

Biological Cybernetics, 43, pp. 59-69, 1982. 5. Haykin S. *Neural Networks: A comprehensive foundation*, Prentice-Hall, New Jersey, 1999. 6. Si J., Lin S., Vuong M.-A. *Dynamic topology representing network*, *Neural Networks Vol 13*, pp. 617-627, 2000. 7. Fritzke B. *Growing cell structure – a self-organising network for unsupervised and supervised learning*, *Neural Networks, Vol. 7 (9)*, pp. 1441-1460, 1995. 8. Blackmore, J., *Visualizing high-dimensional structure with the incremental grid growing neural network. Thesis*, 1995, www.cs.utexas.edu/users/nn/downloads/papers/blackmore.thesis.pdf 9. Bauer H., Villman T. *Growing a hypercubical output space in a self-organizing feature map*. *IEEE Transactions on Neural Networks*, 8, pp. 218-226, 1997. 10. Годич О. Навчання SOM методом нейронної міграції // Вісник НУ “Львівська політехніка”. – 2004. – № 519. – С. 55-72. 11. Hansen E., Walster G. *Global optimization using interval analysis*, Second Edition, Martkel Dekkel, Inc. New York, 2004. 12. Osowski S. *Sieci Neuronowe w ujeciu algorytmicznym*. Wydawnictwo Naukowo-Techniczne, Warszawa, 1996.

УДК 683.1

А.Г. Григорович*, В.Г. Григорович

*Дрогобицький педагогічний ліцей
Дрогобицький державний педагогічний університет імені Івана Франка,
кафедра інформаційних систем і технологій

НЕНОРМАЛІЗОВАНІ ВІДНОШЕННЯ: ІСТОРІЯ, КОНЦЕПЦІЇ ТА ТЕНДЕНЦІЇ РОЗВИТКУ. ЧАСТИНА 1. БАЗОВІ ПОНЯТТЯ, КОНЦЕПТУАЛЬНЕ МОДЕЛЮВАННЯ, МОДЕЛЬ ВЕРСО

© Григорович А.Г., Григорович В.Г., 2005

Наведено першу частину дослідження, в якому проаналізовано історію, основні концепції та тенденції розвитку ненормалізованих відношень. Розглянуто модель Версо та реляційні операції над ненормалізованими відношеннями. Запропоновано розширення моделі “Сутність–Зв’язок” для випадку складних та вкладених сутностей.

The article is the first part of the research in which the authors analyze history, main conceptions and trends of development of nested relations. There are examined Verso model and relational operations upon nested relations. The authors also propose the extension of Entity-Relationship model for cases of nested entities.

Вступ. Загальна постановка проблеми

Проблема адекватного подання даних в інформаційних системах не перестає бути актуальною: з одного боку – для існуючих задач будують нові моделі даних, які адекватніше подають предметну галузь; з іншого боку – виникають нові задачі, які потребують опрацювання даних нової природи. Композитні класи сутностей – ті, які містять сутності інших класів, – зустрічаються надзвичайно часто. Зокрема, в класичній задачі складського обліку сутності *Накладна* та *Замовлення* містять сутності *Відпущений_Товар* та *Замовлений_Товар* відповідно. У межах класичної моделі “Сутність–Зв’язок” вкладені сутності подають за допомогою слабких класів, відношення яких містять зовнішні ключі для зв’язку із відношеннями відповідних сильних класів. Під час переходу від ER-діаграми до реляційної схеми в класичній реляційній моделі втрачається відмінність між відношенням, побудованим на основі сильного класу, та відношенням, побудованим на базі слабого класу сутностей. Отже, класична реляційна модель даних не є адекватною в надзвичайно широкому спектрі застосувань. Ненормалізовані відношення, ідея яких виникла в 1977 році [1], успішно вирішують проблему подання вкладених сутностей.

Для підтримки операцій над ненормалізованими відношеннями вже реалізовано відповідні розширення функцій систем управління базами даних (СУБД). Багато існуючих СУБД (зокрема, такі промислові СУБД, як Oracle) містять інструментальні засоби для опрацювання ненормалізованих відношень.

Зв'язок проблеми із важливими науковими та практичними завданнями

Дослідження в галузі адекватного подання даних загалом та ненормалізованих відношень зокрема надзвичайно актуальні. Особливо багато обіцяє застосування ненормалізованих відношень для дослідження моделей просторових даних, – це дасть змогу на рівні моделі даних реалізувати масштабування: залежно від масштабу композитний об'єкт може розглядатися як “простий” або як такий, що має певну просторову структуру.

Постановка задачі

Стаття є першою частиною огляду досягнень та аналізу основних напрямків досліджень в галузі ненормалізованих відношень.

Аналіз досліджень і публікацій в галузі ненормалізованих відношень

Аналізуючи роботи, присвячені ненормалізованим відношенням, можна виділити такі напрями та тематику публікацій:

- Поняття ненормалізованих відношень;
- Модель Версо;
- Реляційні операції над ненормалізованими відношеннями;
- Нормальні форми для вкладених відношень;
- Реструктурування ненормалізованих відношень;
- Моделі зберігання даних для ненормалізованих відношень;
- Семантика ключів для ненормалізованих відношень;
- Реалізація дій з ненормалізованими відношеннями в промислових СУБД;
- Застосування ненормалізованих відношень.

З огляду на обсяг статті розглянемо тільки публікації за темами:

- Поняття ненормалізованих відношень;
- Модель Версо.

У параграфах 1 та 3 використано приклади із відповідних літературних джерел, на які наведено посилання в тексті.

1. Поняття ненормалізованих відношень

Історія

У 1977 р. А.Т. Makinouchi зазначив [1], що умови першої нормальної форми (1NF), накладені на реляційні бази даних, не підходять для різноманітних програм опрацювання баз даних, таких як системи отримання інформації (*Information Retrieval Systems*) або системи автоматизованого проектування та виробництва (*Computer-Aided Design and Manufacturing, CAD/CAM*). Такі програми потребують оперування структурованими сутностями, тоді як перша нормальна форма допускає лише атомарні (неподільні) значення для атрибутів.

Ненормалізовані реляційні бази даних (Nested Relational Databases) – це спроба задовольнити ці нові вимоги: умова першої нормальної форми відкидається, і для атрибутів допускаються значення, які являють собою набори простих значень.

Новий поштовх для розвитку ненормалізованих відношення отримали в зв'язку із широким застосуванням *об'єктно-реляційної моделі даних (Object-Relational Data Model)* і зокрема – із введенням відповідних можливостей до промислових СУБД (наприклад, Oracle) та до діючих стандартів – SQL3 (SQL-99).

Базові поняття

У роботі [2] подано стислий огляд основних понять ненормалізованих відношень як окремих випадок об'єктно-реляційних моделей даних. Об'єктно-реляційні моделі даних (ОРМД) розширюють реляційну модель шляхом введення об'єктно-орієнтованих можливостей та забезпечення конструювання нестандартних типів даних; допускають складні типи даних для атрибутів в кортежах, зокрема неатомарні значення, – такі, як вкладені відношення. ОРМД зберігають реляційну основу, зокрема декларативний доступ до даних та розширюють можливості моделі даних; вони сумісні з існуючими реляційними мовами.

Ненормалізовані відношення (Nested Relations) допускають неатомарні домени, наприклад: набір цілих чисел, набір кортежів. Ненормалізовані відношення дають змогу на інтуїтивному рівні моделювати задачі з складними типами даних.

Інтуїтивне означення вкладених відношень (строге означення наведено нижче): скрізь, де ми допускаємо атомарні (скалярні) значення, допускається також відношення, – тобто допускаються відношення в складі відношень. При цьому зберігається математична основа реляційної моделі, хоч і порушується вимога першої нормальної форми.

Приклад вкладеного відношення.

Як приклад розглянемо систему отримання документації.

Кожний документ містить:

- заголовок,
- список авторів,
- дату створення (отримання),
- набір ключових слів.

Отже, ненормалізоване відношення *Документи* матиме вигляд, як на рис. 1.1.

Документи

Заголовок	Список авторів	Дата		Ключові слова
		день	місяць рік	
План продажів	{ Сміт, Джонс }	1	квітня 1979	{ прибуток, стратегія }
Звіт	{ Джонс, Фрік }	17	червня 1985	{ прибуток, персонал }

Рис. 1.1. Приклад ненормалізованого відношення

У цьому прикладі атрибути *Список_авторів* та *Ключові_слова* – неатомарні.

Перша нормальна форма (1NF) для ненормалізованого відношення.

Якщо дотримуватися вимог першої нормальної форми, то 1NF-версія відношення *Документи* набуде вигляду, зображеного на рис. 1.2.

1NF-Документи

Заголовок	Автор	День	Місяць	Рік	Ключове слово
План продаж	Сміт	1	квітень	1979	прибуток
План продаж	Джонс	1	квітень	1979	прибуток
План продаж	Сміт	1	квітень	1979	стратегія
План продаж	Джонс	1	квітень	1979	стратегія
Звіт	Джонс	17	червень	1985	прибуток
Звіт	Фрік	17	червень	1985	прибуток
Звіт	Джонс	17	червень	1985	персонал
Звіт	Фрік	17	червень	1985	персонал

Рис. 1.2. 1NF-версія ненормалізованого відношення

Таке відношення характеризується надлишковістю даних, яку можна усунути шляхом декомпозиції.

Декомпозиція відношення *INF-Документи*.

Проводячи декомпозицію, забезпечують її неадитивність, тобто декомпозиція повинна відбуватися без втрат інформації; для цього потрібно зберегти існуючі залежності між атрибутами. Відношення *INF-Документи* характеризується такими багатозначними та функціональними залежностями

Заголовок → Автор, Заголовок → Ключове слово, Заголовок → День, Місяць, Рік.

Тому декомпозицію будемо проводити проєкціями на атрибути {Заголовок, Автор}, {Заголовок, Ключове_слово}, {Заголовок, День, Місяць, Рік} і отримаємо три відношення в четвертій нормальній формі (4-NF) (рис. 1.3).

Проблеми, до яких приводить декомпозиція.

Насамперед декомпозиція вимагає від користувачів вводити операцію з'єднання до своїх запитів. Можна побудувати *погляд (View)*, визначений як з'єднання отриманих після декомпозиції відношень *Автори документів*, *Ключові слова документів* та *Дати отримання документів*. Цей погляд, очевидно, задовольняє вимоги першої нормальної форми, збігається з відношенням *INF-Документи* та усуває необхідність виконувати операцію з'єднання; проте приводить до втрати відповідності один-до-одного між кортежами та документами.

Автори документів

Заголовок	Автор
План продаж	Сміт
План продаж	Джонс
Звіт	Джонс
Звіт	Фрік

Ключові слова документів

Заголовок	Ключове слово
План продаж	прибуток
План продаж	стратегія
Звіт	прибуток
Звіт	персонал

Дати отримання документів

Заголовок	День	Місяць	Рік
План продаж	1	квітень	1979
Звіт	17	червень	1985

Рис. 1.3. 4NF-версія ненормалізованого відношення

Отже, подання даних за допомогою вкладених відношень природніше, краще підходить до нашого прикладу.

Термінологія

Ненормалізовані відношення (Nested Relations), тобто відношення, до складу яких входять інші відношення, також називають *вкладеними відношеннями, складними об'єктами (Complex Objects)*, *відношеннями не першої нормальної форми (Non-First-Normal-Form Relations)* і позначають $\neg INF$, $NINF$, $NFNF$ або NF^2 .

Особливо важливим для програм опрацювання баз даних є підклас ненормалізованих відношень, який називають відношеннями в *частково нормальній формі (Partitioned Normal Form, PNF)*. Відношення перебуває в частково нормальній формі тоді, коли його атомарні атрибути є суперключем відношення і коли будь-який неатомарний компонент кортежу цього відношення також перебуває в частково нормальній формі. Відношення в частково нормальній формі досліджено в [3–7].

2. Розширення моделі “Сутність–Зв’язок”

Ненормалізовані відношення можна трактувати як розширення реляційної моделі з метою більш адекватного представлення предметної області, а саме – складних об'єктів. У такому випадку розширяють класичну модель “Сутність–Зв’язок” [8–10] і вводять до ER-діаграм засоби для подання складних (компонентних) сутностей.

Розглянемо класичну задачу автоматизації складського обліку.

Підсистема відпуску товарів оперує сутностями *Клієнт*, *Накладна*, *Відпущений Товар*, *Наявний Товар*, зв'язки між якими зображено за допомогою ER-діаграми, наведеної на рис. 2.1.

Клієнт (позначається одинарною лінією) *може* бути зв'язаний з багатьма накладними, кожна накладна (позначається подвійною лінією) *повинна* бути зв'язана з одним конкретним клієнтом. Кожна накладна *повинна* містити кілька відпущених товарів, кожний відпущений товар *повинен* бути вказаний у відповідній накладній. Кожний відпущений товар *повинен* бути одним із наявних товарів, наявний товар *може* бути серед відпущених товарів. Оскільки немає можливості відобразити, що в накладній вказані відпущені товари, для відображення того факту, що сутність *Відпущений Товар* не може існувати поза зв'язком із сутністю *Накладна*, *Відпущений Товар* позначають як *слабкий клас* сутностей (подвійна рамка). Строго кажучи, сутність *Накладна* не може існувати поза зв'язком із сутністю *Клієнт*, тому *Накладна* також є слабким класом.

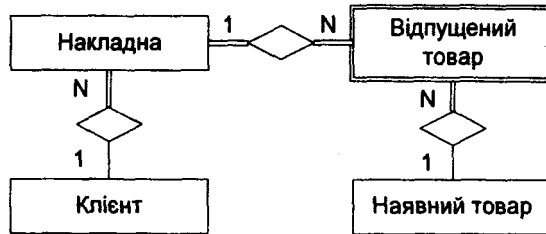
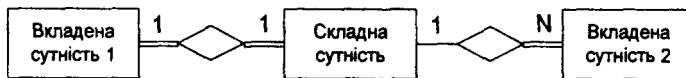


Рис. 2.1. ER-діаграма для підсистеми відпуску товару задачі складського обліку

Можна запропонувати таке розширення для ER-діаграм, яке дає змогу відображати зв'язки "містить–вкладений" (рис. 2.2).



а) класичні позначення на ER-діаграмах



б) розширення позначень для складних і вкладених сутностей

Рис. 2.2. Елементи ER-діаграм для зображення складних та вкладених сутностей

На рис. 2.2, б зображено, що допускається кілька рівнів вкладеності складних сутностей. Якщо складна сутність обов'язково *повинна* містити вкладену сутність (як Вкладену сутність 1 на рис. 2.2), то це будемо позначати подвійною рамкою для блока вкладеної сутності – оскільки обов'язковий тип участі позначають подвійною лінією – рис. 2.2, а. Якщо ж складна сутність *може* і не містити вкладену сутність (як Вкладену сутність 2 на рис. 2.2), то це будемо позначати одинарною рамкою для блока вкладеної сутності – адже необов'язковий тип участі позначають одинарною лінією – рис. 2.2, а. Кардинальність зв'язку вкладеної сутності із складною сутністю, яка безпосередньо її охоплює, позначатимемо показником, записаним у верхньому лівому куті рамки для блока вкладеної сутності. Так, на рис. 2.2 показано, що Складна сутність *повинна* містити *не більше одного* екземпляру (тобто, зв'язана рівно з одним і лише одним екземпляром) Вкладеної сутності 1 і *може* містити *багато* екземплярів (тобто, зв'язана з нулем та більшою кількістю екземплярів) Вкладеної сутності 2.

З врахуванням запропонованого розширення, наведена на рис. 2.1 ER-діаграма моделі даних для підсистеми відпуску товару задачі складського обліку набуде вигляду, вказаного на рис. 2.3.

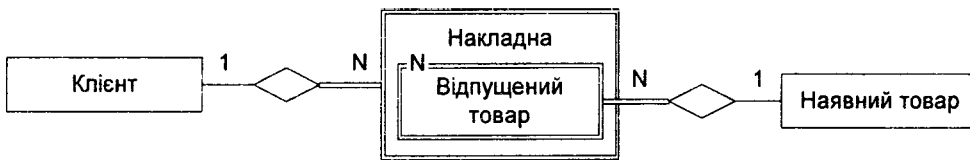


Рис. 2.3. ER-діаграма для підсистеми відпуску товару задачі складського обліку, розширена складною та вкладеною сутностями

Подвійні рамки для блоків складної сутності *Накладна* та вкладеної сутності *Відпущений товар* мають різне значення. Для “зовнішньої” складної сутності подвійна рамка означає, що *Накладна* – це слабкий клас сутностей, екземпляри якого не можуть існувати поза зв’язком із відповідними екземплярами сильного класу (у нашому прикладі – *Клієнт*; тобто не може бути жодної накладної на відпуск товару, в якій не зазначено конкретного клієнта). Для “внутрішньої” вкладеної сутності подвійна рамка означає, що кожний екземпляр сутності *Накладна* повинен містити принаймні один екземпляр сутності *Відпущений товар*.

3. Модель Версо: реляційні операції над ненормалізованими відношеннями

Модель Версо

Ненормалізовані відношення не лише забезпечують зручніше подання даних, а ще й приводять до ефективнішого маніпулювання даними.

Згідно з моделлю Версо [11] (з лат. Verso – трактую, тлумачу) дані організовані у відношення непершої-нормальної-форми. Рекурсивне визначення схеми даних Версо приводить до ієрархічної організації даних і специфікації з’єднання, яке мається на увазі. Як сказано в [11], операції, запропоновані для екземплярів Версо, використовують семантичний зв’язок атрибутів, – деякі запити, які в класичній реляційній моделі потребують з’єднання, можна подати простою вибіркою в моделі Версо.

Модель Версо – одна із перших спроб формального подання ненормалізованих відношень, – це формалізація деяких концепцій, використаних в системі бази даних Версо (*Verso database Machine*) [12]. Структура даних в системі бази даних Версо являє собою екземпляри Версо, які зберігаються послідовно та повністю впорядковано. Такий спосіб зберігання приводить до швидкого доступу та опрацювання даних [3].

Розглянемо приклад (наведений в [13]): У *навчальному закладі* зберігається інформація про набір ДИСЦИПЛІН; ПІДРУЧНИКИ, необхідні для кожної дисципліни; СТУДЕНТІВ, які записалися на вивчення кожної дисципліни та про їхні ОЦІНКИ. Рис. 3.1 подає інформацію в такому *навчальному закладі*. На інтуїтивному рівні *навчальний заклад* можна розглядати як відношення над трьома атрибутами: ДИСЦИПЛІНА, A_1 та A_2 . Значення атрибута ДИСЦИПЛІНА – атомарні, тоді як значення атрибутів A_1 та A_2 являють собою відношення Версо. Зауважимо, що жодного ПІДРУЧНИКА не вимагається для ДИСЦИПЛІНИ “фізика” – отже, значення NULL можуть міститися у відношеннях Версо. Також зауважимо, що наведена схема неявно передбачає існування зв’язку між атрибутами СТУДЕНТ та ПІДРУЧНИК через атрибут ДИСЦИПЛІНА. Іншими словами, завжди мається на увазі “з’єднання” ДИСЦИПЛІНА–СТУДЕНТ та ДИСЦИПЛІНА–ПІДРУЧНИК.

ДИСЦИПЛІНА (ПІДРУЧНИК) * (СТУДЕНТ (ОЦІНКА)) * *

математика	<table border="1"><tr><td>b</td></tr><tr><td>g</td></tr></table>	b	g	toto	<table border="1"><tr><td>4</td></tr><tr><td>8</td></tr></table>	4	8
b							
g							
4							
8							
		zaza	<table border="1"><tr><td></td></tr></table>				
фізика	<table border="1"><tr><td></td></tr></table>		lulu	<table border="1"><tr><td>9</td></tr></table>	9		
9							
		toto	<table border="1"><tr><td>6</td></tr><tr><td>9</td></tr></table>	6	9		
6							
9							

а) екземпляр *f*



б) ієрархічне дерево, яке представляє *f*

Рис. 3.1. Відношення Версо як екземпляри формату

Щоб визначити структуру відношення Версо, використовують допоміжну концепцію формату [13]. Неформально *формат* (*format*) f – це регулярний вираз виду $X(f_1)^* \dots (f_n)^*$, де X – скінченний набір атрибутів, f_1, \dots, f_n – формати (можливо, порожні); причому жодний атрибут не входить двічі до f .

Означення: Нехай f – це формат. Набір всіх *екземплярів формату* f (set of all instances over f), позначається $Inst(f)$ і визначається рекурсивно наступним чином:

1. Якщо $f \equiv X$, то $Inst(f) = REL(X)$. Тут $REL(X)$ – набір відношень над скінченним набором атрибутів X (set of relations over X).

2. Якщо $f \equiv X(f_1)^* \dots (f_n)^*$, то l входить до $Inst(f)$ тоді, коли одночасно виконуються дві умови:

a. l – скінченна підмножина декартового добутку $TUP(X) \times Inst(f_1) \times \dots \times Inst(f_n)$, де $TUP(X)$ – набір кортежів на множині атрибутів X (set of tuples over X);

b. Якщо $\langle u_1 \dots l_n \rangle$ та $\langle u_1' \dots l_n' \rangle$ належать до $Inst(f)$ для деяких значень $u, l_1, \dots, l_n, l_1', \dots, l_n'$, то $l_i = l_i'$ для всіх $i = 1..n$.

Наведене означення стверджує, що екземпляр l набуває атомарних значень атрибутів з набору X та не атомарних, а „екземплярних” значень „атрибутів” f_1, \dots, f_n . Означення також примушує X бути ключем. Екземпляр $ДИСЦИПЛІНА(ПІДРУЧНИК)^*(СТУДЕНТ(ОЦІНКА))^*$ зображено на рис. 3.1.

Алгебра Версо

У роботі [4] наведено повний набір операцій над відношеннями Версо у формі алгебри. Бінарні операції алгебри Версо визначені не тільки для відношень Версо, які мають однакову структуру, а також і для відношень Версо, які мають сумісні структури. Поняття сумісності форматів визначається за допомогою поняття підформату. Формат g є *підформатом* (*subformat*) формату f , якщо ієрархічне дерево, яке подає g , є піддеревом ієрархічного дерева, що подає f і має той самий корінь. Іншими словами, якщо g – це підформат f , і якщо J – це екземпляр формату g , то екземпляр формату f , який будемо називати *розширенням J на f* і позначати J^f , містить ті самі дані, що і J , і може бути отриманий шляхом “заповнення” його кожного рівня порожніми екземплярами. На рис. 3.2 зображені екземпляр J формату $ДИСЦИПЛІНА(СТУДЕНТ)^*$ та його розширення на формат $ДИСЦИПЛІНА(ПІДРУЧНИК)^*(СТУДЕНТ(ОЦІНКА))^*$.

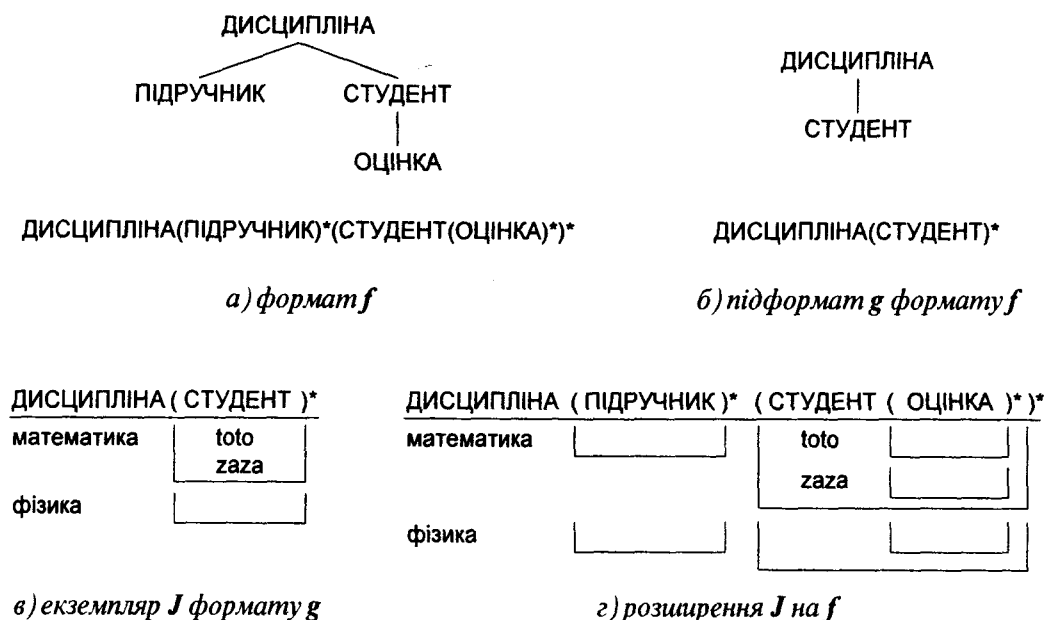


Рис. 3.2. Підформати та розширення формату

Два формати f та g є *сумісними*, якщо вони обидва є підформатами деякого формату h . Цей формат h являє собою загальну структуру, яку можуть подавати екземпляри форматів f та g .

Бінарні операції Версо – це *об'єднання* (*union*), яке додає інформаційний вміст двох відношень Версо; *перетин* (*intersection*), виділяє інформацію, спільну для двох відношень Версо;

різниця (difference) вилучає із інформації, яка міститься у відношенні Версо, інформацію, яка міститься в іншому відношенні Версо; *з'єднання (join)* комбінує інформаційний вміст двох відношень Версо, які мають сумісні структури (та знаходить перетин інформаційних вмістів двох відношень Версо однакової структури).

У роботі [11] бінарні операції визначаються за допомогою відношення введення на відношеннях Версо.

Означення: Нехай f – це формат, а I, J – два екземпляри формату f . Тоді I введено до J , позначається $I \leq J$, якщо справедлива умова:

Якщо $f \equiv X(f_1)^* \dots (f_n)^*$, $i f_1, \dots, f_n$ – не порожні, тоді $\forall \langle uI_1 \dots I_n \rangle \in I, \exists \langle uJ_1 \dots J_n \rangle \in J \mid I_i \leq J_i$ для всіх $i = 1..n$

Об'єднання та різниця Версо формально визначаються за допомогою відношення введення для відношень Версо. Перетин Версо не розглядається, оскільки його означення міститься в означенні з'єднання Версо.

Означення: Нехай f та g – два сумісні формати, а h – такий формат, що f та g – це підформати h . Нехай I, J – два екземпляри форматів f та g відповідно.

Тоді об'єднання I та J згідно з h , позначається $I \oplus_h J$ – це найменший екземпляр формату h , який містить I^h та J^h .

Різниця I та J згідно з h позначається $I \ominus_h J$ – це найменший екземпляр формату h , введений до I^h (тут [13] містить друкарську помилку) такий, що його об'єднання з J згідно з h дорівнює $I \oplus_h J$.

Визначення операції з'єднання Версо потребує формального визначення операції проєкції, тому спочатку наводимо конструктивне визначення операції з'єднання Версо. У [11] бінарні операції визначають як алгебраїчним, так і конструктивним способом.

Означення: Нехай f та g – два сумісні формати, а h – такий формат, що f та g – це підформати h . Нехай I, J – два екземпляри форматів f та g відповідно.

Тоді з'єднання I та J згідно з h позначають $I \otimes_h J$ – визначається рекурсивно так:

1. Якщо $h \equiv X$, то (оскільки $f \equiv g \equiv X$): $I \otimes_h J = I \cup J$.
2. Якщо $h \equiv X(h_1)^* \dots (h_n)^*$, $f \equiv X(f_1)^* \dots (f_n)^*$, $g \equiv X(g_1)^* \dots (g_n)^*$, де для кожного $i = 1..n$: f_i (чи g_i , відповідно) – є підформатами h_i , можливо, порожніми. Тоді:

$$I \otimes_h J = \{ \langle uK_1 \dots K_n \rangle \mid \langle uI_1 \dots I_n \rangle \in I^h, \langle uJ_1 \dots J_n \rangle \in J^h,$$

$$K_k = I_k \otimes_{h_k} J_k, \text{ якщо } f_k \text{ та } g_k \text{ – не порожні,}$$

$$K_k = I_k, \quad \text{якщо } f_k \text{ – не порожнє, а } g_k \text{ – порожнє,}$$

$$K_k = J_k, \quad \text{якщо } f_k \text{ – порожнє, а } g_k \text{ – не порожнє } \}$$

На рис. 3.3. зображено приклади бінарних операцій над екземпляром I формату ДИСЦИПЛІНА(СТУДЕНТ)* та екземпляром J формату ДИСЦИПЛІНА(ПІДРУЧНИК)*; загальний формат, який використовується, щоб відобразити результати операцій над I та J – це $f \equiv$ ДИСЦИПЛІНА(СТУДЕНТ)*(ПІДРУЧНИК)*.

Унарними операціями моделі Версо є *розширення (extension)* (визначене вище), *проєкція (projection)*, *вибірка (selection)*, *обмеження (restriction)*, *перейменування (renaming)* та *реструктурування (restructuring)*. Остання із цих операцій – реструктурування – дає змогу переорганізувати дані і у деяких випадках [4] саме реструктурування потребує особливої уваги.

Для того, щоб ввести поняття проєкції Версо, необхідна проста концепція формату, спроектваного на множину атрибутів. Якщо f – це формат, визначений на множині атрибутів Z (будемо говорити, що f – це формат на Z), P – підмножина Z , тоді проєкція f на P , – позначається $f|_P$, – отримується шляхом вилучення із f всіх атрибутів, які не входять до P , разом з дужками та зірочками, які будуть зв'язані з порожніми рядками.

Приклад. Нехай $f \equiv$ ДИСЦИПЛІНА(ПІДРУЧНИК)*(СТУДЕНТ(ОЦІНКА))* – це формат. Розглянемо три набори атрибутів: $X = \{ \text{ДИСЦИПЛІНА, СТУДЕНТ} \}$, $Y = \{ \text{СТУДЕНТ, ОЦІНКА} \}$, $Z = \{ \text{ПІДРУЧНИК, ОЦІНКА} \}$. Тоді

1. $f|_X = \text{ДИСЦИПЛІНА(СТУДЕНТ)}^*$ – це формат, який є підформатом f .
2. $f|_Y = \text{СТУДЕНТ(ОЦІНКА)}^*$ – це формат, але не підформат f .
3. $f|_Z = (\text{ПІДРУЧНИК})^*(\text{ОЦІНКА})^*$ – це не формат.

Отже, проєкція формату f на набір атрибутів P не завжди буде форматом.

Проєкція екземпляра I формату f на набір атрибутів P визначається лише тоді, коли $f|_P$ – це формат. Отриманий після проєкції екземпляр буде екземпляром формату $f|_P$.

ДИСЦИПЛІНА (СТУДЕНТ)*

математика	toto zaza
гімнастика	mimi
фізика	

а) екземпляр I

ДИСЦИПЛІНА (СТУДЕНТ)*(ПІДРУЧНИК)*

математика	toto zaza	b1 b2
гімнастика	mimi	
музика		b3
фізика		b4

в) екземпляр $I \oplus J$

ДИСЦИПЛІНА (ПІДРУЧНИК)*

математика	b1 b2
музика	b3
фізика	b4

б) екземпляр J

ДИСЦИПЛІНА (СТУДЕНТ)*(ПІДРУЧНИК)*

математика	toto zaza	
гімнастика	mimi	

г) екземпляр $I \ominus J$

ДИСЦИПЛІНА (СТУДЕНТ)*(ПІДРУЧНИК)*

математика	toto zaza	b1 b2
фізика		b4

д) екземпляр $I \otimes J$

Рис. 3.3. Бінарні операції Версо

Означення. Нехай f – формат на Z , P – не порожня підмножина Z , така що $f|_P$ – це формат. Нехай I – це екземпляр формату f . Тоді *проєкція I на P* , – позначається $I|_P$ (або $I|_P$), якщо зрозуміло, що формат f мається на увазі), – це екземпляр формату $f|_P$, визначений рекурсивно так:

1. Якщо $f \equiv X$, то $I|_P = \{ v \mid v \in \text{TUP}(P) \text{ та } \exists u \in I \mid \forall A \in P: u(A) = v(A) \}$.
2. Якщо $f \equiv X(f_1)^* \dots (f_n)^*$, f_1, \dots, f_n – формати на Y_1, \dots, Y_n , відповідно, то

$$I|_P = \bigoplus_{\langle u_1, \dots, u_n \rangle \in I} \{ \langle v \mid J_1 \dots J_n \rangle \mid v = u[X \cup P] \text{ та } J_i = I_i[Y_i \cup P], \text{ де } J_i \text{ – визначене на } f_i|_{Y_i \cap P}, Y_i \cup P \neq \emptyset \}$$

Це означення узагальнює означення реляційної проєкції. На рис. 3.4 зображено екземпляр I формату $\text{ДИСЦИПЛІНА(ПІДРУЧНИК)}^*(\text{СТУДЕНТ(ОЦІНКА)}^*)^*$ та його проєкцію на атрибути { СТУДЕНТ, ОЦІНКА }.

Другою важливою унарною операцією в моделі Версо є вибірка. Ця операція складніша, ніж реляційна вибірка, тому що структура екземплярів Версо – багатша. Для ясності, операція вибірки вводиться за два кроки. Перший крок – визначається варіант вибірки як просте розширення реляційної вибірки. Другий крок – це означення узагальнюється для того, щоб використати семантичний зв'язок між атрибутами, тобто з'єднання, яке неявно мається на увазі у визначенні формату.

Спочатку розширюється поняття умови для атрибутів. У реляційній моделі умова для деякого атрибуту – це алгебраїчний вираз, в який завжди входять константи (тобто, значення із домену цього атрибуту). Нехай $f \equiv X(f_1)^* \dots (f_n)^*$ – це формат. Оскільки елементи “домену” f_i – екземпляри формату,

то природно під час побудови умови для f_i розглядати екземпляри формату f_i як константи, зв'язані з f_i . Проте такий спосіб запису умови на f_i – громіздкий та важкий для розуміння.

ДИСЦИПЛІНА (ПІДРУЧНИК)*(СТУДЕНТ (ОЦІНКА))*		СТУДЕНТ (ОЦІНКА)*	
математика	b1	toto	6
	b2		9
		zaza	8
фізика		toto	10

а) екземпляр I формату f

б) проекція I на {СТУДЕНТ, ОЦІНКА}

Рис. 3.4. Проекція Версо

Означення. Нехай $f \equiv X(f_1)*...*(f_n)*$. Тоді вираз (або оператор) S простої вибірки Версо на f у формі $S \equiv X:C (e_1(S_1)...e_n(S_n))$, де

1. C – умова для набору атрибутів X.
2. S_i – вираз простої вибірки Версо на f_i , для $i = 1..n$.
3. $e_i \in \{ \exists, \neg\exists, ? \}$ для $i = 1..n$. (\exists – означає “існує”, $\neg\exists$ – “не існує”, $?$ – “все одно”)

Приклад: Нехай $f \equiv$ ДИСЦИПЛІНА(СТУДЕНТ(ОЦІНКА))* – це формат. Розглянемо такі два запити:

1. Q_1 : “Отримати список СТУДЕНТІВ, які вивчають математику та отримали ОЦІНКУ більшу, ніж 10; а також ОЦІНОК більших, ніж 10, отриманих цими СТУДЕНТАМИ.”
2. Q_2 : “Отримати ДИСЦИПЛІНИ, які вивчаються принаймні одним СТУДЕНТОМ, який не отримав жодної ОЦІНКИ, та список імен СТУДЕНТІВ, які не мають жодної ОЦІНКИ”.

Запит Q_1 формулюють таким виразом простої вибірки Версо на f (рис. 3.5):

$$S_1 \equiv \text{ДИСЦИПЛІНА : ДИСЦИПЛІНА} = \text{„математика”} \\ (? (\text{СТУДЕНТ} : (\exists (\text{ОЦІНКА} : \text{ОЦІНКА} \geq 10))))$$

Запит Q_2 формулюють таким виразом простої вибірки Версо на f (рис. 3.5):

$$S_2 \equiv \text{ДИСЦИПЛІНА : } (\exists (\text{СТУДЕНТ} : (\neg\exists (\text{ОЦІНКА}))))$$

Вираз S простої вибірки Версо на f визначає таку операцію на $\text{Inst}(f)$:

Означення: Нехай $S \equiv X:C (e_1(S_1)...e_n(S_n))$ – вираз простої вибірки Версо на форматі $f \equiv X(f_1)*...*(f_n)*$. Нехай I – екземпляр на f. Нехай $e \in \{ \exists, \neg\exists, ? \}$, тоді I задовольняє e, позначається I e, якщо $e \equiv ?$ або $e \equiv \exists$ та $I \neq \emptyset$, або $e \equiv \neg\exists$ та $I = \emptyset$.

Тоді результат виразу S, застосованого до I, позначається S(I), визначають так:

$$S(I) = \{ \langle uS_1(I_1)...S_n(I_n) \rangle \mid \langle uI_1...I_n \rangle \in I, \text{ такі, що } u \text{ C та } S_i(I_i) \text{ e}_i \text{ для } i = 1..n \}$$

Це означення ілюструється рис. 3.5, де показано результати виразів простої вибірки Версо S_1 та S_2 із попереднього прикладу.

ДИСЦИПЛІНА (СТУДЕНТ (ОЦІНКА))*		ДИСЦИПЛІНА (СТУДЕНТ (ОЦІНКА))*	
математика	zaza		
	lulu	10	10
	toto	15	15
музика			
фізика	rick	4	
	zoe		
	toto	8	

а) екземпляр I формату f

б) результат Q_1 , застосованого до I

ДИСЦИПЛІНА (СТУДЕНТ (ОЦІНКА))*	
математика	zaza
фізика	zoe

в) результат Q_2 , застосованого до I

Рис. 3.5. Проста вибірка Версо

На основі вищевказаного можна побудувати розширення попереднього означення, яке збільшує потужність операції [13].

Розглянемо наступний запит Q_3 за форматом ДИСЦИПЛІНА(СТУДЕНТ(ОЦІНКА)*)* : “Отримати список ДИСЦИПЛІН, СТУДЕНТІВ та ОЦІНОК, таких, що СТУДЕНТ “toto” отримав ОЦІНКУ “15” з цієї ДИСЦИПЛІНИ та СТУДЕНТ (не обов’язково “toto”) отримав оцінку “10” з такої дисципліни. Цей запит ускладнений наявністю кількох ролей для одного і того самого атрибуту, а саме СТУДЕНТ. Такий запит, як правило, потребує кількох з’єднань у відповідній реляційній моделі. Цей запит означає фактично дві вибірки за значенням ОЦІНКИ: $S_1 \equiv$ ОЦІНКА : ОЦІНКА = “15” та $S_2 \equiv$ ОЦІНКА : ОЦІНКА = “10”. Нам потрібні також дві вибірки за значенням СТУДЕНТ(ОЦІНКА)*:

$$S_1' \equiv \text{СТУДЕНТ} : \text{СТУДЕНТ} = \text{“toto”} (\exists (S_1)),$$

$$S_2' \equiv \text{СТУДЕНТ} : (\exists (S_2)).$$

Перша із них є фільтром для СТУДЕНТА “toto”, якщо він отримав ОЦІНКУ “15”, а друга є фільтром для кожного СТУДЕНТА, який отримав ОЦІНКУ “10”. Можна повністю записати всю вибірку так:

$$S \equiv \text{ДИСЦИПЛІНА} : (? (S') \mid \{ \exists (S_1'), \exists (S_2') \}), \text{ де } S' \text{ ідентичне } \text{СТУДЕНТ(ОЦІНКА)*}.$$

Вираз S не є виразом простої вибірки Версо, визначеним раніше: коли ми виконуємо таку вибірку на екземплярі l формату ДИСЦИПЛІНА(СТУДЕНТ(ОЦІНКА)*)* , для кожного елемента $\langle ul_1 \rangle$ із l , ми виконуємо S_1' та S_2' “паралельно” для l_1 і записуємо $\langle ul_1 \rangle$ (тобто $\langle uS'(l_1) \rangle$) якщо $S_1'(l_1) \neq \emptyset$ та $S_2'(l_1) \neq \emptyset$. Тут S_1' та S_2' використовують виключно як умови, які підтверджують або заперечують результат застосування вибірки ДИСЦИПЛІНА : (? (S')) для $\langle ul_1 \rangle$.

Тепер можна визначити вираз вибірки Версо:

Означення: Нехай $f \equiv X(f_1)^* \dots (f_n)^*$ – формат. Тоді вираз вибірки Версо на f визначають рекурсивно:

1. Вираз простої вибірки Версо на f є виразом вибірки Версо на f .

2. Для $i=1..n$, нехай \hat{S}_i – набір виразів у формі $e'(S')$, де $e' \in \{ \exists, -\exists \}$ та S' – вираз вибірки Версо на f_i . Тоді:

$X:C(e_1(S_1) \mid \hat{S}_1, \dots, e_n(S_n) \mid \hat{S}_n)$ – вираз вибірки Версо на f , де C – умова для X , S_i – вибірка Версо на f_i та $e_i \in \{ \exists, -\exists, ? \}$.

Результат застосування операції вибірки Версо на f до екземпляру формату l визначають так:

Означення: Нехай $X:C(e_1(S_1) \mid \hat{S}_1, \dots, e_n(S_n) \mid \hat{S}_n)$ – вираз вибірки Версо за форматом

$f \equiv X(f_1)^* \dots (f_n)^*$. Нехай l – екземпляр f . Тоді результатом застосування S до l буде екземпляр f , який позначається $S(l)$ і визначений так:

$$S(l) = \{ \langle uS_1(l_1) \dots S_n(l_n) \rangle \mid \langle ul_1 \dots l_n \rangle \in l, \text{ так що } u \in C, \text{ для } i=1..n \ S_i(l_i) \ e_i \text{ та } S'(l_i) \ e' \text{ для всіх } e'(S') \in \hat{S}_i \}$$

На рис. 3.6. зображено результат застосування вибірки Версо до екземпляру l , зображеного на рис. 3.5.

ДИСЦИПЛІНА (СТУДЕНТ (ОЦІНКА) *)*

математика	zaza	
	lulu	10 9
	toto	15

результат Q_3 , застосований до l

Рис. 3.6. Вибірка Версо

Доведено [13], що операцію вибірки Версо можна подати за допомогою простих вибірок Версо, які містять лише умови у формі “ \exists ” (такі вибірки називаються \exists -V-вибірки) та за допомогою проєкцій, об’єднань, різниць та з’єднань Версо.

Зв’язок моделі Версо та реляційної моделі

Формат із моделі Версо зв’язаний із схемою відповідної реляційної бази даних (за допомогою *каркаса формату – format skeleton* [13]), це приводить до неієрархічного опису структури даних, визначеної форматом:

Означення: Нехай f – формат. Тоді *каркас формату* f позначається $\text{Skel}(f)$ – це схема реляційної бази даних, яку рекурсивно визначають так:

1. Якщо $f \equiv X$, то $\text{Skel}(f) = \{X\}$.
2. Якщо $f \equiv X(f_1)^* \dots (f_n)^*$, то $\text{Skel}(f) = \{X\} \cap \{XY \mid Y \in \text{Skel}(f_i) \text{ для деякого } i \in [1..n]\}$.

Наприклад, схема бази даних, зв’язана з форматом

ДИСЦИПЛІНА(ПІДРУЧНИК)*(СТУДЕНТ(ОЦІНКА))* буде такою: $R = \{\{\text{ДИСЦИПЛІНА}\}, \{\text{ДИСЦИПЛІНА, ПІДРУЧНИК}\}, \{\text{ДИСЦИПЛІНА, СТУДЕНТ}\}, \{\text{ДИСЦИПЛІНА, СТУДЕНТ, ОЦІНКА}\}\}$.

Використовуючи поняття каркаса формату, інформаційний вміст екземпляру формату можна “описати” за допомогою екземплярів реляційної бази даних.

Означення: Нехай f – формат, l – екземпляр формату f . Екземпляр каркаса l , позначається $\text{skel}(l)$, – це екземпляр реляційної бази даних із схемою $\text{Skel}(f)$, визначений так:

1. Якщо $f \equiv X$, то $\text{skel}(l)(X) = l$.
2. Якщо $f \equiv X(f_1)^* \dots (f_n)^*$, то $\text{skel}(l)(X) = \{u \mid \langle u_1 \dots u_n \rangle\}$ та $\text{skel}(l)(XY) = \bigcap_{\langle u_1 \dots u_n \rangle \in l} \{u \times \text{skel}(l_i)(Y)\}$, $Y \in \text{Skel}(f_i)$ для деякого $i \in [1..n]$.

Зрозуміло, що не всі схеми баз даних зв’язані з деякими форматами; більше того, не всі екземпляри баз даних, визначені на каркасі формату, зв’язані з екземплярами формату.

Доведено [4] теорему, яка зв’язує екземпляр реляційної бази даних з екземпляром формату:

Теорема: Нехай f – формат, та $R = \text{Skel}(f)$. Нехай r – екземпляр R . Тоді наступні два твердження – еквівалентні:

1. $r = \text{skel}(l)$ для деякого екземпляра l формату f .
2. r задовольняє універсальному припущенню реляційної схеми (*Universal Relation Schema Assumption, URSA*), тобто

$\Pi_X(r(Y)) \subseteq r(X)$ для кожного X, Y із R та $X \subseteq Y$ [14].

Тут r – екземпляр реляційної бази даних деякої реляційної схеми R ; X та Y – множини атрибутів із R ; Π_X – проєкція на атрибути X ; $r(X)$ – значення атрибутів X в кортежах r . URSA означає, що для всіх множин атрибутів X та Y із R таких, що $X \subseteq Y$, кортежі відношення $\Pi_X(r(Y))$ входять до сукупності кортежів відношення $r(X)$.

Після визначення відображення екземплярів формату f на екземпляри бази даних схеми $\text{Skel}(f)$, які задовольняють URSA, можна виразити операції Версо для V-відношень в термінах реляційних операцій для екземплярів реляційної бази даних.

В [4] дається проста інтерпретація таких бінарних операцій Версо, як об’єднання, різниця та з’єднання:

Теорема: Нехай f та g – два сумісні формати, такі що f та g – підформати формату h . Нехай l та J – екземпляри форматів f та g відповідно. Нехай $r = \text{skel}(l^h)$ та $s = \text{skel}(J^h)$. Тоді:

1. $\text{skel}(l \oplus_h J) = r \cap s$.
2. $\text{skel}(l \text{ }_h J)$ – найменший URSA-екземпляр схеми $\text{Skel}(h)$, який містить r - s .

(Нехай r та s – два екземпляри бази даних із схемою R , тоді $r \subseteq s$, якщо $\forall X \in R \ r(X) \subseteq s(X)$)

3. $\text{skel}(I \otimes_h J)$ – найбільший URSA-екземпляр схеми $\text{Skel}(h)$, який містить екземпляр t схеми $\text{Skel}(h)$, визначений так:

- a. $t(X) = r(X) \cup s(X)$, якщо $X \in \text{Skel}(f) \cup \text{Skel}(g)$,
- b. $t(X) = r(X)$, якщо $X \in \text{Skel}(f) - \text{Skel}(g)$,
- c. $t(X) = s(X)$, якщо $X \in \text{Skel}(g) - \text{Skel}(f)$,
- d. $t(X) = \emptyset$ в інших випадках.

Унарні операції проєкції та вибірки Версо характеризуються так:

Теорема [15]: Нехай f – формат, визначений на Z , підмножина $P \subseteq Z$ така, що проєкція fl_P формату f на атрибути P – це формат та g – це fl_P . Нехай l – екземпляр формату f , $r = \text{skel}(l)$ – екземпляр каркаса l схеми $\text{Skel}(f)$. Тоді $\text{skel}(l[g])$ – найменший URSA-екземпляр, визначений на схемі $\text{Skel}(g)$, який містить $\text{Pr}(r)$.

Як зазначалося раніше, операцію вибірки Версо можна подати за допомогою Версо-операцій проєкції, \exists -V-вибірки, об'єднання, різниці та з'єднання. Операцію \exists -V-вибірки трактують так:

Теорема [15]. Нехай f – формат, S – вираз \exists -V-вибірки на f . Нехай l – екземпляр формату f та $r = \text{skel}(l)$. Тоді:

$\forall Z \in \text{Skel}(f)$, $\text{skel}(S(l))(Z) = \Pi_Z(\sigma)$, де σ – екземпляр реляційної схеми, визначений на $\bigcap_{Z \in \text{Skel}(f)} Z$ так:

$\sigma = \bowtie_{Z \in \text{Skel}(f)} [\text{Select}_{\wedge_{Y \subseteq Z} C_Y} r(Z)]$, де C_Y – умова на Y , яка використовується в S .

Тут позначено реляційні операції:

Π_Z – проєкція на атрибути Z ;

\bowtie – з'єднання;

Select_C – вибірка, де C – умова на значення атрибутів.

Ефективне обчислення реляційних запитів за допомогою вибірки та проєкції Версо.

У роботі [4] показано, що алгебра Версо – “повна”. Це означає, що всі запити реляційної алгебри можна подати алгеброю Версо. В роботі [13] досліджено потужність виразів вибірки Версо: великий набір реляційних запитів може бути замінений вибіркою Версо та проєкцією Версо. Для характеристики такого класу реляційних запитів використовують подання реляційних виразів проєкції–вибірки–з'єднання (*projection–selection–join, PSJ-виразів*) за допомогою схеми табло (*schema tableau*), введене в роботах [16–19].

У роботі [13] показано запити, які потребують з'єднання в реляційній моделі, але можуть бути просто подані вибіркою в моделі Версо:

Наприклад, розглянемо формат $f \equiv \text{ДИСЦИПЛІНА}(\text{ПІДРУЧНИК})^*(\text{ДАТА_ЕКЗАМЕНУ})^*$ та запит “Які є ДИСЦИПЛІНИ, такі що СТУДЕНТ „toto” складає з них ЕКЗАМЕН „1 листопада”?”. В реляційній моделі дані типово представляються за допомогою двох відношень ДИСЦИПЛІНА–СТУДЕНТ та ДИСЦИПЛІНА–ДАТА_ЕКЗАМЕНУ, і запит буде потребувати операції з'єднання. Цей запит можна сформулювати такою простою вибіркою Версо:

$S \equiv \text{ДИСЦИПЛІНА} : (\exists (\text{СТУДЕНТ} : \text{СТУДЕНТ} = \text{„toto”}),$
 $\exists (\text{ДАТА_ЕКЗАМЕНУ} : \text{ДАТА_ЕКЗАМЕНУ} = \text{„1 листопада”}))$

Деякі природні запити типу „Отримати список дисциплін, для яких невідома дата проведення екзамену” можна подати простою вибіркою Версо, тоді як в реляційній моделі вони потребують використання операції різниці.

У роботі [13] розглянуто реляційні вирази вибірки, проєкції та з'єднання таких категорій:

1. PSJ-вирази, які можуть бути сформульовані в термінах \exists -V-вибірки;
2. PSJ-вирази, які можуть бути сформульовані в термінах \exists -V-вибірки та проєкції Версо;
3. PSJ-вирази, які можуть бути сформульовані в термінах простої вибірки Версо (без символу $\neg\exists$, – така вибірка називається \exists -?-V-вибіркою) та проєкції Версо;

4. PSJ-вирази, які можуть бути сформульовані в термінах вибірки Версо (без символу $\neg\exists$) та проєкції Версо.

Пропускаючи доведення відповідних тверджень, розглянемо приклад:

Дано формат $f \equiv \text{ДИСЦИПЛИНА}(\text{ПІДРУЧНИК})^*(\text{СТУДЕНТ}(\text{ОЦІНКА})^*)^*$

та схема реляційної бази даних $R = \text{Skel}(f) = \{\{\text{ДИСЦИПЛИНА}\}, \{\text{ДИСЦИПЛИНА}, \text{ПІДРУЧНИК}\}, \{\text{ДИСЦИПЛИНА}, \text{СТУДЕНТ}\}, \{\text{ДИСЦИПЛИНА}, \text{СТУДЕНТ}, \text{ОЦІНКА}\}\}$.

Розглянемо запит: “Отримати ДИСЦИПЛИНИ, які вивчає СТУДЕНТ „toto” і такі, що всі СТУДЕНТИ, які вивчають ці ДИСЦИПЛИНИ, отримали ОЦІНКУ, вищу „10” балів”.

У реляційній моделі такий запит може бути сформульований таким PSJ-виразом на R:

$$E \equiv \Pi_{\text{ДИСЦИПЛИНА}, \text{СТУДЕНТ}} [\text{Select}_{\text{ОЦІНКА} > "10"} [\text{ДИСЦИПЛИНА}, \text{СТУДЕНТ}, \text{ОЦІНКА}] \bowtie (\Pi_{\text{ДИСЦИПЛИНА}} [\text{Select}_{\text{СТУДЕНТ} = "toto"} [\text{ДИСЦИПЛИНА}, \text{СТУДЕНТ}]])]$$

Наведемо вираз вибірки Версо S на f, такий що його проєкція Версо

$[\text{ДИСЦИПЛИНА}, \text{СТУДЕНТ}] \circ S$ еквівалентна E:

$$S \equiv \text{ДИСЦИПЛИНА} : (? (\text{СТУДЕНТ} : (\exists (\text{ОЦІНКА} : \text{ОЦІНКА} > "10"))) \mid \{ \exists (\text{СТУДЕНТ} : \text{СТУДЕНТ} = "toto" (? (\text{ОЦІНКА} :))) \} , ? (\text{ПІДРУЧНИК} :))$$

Отже, за моделлю Версо можна сформулювати широкий клас запитів, використовуючи лише операції проєкції та вибірки Версо, – реляційна модель в цих випадках вимагає ще операції з’єднання; а отже, для таких запитів вирази алгебри Версо простіші, ніж вирази реляційної алгебри.

Висновки

Ця робота є першою із циклу статей, присвячених огляду досягнень та аналізу основних напрямків досліджень в галузі ненормалізованих відношень.

У статті розглянуто основні поняття та переваги ненормалізованих відношень.

Запропоновано розширення моделі “Сутність–Зв’язок” на випадок складних (компонентних) та вкладених сутностей, завдяки чому можна адекватніше подавати предметну галузь, яка містить складні об’єкти.

У рамках моделі Версо описано реляційні операції над ненормалізованими відношеннями та показано більшу ефективність обчислення запитів, сформульованих за допомогою виразів алгебри Версо порівняно із запитам на основі виразів реляційної алгебри. Ефективність збільшується за рахунок того, що для формулювання широкого класу запитів достатньо лише операцій проєкції та вибірки Версо, – операція з’єднання, яка необхідна для виразів реляційної алгебри, завжди мається на увазі і неявно присутня в самій структурі ненормалізованих відношень.

1. A. Makinouchi. A consideration on normal form of not-necessarily-normalized relation in the relational data model. In Proc. 3rd International Conference on Very Large Databases, pages 447-453, Tokyo, Oct. 1977. 2. Silberschatz, Korth and Sudarshan. Database System Concepts. Ch. 9. Object-Relational Databases. Microsoft PowerPoint - ch9.ppt. www.cs.brown.edu/courses/cs127/lectures/L13.or-db.pdf 3. F. Bancilhon, P. Richard and M. Scholl. On line processing of compacted relations. In Proc. 8th International Conference on Very Large Databases, pages 263-269, Mexico City, Sep. 1982. 4. S. Abiteboul and N. Bidoit. Non first normal form relations: An Algebra allowing data restructuring. Journal of Computer and System Sciences, 33: 361-393, 1986. 5. M. A. Roth, H. F. Korth and A. Silberschatz. Extended algebra and calculus for nested relational databases. ACM Transactions on Database Systems, 13 (4): 389-417, Dec. 1988. 6. G. Hulin. A Relational Algebra for Nested Relations in Partitioned Normal Form. Manuscript M318, Philips Research Laboratory Brussels, Oct. 1989. 7. G. Hulin. On Restructuring Nested Relations in Partitioned Normal Form. In Proc. 16th VLDB Conference, p. 626-637. Brisbane, Australia 1990. www.vldb.org/conf/1990/P626.PDF 8. Chen P. P. The Entity-Relationship Model: Toward a Unified View of Data. ACM Trans. on Database Systems. 1976, 1(1), p. 9-36. 9. Batini Carlo, Ceri S., Navathe S.B., and Batani Carol. Conceptual Database Design: An Entity/Relationship

Approach. Addison-Wesley, reading, MA, 1991. 10. Thalheim B. *Fundamentals of Entity-Relationship Modeling*, Springer-Verlag, Berlin, 2000. 11. Abiteboul, S. and Bidoit, N. *Non First Normal Form Relations to Represent Hierarchically Organized Data*. In Proc. 3rd ACM SIGACT-SIGMOD, Waterloo, 1984, p. 191-198. 12. Bancilhon, F. and al. *Verso: A Relational Back-End Data Machine*. In Proc. Inter., Workshop on Database Machines, san Diego, 1982. 13. Bidoit, N. *Efficient Evaluation of Relational Queries Using "Nested Relations"*. Rapport de recherche de l'INRIA-Rocquencourt RR-0480 -1986. www.inria.fr/rrrt/rr-0480.html - 3k 14. Ullman, J. *Principles of Databases Systems*. Computer Science Press. 15. Bidoit, N. *Un Modele de Donnees Relationnel Non Normalise: Algebre et Interpretation*. PhD thesis, Universite de Paris-Sud, Centre d'Orsay, 1984. 16. Maier, D., Mendelzon, A., Sagiv, Y. *A Consideration on Normal Form of Not-Necessarily Normalized Relation in the Relation Model*. In Proc. Inter. Conf. on VLDB, Tokyo, 1977, p. 447-453. 17. Aho, A., Sagiv, Y., Ullman J. *Efficient Optimization of a Class of Relational Expressions*. ACM Trans. On Database Systems, 1979, 4(4): 435-454. 18. Aho, A., Sagiv, Y., Ullman J. *Equivalences Among Relational Expressions*. Siam Journal Comp., 1979, 8(2): 218-246. 19. Klug, A. *On Inequality Tableaux*. Technical Report 403, Computer Science Department, University of Wisconsin-Madison, November, 1980.

УДК 004.02

М.В. Давидов, Ю.В. Нікольський
Національний університет "Львівська політехніка",
кфедра інформаційних систем та мереж

КЛАСИФІКАЦІЯ ЕЛЕМЕНТІВ ВІДЕОЗОБРАЖЕНЬ РЕАЛЬНОГО ЧАСУ ЗА ДОПОМОГОЮ НЕЙРОМЕРЕЖІ

© Давидов М. В., Нікольський Ю.В., 2005

Описаний прототип системи керування виступом за допомогою набору жестів. Задачу вирішено побудовою класифікатора елементів зображення за допомогою багатошарової нейронної мережі. Модифіковано алгоритм навчання зі зворотним поширенням похибки. Запропоновано новий метод визначення еталону та навчальних прикладів.

The prototype of conference control system is described. The problem is solved by applying the neural network as classifier of image elements. New modification of the backpropagation algorithm for training the neural network was built. The new way to define the patterns and standards is used.

Постановка проблеми

Пропонуємо результати досліджень, здійснених з метою побудови прототипу системи інтерактивного керування виступом. Доповідач може застосовувати широкий спектр засобів супроводження виступу, до яких належать мультимедійний проектор, кодоскоп, проектор слайдів, системи керування освітленням приміщення, шторами, мікрофонами, засобами запису та відображення інформації: диктофоном, магнітофоном, відеокамерою, цифровим фотоапаратом тощо. Цими пристроями може керувати оператор з центрального пульта. У такому випадку для виступу виділяють спеціально обладнане приміщення з вказаним комплектом обладнання, а оператору необхідно мати сценарій виступу. У запропонованій системі, спрощену схему якої показано на рис.1, відсутня необхідність у такому обладнаному приміщенні. Систему можна скласти з наявних пристроїв у вільній конфігурації з мінімальними вимогами до вартості та сумісності їх роботи, зокрема, не потрібний центральний пульт та спеціально підготовлений оператор. Всі пристрої під'єднано до центрального сервера, який керує їхньою роботою. Доповідача постійно знімає відеокамера, комп'ютер опрацьовує відеосигнал, виділяє у ньому жести доповідача та передає керуючий сигнал на відповідний пристрій. Складовими цієї системи є доповідач, екран, відеока-