки вони містить дані, що володіють наступними властивостями:

Цілісністю і внутрішнім взаємозв'язком. Хоча дані занурюються з різних джерел, але вони об'єднані єдиними законами іменування, способами вимірювання атрибутів і т.д. Це має велике значення для корпоративних організацій, в яких одночасно можуть експлуатуватися різні по своїй архітектурі обчислювальні системи, що представляють однакові дані по-різному.

Предметною орієнтованістю. Локальні бази даних містять мегабайти інформації, абсолютно не потрібної для аналізу (адреси, поштові індекси, ідентифікатори записів і т.п.). Подібна інформація не заноситься в сховище, що обмежує спектр розглядаємих при ухваленні рішення даних до мінімуму.

Відсутністю часової прив'язки. Оперативні системи охоплюють невеликий інтервал часу, що досягається за рахунок періодичної архівації

даних. Сховища даних, навпаки, містять історичні дані, накопичені за великий інтервал часу (роки, десятиліття).

Доступністю виключно для читання. Модифікація даних не проводиться, оскільки вона може привести до порушення цілісності сховища даних. Оскільки не потрібно мінімізувати час занурення, то структура сховища може бути оптимізована для обробки певних запитів, що досягається за рахунок денормалізації реляційної схеми, попередньої агрегації і побудови найбільш доречних індексів.

Інтегрованість означає, що дані задовольняють вимогам всього підприємства, а не одній функції бізнесу. Цим сховище даних гарантує, що однакові звіти, що згенерували для різних аналітиків, міститимуть однакові результати.

Незмінність означає, що, потрапивши один раз в сховищі, дані там зберігаються і не змінюються. Дані в сховищі можуть лише додаватися

2.3. Вітрини та кіоски даних

У найбільш загальному виді сховища даних можуть бути розбиті на два типи: корпоративні сховища даних (*Enterprise Data Warehouses*) і кіоски або вітрини даних (*Data Marts*).

Корпоративні сховища даних містять інформацію, що відноситься до всієї корпорації і зібрану з безлічі оперативних джерел для консолідованого аналізу. Зазвичай такі сховища охоплюють цілий ряд аспектів діяльності корпорації і використовуються для ухвалення як тактичних, так і стратегічних рішень. Корпоративне сховище містить детальну і узагальнену інформацію, його об'єм може досягати від 50 Гбайт до одного або декількох терабайт. Вартість створення і підтримки корпоративних сховищ може бути дуже високою. Зазвичай їх створенням займаються централізовані відділи інформаційних технологій, причому створюються вони зверху вниз, тобто спочатку проектується загальна схема, і тільки тоді починається заповнення даними. Такий процес може займати декілька років.

Кіоски або вітрини даних містять підмножину корпоративних даних і будуються для відділів або підрозділів усередині організації. Кіоски даних часто будуються силами самого відділу і охоплюють конкретний аспект, що цікавить співробітників даного відділу. Кіоск даних може отримувати дані з корпоративного сховища (залежний кіоск) або, що поширеніше, дані можуть поступати безпосередньо з оперативних джерел (незалежний кіоск).

Прийоми моделювання кіосків (вітрин) даних відрізняються від прийомів моделювання сховищ даних через різні вимоги до структур даних. Якщо основною задачею сховища даних є зберігання консолідованої історичної інформації, то вітрина даних будується з урахуванням вимог по доступу до даних і представлення інформації. Як правило, для моделювання вітрин (кіосків) даних використовуються типи моделі під назвою: схема

«зірка» і схема «сніжинка». Зупинимося докладніше на кожному з цих типів моделей.

Схема *«зірка»* – популярний тип моделі даних для вітрин даних. Дана модель характеризується наявністю таблиці фактів, оточеної пов'язаними з нею таблицями розмірностей. Запити до такої структури включають прості об'єднання таблиці фактів з кожною з таблиць розмірностей. Характеризується високою продуктивністю запитів. Проектується для виконання аналітичних запитів. Характеризується невеликою надмірністю даних і високою в порівнянні з нормалізованими структурами продуктивністю.

Схема *«сніжинка»* використовується для нормалізації схеми «зірка». Вона декілька скорочує надмірність в таблицях розмірності. Одним з достоїнств є швидше виконання запитів про структуру розмірності (запити вигляду «Вибрати всі рядки з таблиці розмірності на певному рівні»), які дуже часто виконуються при аналізі даних, і можуть затримувати хід аналізу. Проте основною відмінністю схеми «сніжинка» є не економія дискового простору, а можливість мати таблиці фактів з різним рівнем деталізації. Наприклад, фактичні дані на рівні дня, а планові – на рівні місяця.

2.4. OLAP - технологія

Власне, аби спростити роботу з багатоцільовими даними і не загрузнути в їх океані, а також уміло перетворити набір кількісних показників на якісні, і застосовується метод *OLAP – On-Line Analytical Processing* (оперативна аналітична обробка). Останній, на відміну від інших способів автоматизації бізнес-діяльності, дає можливість отримати користувачеві «на виході» не готове чітко структуроване рішення, що видається після включення раніше налагодженого майстра обробки форм, а своєрідний матеріал для наочної і, якщо можна так виразитися, творчої оцінки існуючої ситуації. Тому сфера вживання *OLAP* - аналізу зазвичай обмежується менеджерським складом підприємств різних розмірів, якому доводиться часто займатися тактичними і стратегічними завданнями на зразок аналізу ключових показників діяльності і сценаріїв розвитку, маркетинговим і фінансово-економічним аналізом груп товарів або послуг, а також довгостроковим прогнозуванням роботи підприємства або його підрозділів.

В чому ж відмінність *OLAP* - системи від сховища даних? З точки зору користувача, відповідь на це питання досить проста: у мірі предметної структурованості інформації. Працюючи з *OLAP* - додатком, користувач

застосовує звичні економічні категорії і показники – види матеріалів і готової продукції, регіони продажів, об'єм реалізації, собівартість, прибуток і тому подібне. А для того, щоб сформувати будь-який, навіть досить складний запит, користувачеві не доведеться вивчати *SQL*. При цьому відповідь на запит буде отримана протягом всього декількох секунд. Крім того, працюючи

з *OLAP* - системою, економіст може користуватися такими звичними для себе інструментами, як електронні таблиці або спеціальні засоби побудови звітів. Таким чином, якщо сховище даних – в основному об'єкт уваги спеціаліста по інформаційним технологіям, то *OLAP* без перебільшення можна назвати програмним засобом з арсеналу економіста. Адже саме економіст має справу з самими різними аналітичними задачами: маркетинговим аналізом, аналізом продажів, аналізом бюджетних показників, аналізом фінансової звітності і так далі.

Розглянемо деякі сценарії практичного застосування *OLAP* - продуктів.

Аналіз фінансових показників діяльності підприємства. Бухгалтерські системи 1С, БЕСТ, Парус, Інфін, RS-Balance та інші день за днем нагромаджують результати обліку господарської діяльності підприємств. Вони забезпечують розрахунок фінансових показників і випуск звітності для наглядових органів. Проте, фіскальна звітність не призначена для управління організацією. Керівника цікавить динаміка залишків і рух фінансів, структура доходів і їх розподіл по клієнтах, товарах, днях тижня, місяцях, кварталах, за рік і так далі. Аби забезпечити керівників управлінськими звітами, *OLAP*-система налаштовується на базу даних будь-якої облікової системи.

Корпоративна звітність. У розподіленій організації філії регулярно передають дані в центральний офіс. Тут дані потрапляють в єдине сховище. Над ними виконуються додаткові розрахунки, для яких в філіях немає даних. Наприклад, загально корпоративні витрати розносяться на філії, зменшуючи тим самим їх прибуток в звіті про прибутки і збитки. Аби філії могли ознайомитися з остаточними звітами після виконання всіх розрахунків і перевірок, запускається генератор кубів, що є програмним модулем *OLAP* - системи. В результаті для кожної філії генерується окремий мікрокуб, який відправляється по e-mail. Одержувач – співробітник планово-економічного відділу філії – відкриває куб в *OLAP Browser*, аналізує, роздруковує і підшиває звіти.

Аналіз бюджетних даних. Для ведення фінансового планування і обліку фактичного виконання бюджетів підприємства застосовують прикладні модулі у складі комплексних *ERP* - систем Галактика, БЕСТ та ін., спеціалізовані програмні комплекси, наприклад, Контур Корпорація, Бюджет холдингу, Інтальов, Бюджетне управління і тому подібне.

Аналіз складських даних. Інформація про перебування і рух товарів на складі (товарні запаси, терміни зберігання товарів, постачальники і одержувачі продукції, накладні переміщення товарів) міститься в базі даних *OLTP* - модуля складського обліку. Аналіз цієї інформації дає відповіді на питання: «Скільки продукції було куплено замовником Івановим в третій декаді вересня?», «Який оптимальний об'єм активних і резервних запасів по даній товарній позиції?», «Чи існують сезонні коливання за даним типом товарів і яка їх амплітуда?» і тому подібне.

Аналіз відвідуваності Web-сайту. Web - сайт є серйозним маркетинговим інструментом для багатьох компаній. Аналіз поведінки відвідувачів сайту дозволяє оцінити віддачу від маркетингових заходів і рекламних акцій, ефективність застосування on - line сервісів, інтерес до продуктів і послуг компанії і т.д. Для аналізу використовуються дані log-файлів веб - сервера, вивантажені в локальні або реляційні таблиці, або база даних сайту. Оскільки розміри таких баз, як правило, дуже великі, застосовується технологія мікрокубів.

Створення інформаційного сервісу. Електронні біржі і інформаційні агентства публікують на своїх сайтах проспекти біржових індексів, котирування цінних паперів різних емітентів, рейтинги учасників фондового ринку за різними показниками і іншу інформацію у вигляді мікрокубів. Комерсанти знайомляться з актуальними даними з будь-якої точки земної кулі через Інтернет і з допомогою *OlapBrowser* проводять аналіз архівних і поточних біржових зведень і аналітичних довідок. Підтримка інформації в актуальному стані забезпечується за рахунок генерації мікрокубів за розкладом.

2.5. Основні архітектури OLAP - систем

Системи інтелектуального аналізу даних зазвичай володіють засобами надання користувачеві агрегатних даних для різних вибірок з початкового набору в зручному для сприйняття і аналізу вигляді. Як правило, такі агрегатні функції утворюють багатовимірний (і, отже, не реляційний) набір даних (нерідко званий гіперкубом або метакубом), осі якого містять параметри, а ячейки – залежні від них агрегатні дані – причому зберігатися такі дані можуть і в реляційних таблицях, але в даному випадку ми говоримо про логічну організацію даних, а не про фізичну реалізацію їх зберігання. Уздовж кожної осі дані можуть бути організовані у вигляді ієрархії, що представляє різні рівні їх деталізації. Завдяки такій моделі даних користувачі можуть формулювати складні запити, генерувати звіти, отримувати підмножини даних.

Технологія комплексного багатовимірного аналізу даних отримала назву *OLAP (On-Line Analytical Processing*). *OLAP* – це ключовий компонент організації сховищ даних. Концепція *OLAP* була описана в 1993 році Едгаром Коддом, відомим дослідником баз даних і автором реляційної моделі даних. У 1995 році на основі вимог, викладених Коддом, був сформульований так званий тест *FASMI* (*Fast Analysis of Shared Multidimensional Information* – швидкий аналіз розподіленої багатовимірної інформації), що включає наступні вимоги до додатків для багатовимірного аналізу:

Fast (Швидкий). Надання користувачеві результатів аналізу за прийнятний час (зазвичай не більше 5 с), нехай навіть ціною менш детального аналізу;

Analysis (Аналіз). Можливість здійснення будь-якого логічного і статистичного аналізу, характерного для даного додатку, і його збереження в доступному для кінцевого користувача вигляді;

Shared (Розподілений доступ). Розрахований на багато користувачів доступ до даних з підтримкою відповідних механізмів блокувань і засобів авторизованого доступу;

Multidimensional (Багатовимірність). Багатовимірне концептуальне представлення даних, включаючи повну підтримку для ієрархій і множинних ієрархій (це – ключова вимога *OLAP*);

Information (Інформація). Можливість звертатися до будь-якої потрібної інформації незалежно від її об'єму і місця зберігання.

Відзначимо, що ієрархії можуть бути збалансованими (balanced), а також ієрархії, засновані на даних типу «дата-час», і незбалансованими (unbalanced). Типовий приклад незбалансованої ієрархії – ієрархія типу «начальник-підлеглий (її можна побудувати, наприклад, використовуючи значення поля *SalesPerson* початкового набору даних з розглянутого вище прикладу)». Іноді для таких ієрархій використовується термін *Parent-child* hierarchy.

Існують також ієрархії, що займають проміжне положення між збалансованими і незбалансованими (вони позначаються терміном *ragged* – «нерівний»). Зазвичай вони містять такі члени, логічні «батьки» яких знаходяться не на безпосередньо вищестоящому рівні (наприклад, в географічній ієрархії є рівні Country, City і State, але при цьому в наборі даних є країни, що не мають штатів або регіонів між рівнями Country і City.

2.6. OLAP – системи та Інтернет - технології

Одним із сучасних напрямків розвитку систем інтелектуального аналізу даних є об'єднання *OLAP* з технологією *Data Mining* і сховищами даних. Ці всі три технології розвиваються у міру того, як компанії починають усвідомлювати цінність даних. Реляційні бази даних свого часу були революційним рішенням, яке дозволило підприємствам збирати дані з щоденних транзакцій у великомасштабні засоби зберігання. За допомогою *SQL* було можливо виконувати елементарний аналіз цих даних. Коли ж був потрібен складніший аналіз, з'ясувалося, що *SQL* і РБД зовсім не ідеальне рішення. Таблиці були в змозі забезпечувати гнучкіший аналіз, але мали ряд істотних недоліків. Дані, що підлягають імпорту в таблицю з бази даних і сама таблиця були не в змозі ефективно оперувати великими об'ємами даних. З часом все більше і більше компаній почали реалізовувати сховища даних і застосовувати до своїх даних засоби *Data Mining*. Сховище даних забезпечує зберігання очищених корпоративних даних. Дані по транзакціях перевіряються на коректність, категорізуються і потім поміщаються в сховищі. Інструменти Data Mining дозволяють аналітикам підприємств виявити приховані тенденції даних. Інструмент *OLAP* дає можливість

виконувати швидкий і простій аналіз даних. В цілому, користувач-аналітик має уявлення про те, що він збирається знайти в деякому представленні даних. Він просто хоче мати засіб маніпулювання даними щоб найнаочніше відобразити деякі їх аспекти.

Важливим напрямком розвитку систем інтелектуального налізу даних, які набуває широкої популярності є поєднання *OLAP* – засобів та Web – технологій. Динамічні технології, поява яких стала можлива в результаті розвитку World Wide Web, є прекрасною альтернативою традиційним клієнт-серверним *OLAP* - методам. За останній час з'явилися цілий ряд *OLAP* - засобів (їх називають *WEB - OLAP* або *WOLAP*), оснащених Web - можливостями. Вони виконують аналітичні функції, такі як агрегація і деталізація (drill-up і drill-down), а також забезпечують високу продуктивність у поєднанні зі всіма перевагами, які дає Web - додаток. Поява *Web OLAP* - засобів стирає межі, що відокремлюють *OLAP* - ринок від суміжних категорій програмного забезпечення. Web-платформи інтерактивної звітності по своїй функціональності все більше і більше схожі на стандартні *Web OLAP* продукти.

Більшість *WEB - OLAP* додатків використовують загальну архітектуру, в якій клієнтський браузер взаємодіє з HTTP - сервером, що пересилає *HTML* – сторінки. Але крім цього надається ще і проміжне ПО, таке, що зберігається на сервері. Такий компонент може безпосередньо зв'язуватися з Web-браузером або взаємодіяти з HTTP-сервером, який потім повертає браузеру HTML-сторінки з додатковими даним.

WEB-OLAP компонент проміжного рівня виконує набір функцій, які не може забезпечити HTML, а саме:

– взаємодія з базою даних, де знаходиться сховище;

– зберігання станів (попередніх транзакцій бази даних);

– обчислення і буферизація даних, що повертаються на клієнт.

На сьогоднішній день реалізовано декілька різних рішень *WEB - OLAP*, у тому числі на основі технологій *HTML (DHTML)*, Java, ActiveX, а також їх комбінацій. Розглянемо основні типи таких програмних продуктів:

HTML (DHTML) – рішення.

HTML з розширеннями – *CGI*.

HTML з використанням Java-апплетів.

Java або *ActiveX* – компоненти.