УДК 681.3 : 004.6

ПОДАННЯ ДИНАМІЧНИХ ВИМІРІВ У OLAP-КУБАХ

Т.В. Панченко

Київський національний університет імені Тараса Шевченка,
вул. Володимирська 60, Київ, Україна,
тел.: 259 0519, e-mail: t.panchenko@infosoft.ua

Розглянуто два відомих підходи до подання динамічних вимірів: модель з відношенням «багато-до-багатьох» між елементами ієрархії вимірів, а також виміри, що повільно змінюються в часі (Slowly Changing Dimensions). Запропоновано метод побудови незалежних вимірів для подання динамічних вимірів у OLAP-кубах. Обгрунтовано, що метод має меншу складність розробки моделі за рахунок прозорості ідеї. Надано порівняльний аналіз трьох методів.

Two well-known approaches to modelling the dynamic dimensions are considered: a model with many-to-many relationship between the dimension hierarchy elements as well as Slowly Changing Dimensions. A method of independent dimensions creating for the dynamic dimensions modelling in the OLAP-cubes is proposed. It is proved that the method has a lower model development complexity because of idea clearance. The comparative analysis of three methods is given here.

Проблематика. Постановка задачі

Згідно з останніми дослідженнями Gartner [1], більшість компаній (зокрема, зі списку Fortune 500) не здатні ефективно опрацьовувати великі обсяги даних для досягнення конкурентних переваг, тобто не в змозі впоратись з поняттям «big data» [2] як організаційно, так і технологічно [1]. Останнє поняття [2, 3], окрім великих обсягів, на сьогодні також включає велике різноманіття джерел інформації (окрім табличних баз даних, ієрархічних даних, документів, пошти, також мобільні пристрої, соціальні мережі ті інші джерела, які підвищують складність та гетерогенність проблеми, а також роблять важчим гарантування цілісності [3]) та великі швидкості (отримання, передачі та накопичення інформації). Разом з цим, все більше компаній намагаються впровадити та використовувати технології аналізу великих обсягів даних (тобто системи бізнес-аналітики, основою яких є багатовимірні сховища даних – OLAP-куби [4]). Тому дослідження проблем, пов'язаних з багатовимірними сховищами (OLAP-кубами) є актуальною задачею.

OLAP-куб, як відомо, складається з мір (measures) і вимірів (dimensions) [4]. Над вимірами будуються ієрархії. Ієрархії можуть бути з жорсткою кількістю рівнів або гнучкими (так званими "parent-child"). В будь-якому випадку ієрархії являють собою дерево (тобто у кожного елемента може бути тільки один «батько» – елемент, якому він підпорядковується). Також ієрархії можуть бути статичними або динамічними – тобто змінюватись у часі. В останньому випадку починається багато реалізаційних складнощів та теоретичних питань, відкриваючи простір для наукових досліджень та практичних експериментів.

У даній роботі проаналізуємо способи завдання ієрархій вимірів (або на вимірах) та покажемо переваги й недоліки кожного з них.

Динамічні ієрархії

Для прикладу розглянемо наступну ситуацію. Нехай у деякій програмній системі ведеться облік продажів продукції корпорації К на території України. В системі аналізу такої інформації має бути присутнім територіальний розріз, у якого наявні наступні атрибути: точка продажу (адреса), населений пункт (де розташована точка продажу), область і регіон, де під регіонами розуміється певне групування точок продажу (як правило, просто об'єднання декількох областей – наприклад, північний, південний, східний, західний та центральний регіони). Також наявний розріз, що містить інформацію про представників компанії (Sales Person), які курують (і стимулюють) продажі корпорації по регіонах. Останній розріз містить принаймні такі атрибути: ім'я представника та його регіон. Далі, очевидно, мають бути розрізи за продуктовою лінійкою, за часом (дата продажу) тощо. І головне – регіональний розподіл не є сталим у часі, тобто з ростом корпорації і кількості представників регіони можуть стати дрібнішими і їх кількість зросте; може мати місце перерозподіл меж регіонів і таке інше. ER-модель джерела інформації для побудови OLAP-кубу може бути такою, як показано на рис. 1.

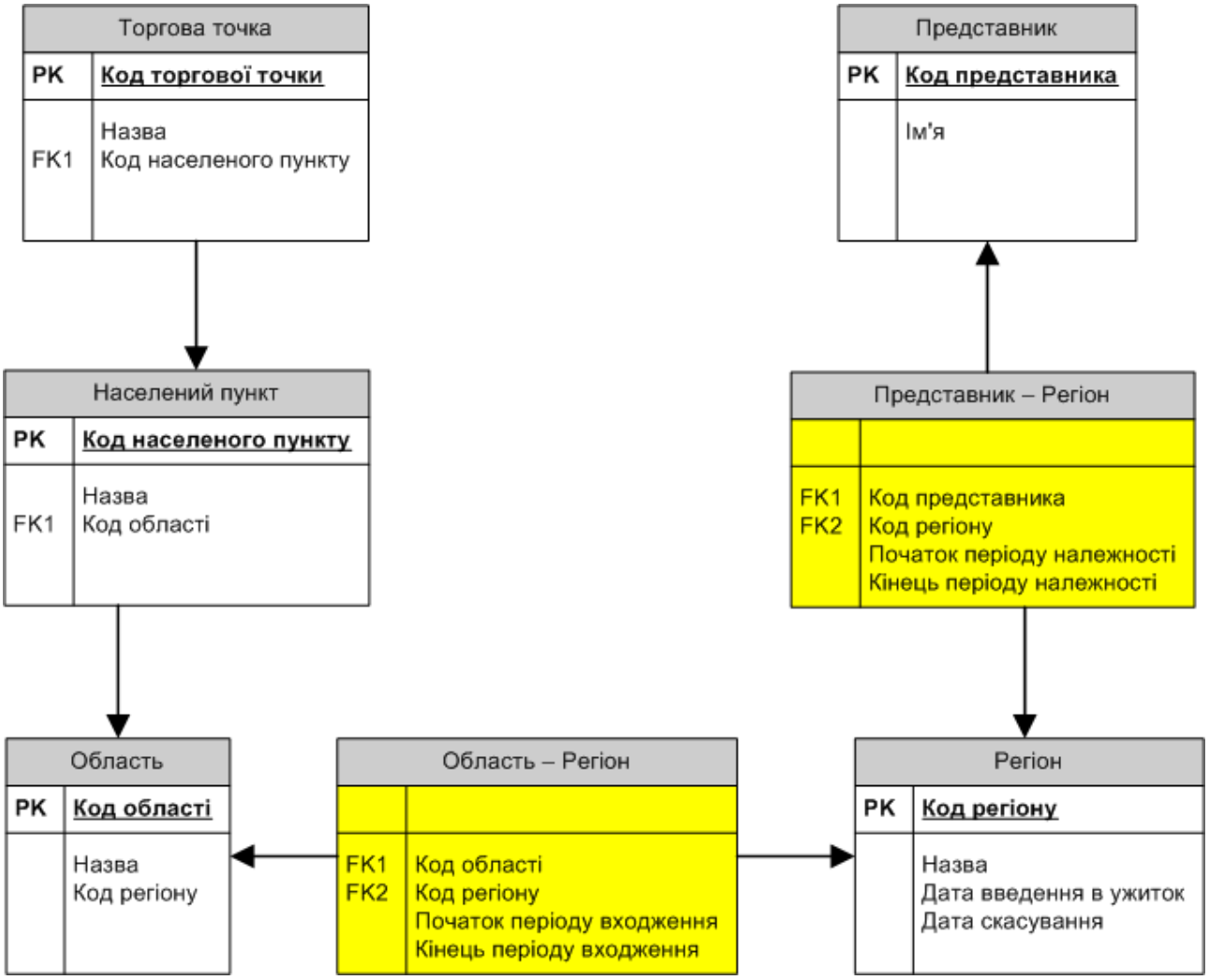

Рис. 1. ER-модель частини джерела інформації для побудови багатовимірного кубу

Далі, згідно з задачею, в аналітичній моделі очікуються наступні ієрархії: регіон – область – населений пункт – торгова точка, а також регіон – представник (див. рис. 2, 3). Причому, зв'язки «область – населений пункт» та «населений пункт – торгова точка» є сталими, а зв'язки «регіон – область» та «регіон – представник» – можуть змінюватись у часі. Будь-які прив'язки до часу і збереження «історичної» інформації ускладнюють структуру системи та призводять до зниження швидкодії опрацювання інформаційних запитів.

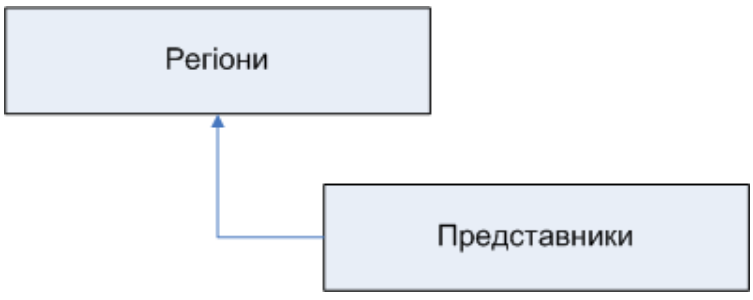

Рис. 2. Організаційна ієрархія

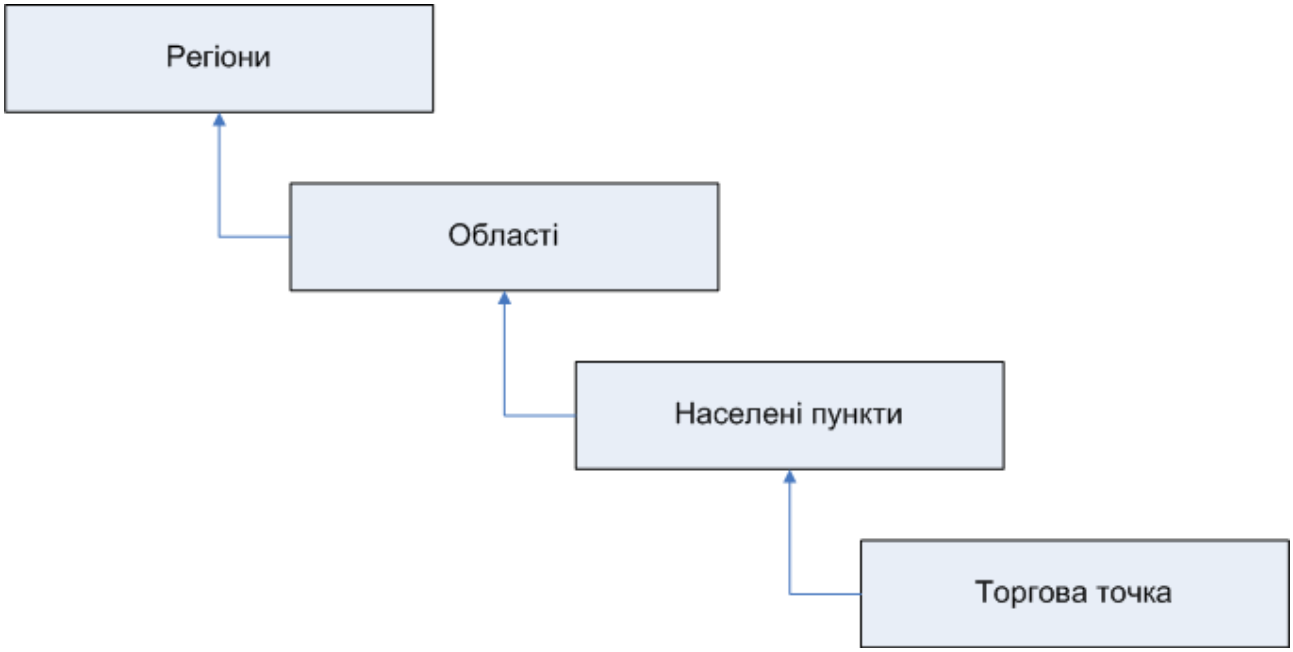

Рис. 3. Територіальна ієрархія

У даному випадку додатковою складністю є те, що ієрархії мають бути деревовидними, однак у нас, фактично, один елемент (область) може мати декілька «батьківських» елементів (регіонів), щоправда, у кожний момент часу – лише одного, чому ми і говоримо саме про *ієрархію*. Іншими словами, ми маємо справу з ієрархіями, що змінюються у часі (з точки зору підпорядкування елементів) або – динамічними ієрархіями. Такого типу ієрархії часто зустрічаються у реальному житті – окрім територіальної прив'язки, що може змінюватись у часі, прикладами є організаційна структура (відношення «керівник – підлеглий» або «департамент – управління – відділ – сектор» за умов реструктуризації), вміст груп (наприклад, студенти навчальних груп або члени команд з певного виду спорту – з часом така належність змінюється, наприклад перехід гравців у інші команди), або склад продуктової лінійки компанії (при реорганізації продуктів за групами просування, нішах або рівнях промоції).

Отже, ми стикаємось з тим, що наведена ER-модель (рис. 1) не вкладається в класичні (деревовидні) ієрархії атрибутів вимірів, а схема багатовимірного сховища не є зіркою [5]. Тому далі є кілька варіантів моделювання даної ситуації:

1) використання відношення багато-до-багатьох для побудови ієрархії [5, 6];

2) скористатись класичним підходом побудови вимірів, що повільно змінюються (Slowly Changing Dimensions, SCD [7]);

3) розглядати «не строго підпорядковані» елементи ієрархії як незалежні виміри.

Тепер розглянемо кожний з варіантів докладніше.

Відношення багато-до-багатьох для побудови ієрархії

Одним із способів завдання динамічних ієрархій є використання відношень багато-до-багатьох у моделі OLAP-кубу. Одразу слід зауважити, що:

1) такий спосіб не є класичним (бо порушується умова деревовидності ієрархії);

2) звідси, підтримується не всіма виробниками систем для бізнес-аналітики (тобто не завжди реалізована підтримка такого виду зв'язків у моделі багатовимірного сховища);

3) нарешті, цей метод є нетривіальним для розуміння і передбачення наслідків, особливо, у складних моделях (кубах з великою кількістю вимірів) [5].

Для нашого прикладу (облік продажів корпорації К) реалізація такого відношення полягатиме у простому відтворенні ER-моделі (див. рис. 4) у моделі багатовимірного сховища, із збереженням зв'язків і, відповідно, порушенням умови деревовидності ієрархій атрибутів вимірів.

Серед всіх можливих застосувань відношення багато-до-багатьох можна виділити такі: належність елемента до багатьох груп одночасно, зміна належності у часі, багато ієрархій «батько – син» на одному вимірі (наприклад, класифікація продукції за різними критеріями – тобто один і той самий товар належить кільком групам (категоріям) за різними критеріями групування) та інші [5].

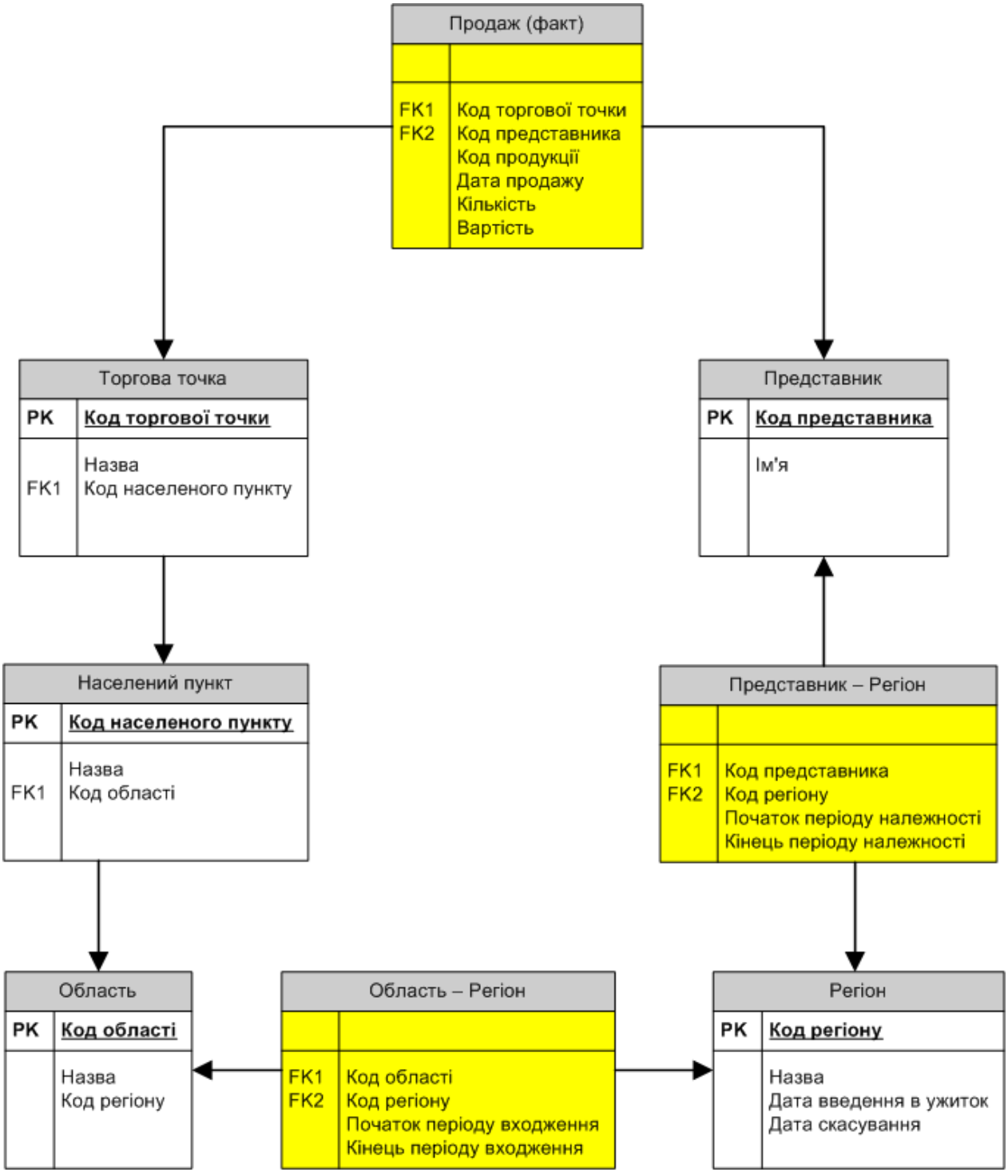

Рис. 4. Модель багатовимірного сховища з використанням відношень багато-до-багатьох

Переваги такого методу – відносно просте моделювання, недоліки – складність розуміння моделі [5] (взаємозв'язки та наслідки) та збільшення ймовірності помилки при її побудові, а також потенційне уповільнення обробки інформаційних запитів (з причини ускладнення внутрішньої реалізації обчислень) [6]. Тобто, в такому випадку складність моделювання трансформується у складність внутрішньої реалізації обчислень у середовищі, яке підтримує систему бізнес-аналітики. Більш того, у випадку часової прив'язки підпорядкування у відношенні багато-до-багатьох, як у нашому прикладі, – ієрархія виміру вже не буде відображатись у моделі «як є», без перетворень (так, модель на рис. 4 не враховує, насправді, часові прив'язки з точки зору багатовимірного сховища, вона навмисно подана у спрощеному вигляді; детальніше про це описано у наступному розділі).

Виміри, що повільно змінюються

Класичним підходом до моделювання вимірів, які змінюються у часі, є методика «вимірів, що повільно змінюються» (Slowly Changing Dimensions, SCD). Цей термін застосовується до вимірів, елементи яких (у тому числі їх підпорядкованість) змінюються відносно повільно, на противагу частим або регулярним змінам за розкладом [7].

У нашому прикладі (облік продажів продукції корпорації К) вимір «регіон – представник» є саме такого типу. Якщо потрібно побудувати звіт у розрізі представників компанії з прив'язкою до їх регіонів, то все просто до моменту, поки представники не будуть переводитись з одного регіону до іншого. Облік представника у двох регіонах (у різні моменти часу – в різних регіонах) стає нетривіальною задачею.

Методика SCD покликана для відтворення таких ситуацій у деревовидних ієрархіях. Існує декілька різних типів SCD. Зокрема, тип 1 зберігає лише поточну прив'язку («регіон – представник»), ігноруючи історичні варіанти. Тип 2 зберігає історичні дані (прив'язки) шляхом створення кількох записів для даного елементу (ідентифікованого за *природнім первинним ключем* представника), які мають різні значення *сурогатного ключа*[1] для кожного запису, або для яких зафіксована версія значення атрибуту. Тип 3 дозволяє зберігати історичні варіанти значень атрибутів у додаткових, спеціально введених, колонках. Застосовуючи SCD типу 2 ми отримаємо найбільш гнучке подання історії зміни значень атрибутів, оскільки для кожного факту зміни значення атрибуту буде додано новий запис в таблицю, де зберігається історія змін.

На рис. 5 можна побачити, як виглядатиме наповнення таблиці «Представник в регіоні» при переведенні представника Іванова з регіону Північний до регіону Центральний з 01.01.2012 р. при застосуванні SCD методики різних типів.

Представник

Код представника	Ім'я
777	Іванов І.В.

Регіон

Код регіону	Назва
1	Північний
2	Південний
3	Східний
4	Західний
5	Центральний

Представник в регіоні (SCD Type 1)

Код представника	Код регіону	Ім'я представника (оригінальне)
777	5	Іванов І.В.

Представник в регіоні (SCD Type 2)

Код представника-в-регіоні	Код представника	Код регіону	Початок періоду належності	Кінець періоду належності	Ім'я представника (оригінальне)
123	777	1	01.01.2000	31.12.2011	Іванов І.В.
125	777	5	01.01.2012	*NULL*	Іванов І.В.

Представник в регіоні (SCD Type 3)

Код представника	Ім'я представника ...	Код початкового регіону	Початок періоду належності	Код поточного регіону
777	Іванов І.В.	1	01.01.2012	5

Рис. 5. Застосування методик SCD типів 1, 2 та 3 до подання динаміки ієрархії «регіон – представник»

У нашому прикладі також є динамічний зв'язок «регіон – область», але з ним ситуація дещо складніша. Так, якщо нам потрібно зберігати і відтворювати всю історію перепідпорядкувань областей в ієрархії регіон – область – населений пункт – торгова точка, то для кожної нової «версії» області (запису про нове підпорядкування області регіону з унікальним значенням сурогатного ключа) ми маємо створювати копію частини ієрархії населений пункт – торгова точка (що входить до складу цієї області), підпорядковуючи новоутворену копію новому запису про область – адже наступні продажі в тих самих торгових точках мають

[1] сурогатний ключ (у даному контексті) – унікальний атрибут або набір атрибутів об'єкта, який однозначно ідентифікує не лише сам об'єкт (поряд зі звичайним, або природним, ключем), а і версію (історичну, часову) даного об'єкта. Як правило, цей ключ не є природнім і генерується самою системою на основі інших даних, причому є прихованим (невидимим і невідомим) для зовнішніх користувачів системи.

відображатись вже у складі нового регіону, а «старі» копії цих торгових точок є «жорстко» прив'язаними до попереднього регіону за ієрархією.

На рис. 6 показано модель багатовимірного сховища з вимірами, що повільно змінюються, (застосування SCD Type 2) для системи обліку продажів корпорації К. Зірочкою (*) відмічено ключі, які відрізняються від аналогічних оригінальних (див. рис. 4), натомість являючи собою:

- ключ «*копії*» торгової точки в межах «*копії*» населеного пункту, підпорядкованого «*історичній версії*» області у складі регіону (*в певний момент часу*), та
- ключ «*копії*» населеного пункту, який підпорядкований «*історичній версії*» області у складі регіону (*в певний момент часу*),

відповідно.

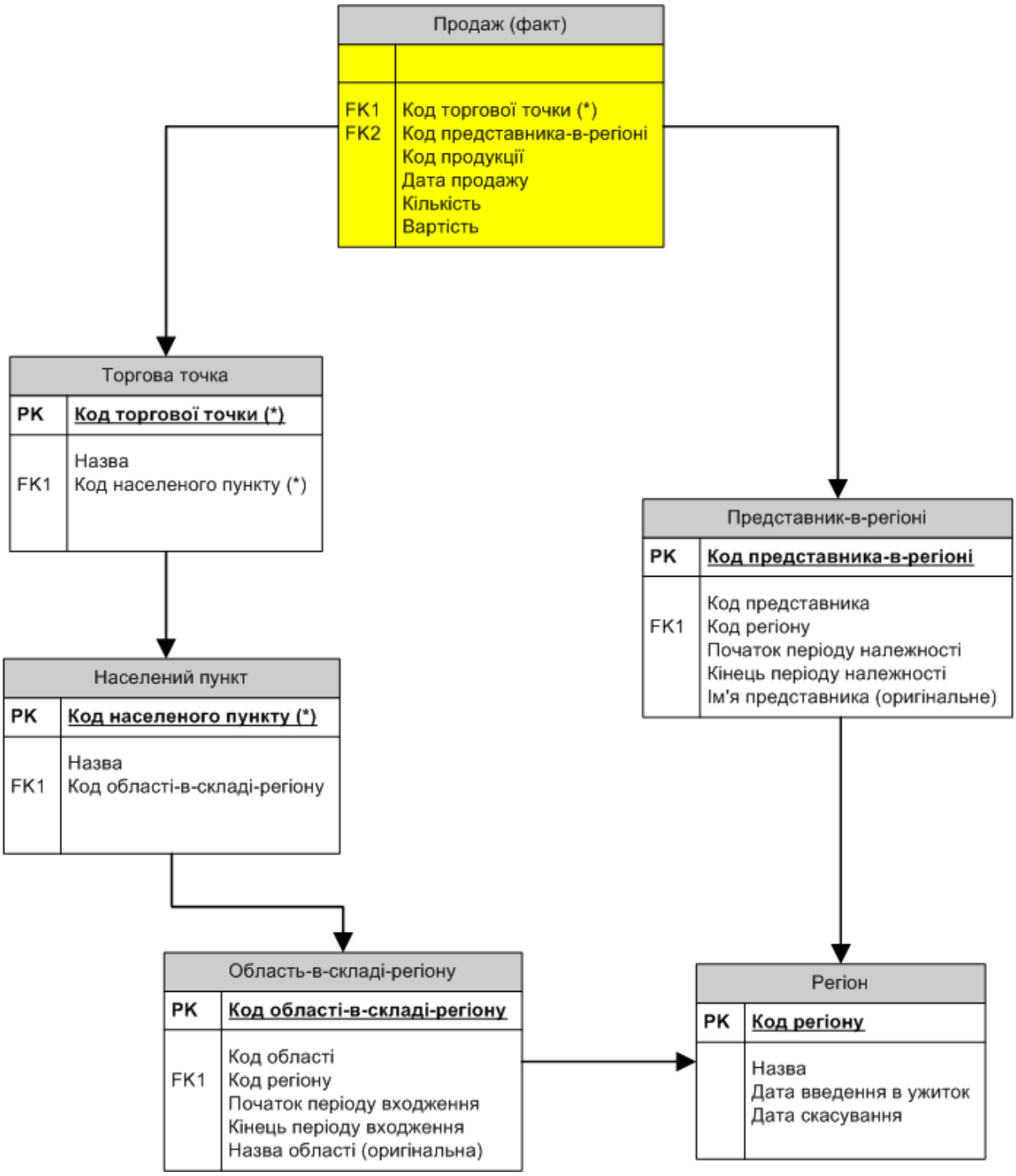

Рис. 6. Модель багатовимірного сховища з вимірами, що повільно змінюються (SCD Type 2)

Таким чином, реалізація найбільш виразної методики SCD типу 2 накладає відбиток на складність подання ієрархії, як наслідок – на швидке зростання OLAP-кубу у розмірах (за рахунок вимушеного дублювання елементів ієрархії на нижніх щаблях) і спадання швидкості опрацювання інформаційних запитів.

Перевага такого методу – визнання як класичного. Він може бути реалізований у будь-якому середовищі, бо складність полягає у підготовці моделі – але далі вона не залежить від реалізаційних особливостей або способу організації багатовимірного середовища.

Незалежні виміри

До побудови моделі можна підійти і «не класично» – «розірвавши» ієрархії на декілька окремих вимірів (рис. 7).

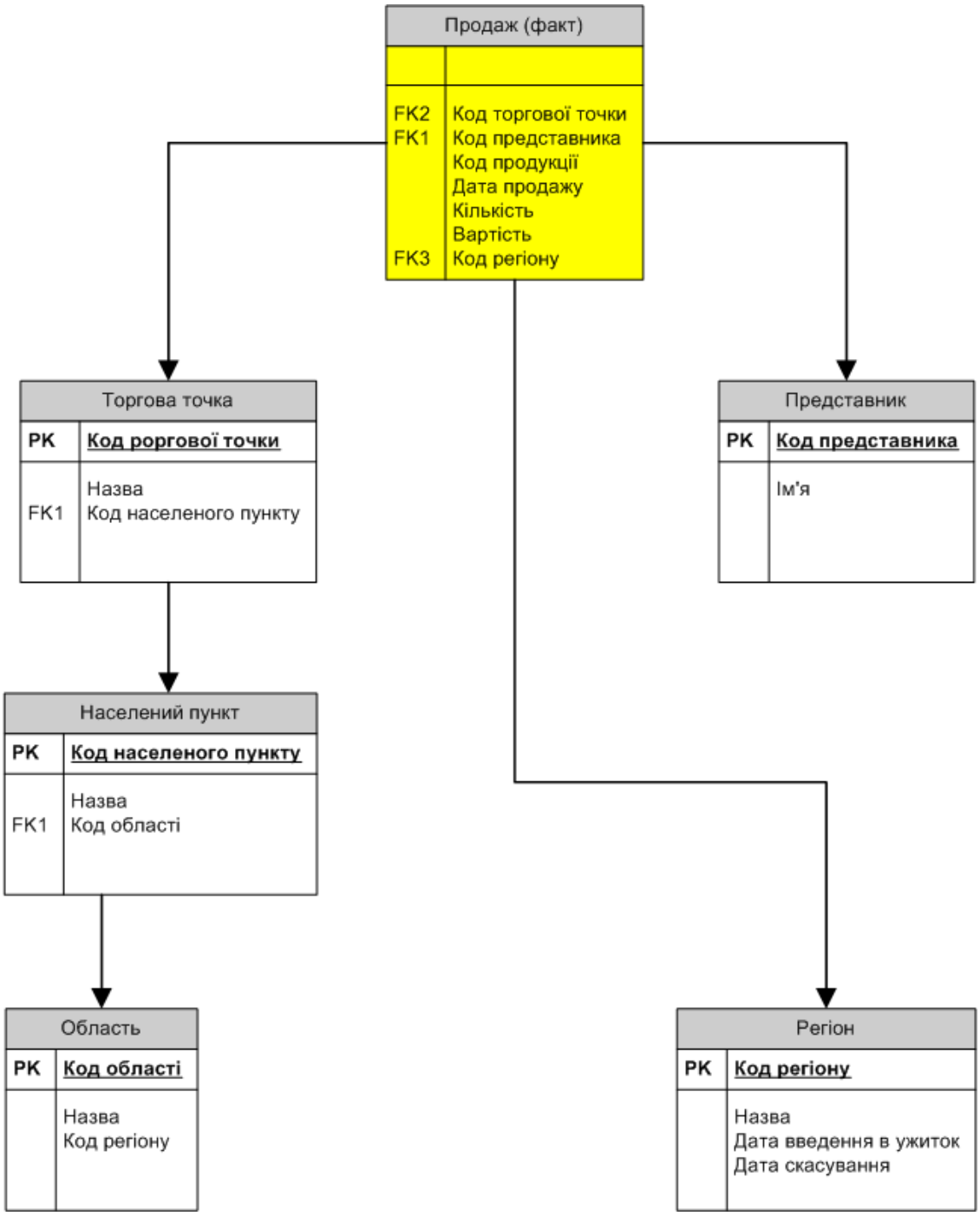

Рис. 7. Модель багатовимірного сховища з незалежними вимірами

При такому підході ми розглядаємо ієрархічні елементи з нестійким (динамічним у часі) підпорядкуванням як незалежні виміри – тобто для нашого прикладу (облік продажів корпорації К) рівень ієрархії «Регіони» виділяється в окремий вимір OLAP-кубу. Тоді для аналізу в звичному для кінцевих користувачів вигляді потрібно комбінувати окремі виміри і влаштовувати з них ланцюжки (типу «регіон» + «область – населений пункт – торгова точка») для відтворення природних, звичних ієрархій.

Щодо адекватності такого подання (розірвання зв'язку з реального світу при побудові моделі), то прагматично це – виправданий крок, адже *стала* функціональна залежність між рівнями ієрархії відсутня.

Перевагами такого підходу є простота реалізації, відтворюваність у довільному середовищі та відсутність необхідності складних попередніх обробок при побудові, розгортанні та обробці (наповненні) моделі, а також відсутність сурогатних ключів. Єдиним спірним питанням залишається «проблема вибуху» [8] – відомо, що при додаванні нових вимірів куб зростає експоненційно в об'ємі. Але, з іншого боку, реальних (відмінних від нуля) даних стане не більше, ніж у інших моделях, розглянутих вище. Отже, потенційне кількісне зростання буде досить обмеженим. Більш відчутним може виявитись зниження швидкодії під час операцій drill-down (розгортання даних) вздовж природних ієрархій, які були штучно розірвані під час побудови даної моделі. Але, по-перше, немає гарантії, що всі недостаючі проміжні суми (які прискорюють швидкість операцій drill-down) будуть обчислені при інших підходах, і, по-друге, ці недоліки можуть виявитись цілком компенсованими суттєвим зростанням бази обчислень (і розміру кубу відповідно) у двох попередніх підходах (за рахунок вимушеного дублювання елементів на вимірах). Це питання потребує додаткового чисельного вивчення.

Порівняння підходів. Адекватність та ефективність

Питання адекватності та ефективності були розглянуті у трьох попередніх розділах. У таблиці наведено порівняльний аналіз трьох розглянутих підходів до моделювання динамічних вимірів у багатовимірних сховищах з точки зору переваг і недоліків.

Таблиця. Порівняльна характеристика методів подання динамічних вимірів у багатовимірних сховищах

Метод	Переваги	Недоліки
1. Відношення багато-до-багатьох для побудови ієрархії	1) відносно просте моделювання	1) не скрізь реалізовано (не всі виробники OLAP/BI-систем підтримують); 2) складність розуміння моделі [5] (взаємозв'язки і наслідки); 3) збільшення ймовірності помилки при її побудові [5]; 4) потенційне уповільнення обробки інформаційних запитів (із-за ускладнення внутрішньої реалізації обчислень); 5) у випадку часової прив'язки підпорядкування відношення багато-до-багатьох потребує додаткової попередньої переробки
2. Виміри, що повільно змінюються (Slowly Changing Dimensions, SCD)	1) визнання як класичного методу; 2) може бути реалізований у будь-якому середовищі	1) складність подання ієрархії; 2) швидке зростання OLAP-кубу у розмірах (за рахунок вимушеного дублювання елементів ієрархії на нижніх щаблях); 3) спадання швидкості опрацювання інформаційних запитів
3. Незалежні виміри для динамічно підпорядкованих елементів	1) простота реалізації; 2) відтворюваність у довільному середовищі; 3) відсутність складних попередніх обробок при побудові, розгортанні та обробці (наповненні) моделі; 4) відсутність сурогатних ключів	1) потенційне зростання OLAP-кубу у розмірах (ефект «експоненційного вибуху» при великій кількості вимірів); 2) потенційне уповільнення обробки інформаційних запитів; 3) зниження швидкодії під час операцій drill-down (розгортання даних) вздовж природних ієрархій, які були штучно розірвані під час побудови даної моделі

З таблиці видно, що усім методам притаманне уповільнення швидкості опрацювання запитів у тій чи іншій мірі, а також зростання об'єму OLAP-кубу (із розглянутих вище причин). Останній метод має порівняно найбільше переваг, а метод побудови вимірів, що повільно змінюються (Slowly Changing Dimensions, SCD), – визнаний класичним для даної задачі. Адекватність всіх методів обґрунтовано у попередніх розділах.

Залишається зауважити, що в кожній конкретній ситуації потрібно зважувати переваги і недоліки підходів при необхідності подання динамічних ієрархій та обирати оптимальну комбінацію методів. Наприклад, якщо користувачі часто аналізують продажі по областях у розрізі регіонів, то оптимально було б створити такі окремі виміри: динамічний «регіон – область» (методом SCD типу 2) та статичний «населений пункт – торгова точка» (скориставшись при цьому третім підходом (незалежних вимірів) – неявно).

Висновки

Розглянуто два відомих підходи до подання динамічних вимірів, а також запропоновано метод побудови незалежних вимірів. Обґрунтовано адекватність та потенційно не нижчу ефективність методу (доведення ефективності у числах винесено за межі даної статті). Показано, що метод має меншу складність розробки моделі (тобто реалізації), дозволяє швидше побудувати модель та зробити при цьому менше помилок (за рахунок прозорості ідеї). Проаналізовано переваги і недоліки даного методу у порівнянні з іншими підходами: модель з відношенням «багато-до-багатьох» між елементами ієрархії вимірів та виміри, що повільно змінюються в часі (Slowly Changing Dimensions).

Викладений метод було апробовано при розробці реальних багатовимірних сховищ для фарма-компаній, що підтвердило його практичну застосовність та переваги.

1. *Mike Blechar, Merv Adrian, Ted Friedman, W. Roy Schulte, Douglas Laney*. Predicts 2012: Information Infrastructure and Big Data. – Gartner Inc. – 2011, 10 p. (http://www.gartner.com/DisplayDocument?id=1861215)
2. *Douglas Laney*. 3D Data Management: Controlling Data Volume, Velocity and Variety // Application Delivery Strategy, META Group, – 6 Feb 2001 (http://blogs.gartner.com/doug-laney/files/2012/01/ad949-3D-Data-Management-Controlling-Data-Volume-Velocity-and-Variety.pdf)
3. *Mark Beyer*. Gartner Says Solving 'Big Data' Challenge Involves More Than Just Managing Volumes of Data. – Stamford, Conn., – June 27, 2011 (http://www.gartner.com/it/page.jsp?id=1731916)
4. *Codd E.F., Codd S.B., and Salley C.T.* Providing OLAP (On-line Analytical Processing) to User-Analysts: An IT Mandate. – Codd & Date, Inc. – 1993.
5. *Marco Russo, Alberto Ferrari*. The Many-to-Many Revolution: Advanced dimensional modeling with Microsoft SQL Server Analysis Services (V.2 Rev.1). – SQLBI.COM. – 2011. (http://www.sqlbi.com/wp-content/uploads/The_Many-to-Many_Revolution_2.0.pdf)
6. *Richard Tkachuk*. Many-to-Many Dimensions in Analysis Services 2005. (http://msdn.microsoft.com/en-us/library/ms345139.aspx#sql2k5_mmdim_topic2)
7. *Ralph Kimball, Margy Ross*. The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling (Second Edition). – Indianapolis, IN: John Wiley & Sons. – 2002. (ISBN 0-471-20024-72002)
8. *Панченко Т.В*. Композиційні методи специфікації та верифікації програмних систем. Дис. канд.. фіз.-мат. наук / Київський національний університет імені Тараса Шевченка. – Київ, 2006. – 177 с.