

УДК 519.85 004.42
Шумська К.О.;
Бондарчук С.В.;
Галаган В.І., к.військ.н, доцент;
Прокопенко О.С.;
Панадій К.В.

Центр воєнно-стратегічних досліджень Національного університету оборони України.

Матрична модель представлення OLAP систем

Матричная модель
представления OLAP систем

The model of predstavlennya OLAP
of the systems a matrix

Резюме. В статті означені певні теоретичні підходи до представлення OLAP (online analytical processing) систем та опису основних етапів їх роботи. Пропонується один зі способів оптимізації роботи з розрідженими матрицями даних. Поряд із теоретичними викладками наведено практичний приклад елемента побудови кубу даних та процедуру ув'язування його компонентів.

Ключові слова: OLAP система, розріджена матриця, гіперкуб даних, факти, виміри.

Резюме. В статье отмечены определенные теоретические подходы к представлению OLAP систем и описанию основных этапов их работы. Предлагается один из способов оптимизации работы с разреженными матрицами данных.

Ключевые слова: OLAP система, разреженная матрица, гиперкуб, факты, измерения.

Resume. In the articles noted the theoretical approaches to the descriptions of OLAP of the systems and description of the basic stages of their work. One of methods of optimization of work is offered with disperse matrix of information.

Keywords: OLAP system, disperse matrix, hypercube, facts, measurings.

Постановка проблеми. Сучасні інформаційні технології в умовах обробки зростаючих об'ємів інформації зі змінною складною структурою висувають високі вимоги щодо ефективності й оптимізації задач управління базами даними.

Основними механізмами, які спроможні вилучати велику кількість записів із масового набору даних і миттєво обчислювати на їх основі результуючі значення, сьогодні є OLAP системи. Такі системи базуються на інтелектуальному процесі узагальнення деталізованих даних і дозволяють отримати знання з консолідованих за різними аспектами аналізу, взаємопов'язаних фрагментів інформації. Завдяки такій інтелектуальній особливості OLAP системи в останні роки отримали широке розповсюдження в практичних додатках.

Однак, можливості й переваги OLAP систем використовуються на сьогодні не в повній

мірі. Причин тому існує декілька, серед них в тому числі - недосконалість алгоритмів попереднього оперативного формування агрегованих даних, що збільшує час виконання різних запитів користувачів; складність і висока вартість інколи невиправдано складних моделей, програм, пов'язаних із втіленням та експлуатацією OLAP систем та ін. Тому, перспективними є роботи направлені на оптимізацію простих, компактних і вискоефективних аналітичних систем на основі OLAP технологій. Такі дослідження супроводжуються широким вивченням концепцій, алгоритмів, що лежать в основі OLAP. В той же час, напрацювання в даній області, головним чином, відносяться до важливих питань практичного характеру і не задовольняють потреб користувачів у матеріалах, що стосуються теоретичних основ OLAP систем.

Ступінь розробленості проблеми.

Теоретичне описання OLAP систем проводилося фрагментально починаючи з моменту їх появи. Так, разом з терміном OLAP в 1993 р. Э. Коддом були сформовані 12 ознак OLAP даних [1]. Цікавою роботою серед небагатьох є стаття [2] Хрустальова Є.М., яка присвячена механізму агрегації даних OLAP кубів. Більш фундаментальною є робота [3], де наведено представлення OLAP кубів замкнутими решітками або еквівалентними їм Quotient-решітками.

Водночас, в [2] охоплено лише вузьке питання формалізації OLAP-кубів з використанням теорії множин та теорії графів, а в роботі [3] не проводиться вибір представлення OLAP кубів іншими способами, аніж вибраний.

Мета статті - розкриття деяких із практичних питань принципів побудови і вимог, що лежать в основі OLAP систем, з теоретичної точки зору, шляхом створення математичних моделей основних етапів їх роботи. Крім того, ставиться за мету ув'язати наведені теоретичні пояснення етапів роботи OLAP систем з практичним прикладом, де деталізується внутрішня структура та процедура побудови компонентів багатовимірного кубу даних.

Для досягнення практичної мети дослідження використовується фрагмент бази даних (БД) функціонального модуля "Майно" автоматизованої системи управління оборонними ресурсами (АС УОР) Збройних Сил України, яку відносять до класу так званих ERP-систем (*Enterprise Resource Planning System*) – систем планування та управління ресурсами підприємства) [4].

Виклад основного матеріалу.

А. Загальний алгоритм роботи OLAP системи.

В OLAP системах для зберігання даних може бути використано три основних способи [5]:

MOLAP (Multidimensional OLAP) – вихідні й агрегатні дані зберігаються в багатовимірній базі даних.

ROLAP (Relational OLAP) – вихідні дані залишаються в тій самій реляційній базі даних, в якій вони знаходились. Агрегатні дані розміщують в спеціально створеній для їх збереження службовій таблиці, в тій же базі даних.

HOLAP (Hybrid OLAP) – вихідні дані залишаються в тій самій реляційній базі даних, де вони знаходились на початковому етапі, а агрегатні дані зберігаються в багатовимірній базі даних.

В якості системи, що аналізується в статті надалі, використана широко розповсюджена MOLAP система (далі OLAP система). Загальна схема роботи такої OLAP системи включає наступні окремі етапи:

1. Отримання даних у вигляді плоскої таблиці чи результату виконання SQL запиту (у випадку, коли побудова багатовимірного кубу даних OLAP системи відбувається "на льоту", як це реалізується в ROLAP системі) [6]. На даному етапі доцільно звернути увагу на OLAP систему, як на об'єкт алгебраїчної системи. З цієї точки зору множина даних OLAP системи являє собою двомірну матрицю:

$$\mathbf{A}^{(k)} = \sum_{i=1}^M \sum_{j=1}^N a_{ij} \mathbf{I}_{ij} = \{a_{ij}^{(k)}\}, \quad i \in 1, M; \quad j \in 1, N, \quad (1)$$

де a - дані (елементи) матриці \mathbf{A} , M, N - розмірність матриці \mathbf{A} в загальному вигляді (факти та виміри - у випадку OLAP системи).

На етапі 1: $k = 1$. \mathbf{I}_{ij} - матриця розміру $M \times N$, всі елементи якої дорівнюють нулю, крім i, j елементів, які дорівнюють одиниці (позиції ненульових елементів матриці \mathbf{A} відповідають позиціям ненульових елементів матриці \mathbf{I}_{ij}).

2. Перетворення даних до багатовимірного кубу.

В математичному розумінні процедуру перетворення до багатовимірного кубу зручно представити у вигляді блочних матриць [7], подібних (1) при дотриманні правила, що для утворення такої матриці \mathbf{A} із серії матриць $\mathbf{A}_{\alpha\beta}$ необхідно, щоб підмножини матриць із серії з однаковим значенням індексу α мали однакову кількість рядків, а підмножини матриць з однаковим значенням індексу β - однакову кількість стовпчиків. Ці підмножини утворюють відповідно "блочні" рядки і "блочні" стовпчики, які відповідають декільком рядкам чи стовпчикам звичайного запису матриці:

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{pmatrix} = \left(\begin{array}{cc|ccc} a_{11} & a_{12} & b_{11} & b_{12} & b_{13} \\ a_{21} & a_{22} & b_{21} & b_{22} & b_{23} \\ \hline c_{11} & c_{12} & d_{11} & d_{12} & d_{13} \\ c_{21} & c_{22} & d_{21} & d_{22} & d_{23} \\ \hline c_{31} & c_{32} & d_{31} & d_{32} & d_{33} \end{array} \right) \quad (2)$$

де
 $\mathbf{A}_{11} = \{a_{ij}\}, \mathbf{A}_{12} = \{b_{ij}\}, \mathbf{A}_{21} = \{c_{ij}\}, \mathbf{A}_{22} = \{d_{ij}\}$
 $i \in 1, M; j \in 1, N,$

що в загальному випадку прийме наступний вигляд:

$$\mathbf{A} = \left(\begin{array}{cccc} \overbrace{\mathbf{A}_{11}}^{n_1} & \overbrace{\mathbf{A}_{12}}^{n_2} & \dots & \overbrace{\mathbf{A}_{1t}}^{n_t} \\ \mathbf{A}_{21} & \mathbf{A}_{22} & & \mathbf{A}_{2t} \\ \dots & \dots & \dots & \dots \\ \mathbf{A}_{s1} & \mathbf{A}_{s2} & \dots & \mathbf{A}_{st} \end{array} \right) \left. \begin{array}{l} \} m_1 \\ \} m_2 \\ \dots \\ \} m_s \end{array} \right\} m_s, \quad (3)$$

де n, m - відповідні розмірності елементів блочної матриці \mathbf{A} .

Для випадку базису тривимірного простору, який є математичним аналогом побудови OLAP системи, матриця \mathbf{A} представляє тривимірну (кубічну) таблицю даних, елементами якої є блочні матриці. Така матриця набуде вигляду:

$$\mathbf{A} = \|\mathbf{A}_{ijk}\|, \quad (i, j, k = 1, 2, \dots, n). \quad (4)$$

Сукупність елементів матриці (4) з фіксованими значеннями індексу (наприклад i)

називається січною орієнтації даного індексу (наприклад i -го) та прийме вигляд (3), який буде називатися двомірною січною (зрізом кубу даних). Користуючись двомірними січними, можна записати кубічну матрицю (базу даних) у вигляді квадратної чи прямокутної таблиці (двомірні січні при цьому відокремлюються вертикальною чи горизонтальною лінією (у випадку порядку бази $p \geq 3$)):

$$\mathbf{A} = \left\| \begin{array}{cc|cc} \mathbf{A}_{111} & \mathbf{A}_{112} & \mathbf{A}_{211} & \mathbf{A}_{212} \\ \mathbf{A}_{121} & \mathbf{A}_{122} & \mathbf{A}_{221} & \mathbf{A}_{222} \end{array} \right\| \begin{array}{l} \rightarrow (i) \\ \rightarrow (k) \\ \downarrow (j) \end{array} \quad (5)$$

Стрілки вказують напрямок, в якому зростають відповідні індекси (факти, виміри). Для більш наглядного зображення на рис. 1 наведено просторове розміщення елементів в типовій кубічній матриці даних:

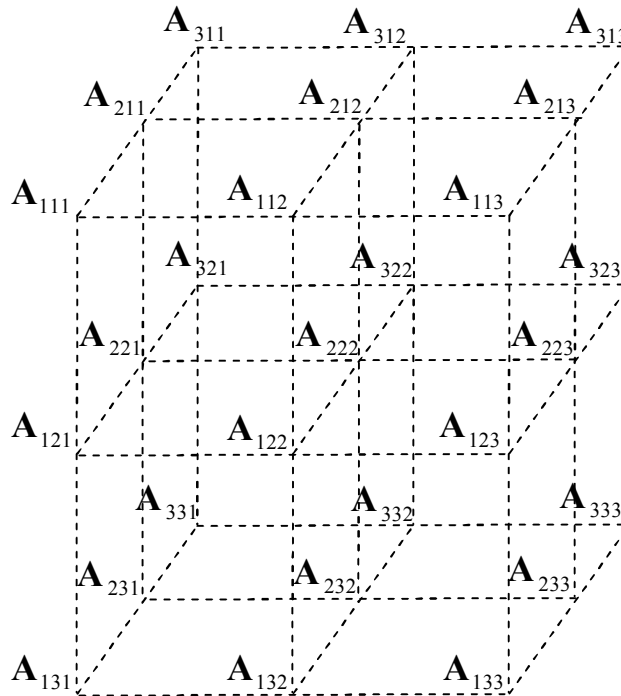


Рис. 1. Структурне зображення матриці даних

3. Відображення побудованого куба за допомогою крос-таблиці чи діаграми і т.п.

У загальному випадку до одного кубу може бути підключена довільна кількість відображень.

Відображення, що використовуються в OLAP системах найчастіше бувають двох видів – крос-таблиці та діаграми.

Необхідно зазначити, що вихідна матриця даних, необхідна на фазі побудови кубу даних, зрізів кубу даних (використовуються під час зазначеного відображення побудованого кубу) буде сильно розрідженою. Тому, для роботи з такою матрицею необхідно застосовувати спеціальні механізми роботи з розрідженими матрицями. Математична інтерпретація одного з таких способів (в загальному випадку), що передбачає “виключення” пустих блоків вказаного масиву, розуміє операцію “викушення” необхідного блоку шляхом перемноження зліва вихідної матриці даних $\tilde{\mathbf{A}}$ на деяку матрицю “вирушення” $\mathbf{E}_m^*(z)$ [8]. Для простоти пояснимо цю операцію на прикладі випадку одновимірної таблиці даних. Нехай матриця $\tilde{\mathbf{A}}$ є розрідженою і має вигляд:

$$\tilde{\mathbf{A}} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \mathbf{A}_{41} & \mathbf{A}_{42} & \mathbf{A}_{43} & \mathbf{A}_{44} & 0 & 0 & 0 & 0 \\ 0 & \mathbf{A}_{52} & \mathbf{A}_{53} & \mathbf{A}_{54} & \mathbf{A}_{55} & 0 & 0 & 0 \\ 0 & 0 & \mathbf{A}_{63} & \mathbf{A}_{64} & \mathbf{A}_{65} & \mathbf{A}_{66} & 0 & 0 \\ 0 & 0 & 0 & \mathbf{A}_{74} & \mathbf{A}_{75} & \mathbf{A}_{76} & \mathbf{A}_{77} & 0 \\ 0 & 0 & 0 & 0 & \mathbf{A}_{85} & \mathbf{A}_{86} & \mathbf{A}_{87} & \mathbf{A}_{88} \end{pmatrix} \quad (6)$$

Проста операція

$$\mathbf{A} = \mathbf{E}_m^*(z) \tilde{\mathbf{A}}, \quad (7)$$

звільнить (“викусить”) повністю розріджений блок. Так, результуюча таблиця буде мати вигляд:

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_{41} & \mathbf{A}_{42} & \mathbf{A}_{43} & \mathbf{A}_{44} & 0 & 0 & 0 & 0 \\ 0 & \mathbf{A}_{52} & \mathbf{A}_{53} & \mathbf{A}_{54} & \mathbf{A}_{55} & 0 & 0 & 0 \\ 0 & 0 & \mathbf{A}_{63} & \mathbf{A}_{64} & \mathbf{A}_{65} & \mathbf{A}_{66} & 0 & 0 \\ 0 & 0 & 0 & \mathbf{A}_{74} & \mathbf{A}_{75} & \mathbf{A}_{76} & \mathbf{A}_{77} & 0 \\ 0 & 0 & 0 & 0 & \mathbf{A}_{85} & \mathbf{A}_{86} & \mathbf{A}_{87} & \mathbf{A}_{88} \end{pmatrix}. \quad (8)$$

У формулі (7) $\mathbf{E}_m^*(z)$ - це $m \times M$ матриця, складена з m суміжних рядків одиничної $M \times M$ матриці \mathbf{I}_M , починаючи з Z -го (в нашому випадку – починаючи з 4-го):

$$\mathbf{E}^*(4) = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}. \quad (9)$$

Ще раз зазначимо, що існує багато інших варіантів організації розрідженої матриці та способів оптимізації роботи з нею, які вимагають більш глибокого їх вивчення й особливостей застосування для конкретного типу зберігання даних.

Б. Порядок підготовки компонентів при створенні кубу даних.

Наведемо практичний приклад структури побудови кубу даних та процедуру ув'язування його компонентів з практичної точки зору. Для цього, як зазначалося, використаємо фрагмент бази даних (БД) функціонального модуля “Житло” АС УОР ЗС України.

Вихідні дані для побудови OLAP кубів, як правило, зберігаються в реляційних базах даних, які також називаються сховищами даних (Data Warehouse) [9]. На відміну від оперативних баз даних, з якими працюють додатки ведення даних, такі сховища даних призначені виключно для обробки і аналізу інформації, тому проектуються вони таким чином, щоб час на виконання запитів до них був мінімальним.

Зазвичай, дані копіюються в сховище із оперативних баз даних відповідно до визначеного регламенту, наприклад, раз на місяць, квартал чи рік. Типова структура сховища даних суттєво відрізняється від структури звичайної реляційної бази даних. Ця структура ненормалізована (це дозволяє підвищити швидкість виконання запитів), тому може допускати надлишковість даних. Основними складовими структури сховищ даних є таблиця фактів (fact table) і таблиця вимірів (dimension tables). Таблиця фактів є основною таблицею сховища даних. Як правило, вона містить відомості щодо об'єктів чи подій, сукупність яких буде в подальшому аналізуватися. Найчастіше зустрічаються чотири типи фактів, до яких відносяться: факти, пов'язані з транзакціями (основані на окремих подіях, наприклад, отримання повідомлення); факти, пов'язані з “моментальними знімками” (основані на стані якогось об'єкта, наприклад, стан оплати оренди по об'єкту нерухомості на кінець місяця чи кінець року); факти, пов'язані з елементами документа (основані на тому чи

іншому документі, наприклад, звітні форми щодо об'єктів нерухомості, рахунки за послуги і т.п.); факти, пов'язані з подіями чи станом об'єкта (за фактом події без уточнення деталей по ній).

Розглянемо структуру таблиці даних (зрізу OLAP кубу) з точки зору OLAP машини. Для OLAP системи колонки таблиці можуть бути або фактами, або вимірами. При цьому логіка роботи з цими колонками буде різною. В гіперкубі виміри фактично є вісями, а значення вимірів – координатами на цих вісях. При цьому куб буде заповнений сильно нерівномірно. При початковій загрузці даних необхідно провести переагрегацію даних, тобто, об'єднати записи, які мають однакові значення вимірів, розрахувавши при

цьому попередні агреговані значення фактів. У результаті такої процедури, в подальшому ми отримаємо можливість працювати з меншою кількістю записів, що підвищить швидкість роботи і зменшить вимоги до об'єму оперативної пам'яті.

Для побудови зрізів гіперкубу необхідні наступні можливості – визначення координат (фактично значення вимірів) для записів таблиці, а також визначення записів, що мають конкретні координати (значення вимірів). Розглянемо, яким чином можна реалізувати ці можливості. З цією метою в табл. 1 наведено фрагмент БД функціонального модуля "Майно" АС УОР України.

Таблиця 1.

Фрагмент набору даних модуля «Майно» АС УОР ЗС України

Виміри				Факти	
Дата	Назва КЕЧ	Ввод	Статус надлишковості	Ідентифікація архітектурного об'єкта	Тип архітектурного об'єкта
29.08.2009	Новоград-Волинська КЕЧ	Petrov	Вивільнено	15/212	02CM
20.09.2009	Новоград-Волинська КЕЧ	Ivachuk	Використовується ЗСУ	21/060	03PR
20.09.2009	КЕВ м.Чернігова	Petrov	Комунальна власність	21/007	04FL
28.08.2009	КЕВ м.Чернігова	Marchenco	Передано в оренду	21/108	04PP
19.08.2009	КЕВ м.Чернігова	Petrov	Підготовка до списання	21/101	05AP

На даному етапі нас цікавлять тільки координати в гіперкубі, тому необхідно визначити координати для значень вимірів. Найпростіше – перенумерувати значення елементів. Для того, щоб в межах одного виміру

нумерація була однозначною, попередньо відсортуємо списки значень вимірів в алфавітному порядку. Крім того, пронумеруємо і факти, причому факти переагреговані, та отримаємо схему, зображену на рис. 2.

	Ввод		Дата		Назва КЕЧ		Статус надлишковості
1	Ivachuk	1	19.08.2009	1	КЕВ м.Чернігова	1	Вивільнено
2	Petrov	2	28.08.2009	2	Новоград-Волинська КЕЧ	2	Використовується ЗСУ
3	Marchenco	3	29.08.2009			3	Комунальна власність
		4	20.09.2009			4	Передано в оренду
						5	Підготовка до списання

	Ідентифікація архітектурного об'єкта	Тип архітектурного об'єкта
1	15/212	02CM
2	21/060	03PR
3	21/007	04FL
4	21/108	04PP
5	21/101	05AP

Рис. 2. Схема відсортування списків вимірів та фактів

Наступним етапом є процес зв'язки елементів різних таблиць між собою. Достатньо кожному запису в таблицях вимірів поставити у відповідність список, елементами якого будуть номери фактів, при формуванні яких використовувались ці виміри (тобто визначити всі факти, що мають однакове значення

координати, яка описана цим виміром). Для фактів кожному запису ставиться у відповідності значення координат, за якими він розташований в гіперкубі. Для наведеного гіпотетичного прикладу отримаємо набір (табл. 2), який визначає внутрішнє уявлення гіперкуба OLAP системи.

Таблиця 2.

Таблиця відповідностей списків вимірів і фактів

Номер	Факти			Виміри		
	Ідентифікація архітектурного об'єкта	Тип архітектурного об'єкта	Назва КЕЧ	Ввод	Статус надлишковості	Дата
1	15/212	02CM	2	2	1	3
2	21/060	03PR	2	1	2	4
3	21/007	04FL	1	2	3	4
4	21/108	04PP	1	3	4	2
5	21/101	05AP	1	2	5	1

Багатовимірний характер побудови кубу даних наглядно показано на рис. 3. Тут виміри – це множина однотипних даних, що утворюють одну із граней гіперкуба (для функціонального модуля “Житло” системи SAP - це можуть бути типи квартир, назва КЕЧ, роки і т.п.). Вони відіграють роль індексів для значень фактів, що

знаходяться всередині гіперкубу. Наприклад, точка в такому просторі буде задавати факт того, що військовослужбовець (ППП, його дані і т.п.) отримав в певній КЕЧ, в певному році квартиру з визначеною кількістю кімнат (площею і т.п.).

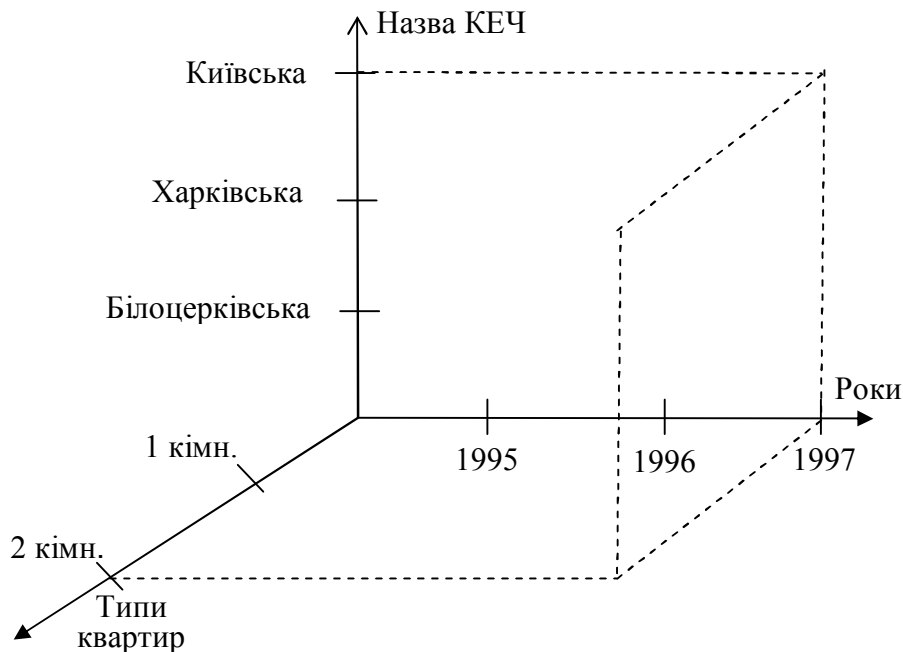


Рис. 3. Просторове представлення OLAP кубу

Висновки.

Принципи побудови OLAP систем забезпечують високий рівень оперативності й ефективності доступу до великих об'ємів інформації в режимі реального часу кількістю користувачів, що може вимірюватись десятками тисяч. Можливість організації і представлення

даних в розрізі різних аналітичних напрямків перетворює дані в цінну інформацію, яка може бути використана для аналітичної обробки інформації і прийняття обґрунтованих рішень.

Показано, що OLAP системи становлять великий інтерес не тільки з практичної, а й з теоретичної точки зору. Питання способів

представлення OLAP кубів є невичерпним на сьогоднішній день і з цією метою, в тому числі, можуть бути використані кроки, запропоновані в даній статті.

Викладений матеріал може бути корисним при застосуванні комплексних підходів аналізу OLAP систем як на етапі повного розуміння їх організації, так і на етапі пошуку шляхів оптимізації роботи, удосконалення способів аналізу та обробки великої кількості даних в OLAP системах.

Подальші дослідження. Подальші дослідження за даною тематикою доцільно зосередити на розширенні означених в даній статті загальних підходів до опису етапів роботи та OLAP систем в цілому. Крім того, додаткової уваги потребують питання оптимізації роботи з розрідженими кубами даних, де поряд з теоретичними дослідженнями повинні бути висвітлені приклади їх реалізації.

СПИСОК ЛІТЕРАТУРИ

1. Codd, E.F. and Codd, S.B. and Salley, C.T.: Providing OLAP (on-line analytical processing) to user-analysts: An IT mandate. In: Codd and Date. 32, 1993.
2. Хрусталеv Е.М. Агрегация данных в OLAP-кубах. Аналитические системы. Информационные технологии. [Электронный ресурс]/ Е.М. Хрусталеv.- Режим доступа: http://www.iteam.ru/publications/it/section_92/article_1759.
3. Кузнецов С.Д., Кудрявцев Ю.А. Математическая модель OLAP-кубов. М.: Академический научно-издательский производственно-полиграфический и книгораспространительный центр Российской академии наук "Наука", 2009, Программирование, том 35, №5, С.26-36
4. Шевченко В.Л. Особливості впровадження та використання ERP-систем як сучасного інструменту управління ресурсами у військовій сфері/ Подобєдов І.В., Козачок В.А. – К, 2009. – Сучасні інформаційні технології у сфері безпеки та оборони №1.
5. Конноли Т., Бегг Л., Странач А. Базы данных // Проектирование, реализация и сопровождение. Теория и практика. 2-е изд. Вильямс. 2000.
6. Бергер А., Горбач И. Меломед Э., Щербинин В., Степаненко В. Microsoft SQL Server 2005 Analysis Services. OLAP и многомерный анализ данных. BHV, 2007.
7. Гантмахер Ф.Р. Теория матриц.- 4-е изд. - М.: Наука Главная редакция физико-математической литературы, 1988. – 552с.
8. Сигорский В.П. Математический аппарат инженера. - 2-е изд. – Киев: "Техника", 1977. - 768 с.
9. Барсегян А.А., Куприянов М.С., Степаненко В.В., Холод И.И. Методы и модели анализа данных: OLAP и Data Mining – СПб.: БХВ-Петербург, 2004. – 336 с.

*Рецензент: Рибидайло А.А. – к.т.н., с.н.с.,
ЦВСД НУО України.
Поступила в редакцію 22.04.13*