

ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ТА КОМП'ЮТЕРНА ТЕХНІКА

УДК 681.51

В. Б. Мокін, д. т. н., проф.;

А. Р. Ящолт, асп.

НОВИЙ МЕТОД АВТОМАТИЗОВАНОГО ФОРМУВАННЯ ЗВІТІВ ІНФОРМАЦІЙНИХ СИСТЕМ НА ОСНОВІ СЕКВЕНЦІЙНОГО ПІДХОДУ

Розроблено теоретичні та алгоритмічні основи нового методу автоматизованого формування складних звітів реляційних баз даних інформаційних систем, який є аналогом секвенційного методу проектування технічних систем. Метод, оснований на представленні вхідних та вихідних даних у вигляді секвенційної моделі, дозволяє синтезувати програмний код для формування звіту для заданого набору вхідних та вихідних даних та не вимагає знань мови SQL. Також формалізовано основні операції з даними, які виконуються під час побудови звітів інформаційних систем.

1. Постановка задачі

Одним з найбільш рутинних процесів під час проектування баз даних інформаційних систем є формування звітів. Тому важливими та актуальними є різні методи автоматизації цих процесів. Побудова кожного типу звіту потребує, як правило, індивідуального підходу. Для автоматизації звіту треба добре знати і предметну область, і володіти навичками програмування.

Як правило, сучасні інформаційні системи використовують реляційні бази даних: MS Access, Oracle, Paradox та інші, в середовищі яких застосовується мова SQL (Structured Query Language).

В реляційній базі даних таблиці є взаємопов'язаними між собою (вступають у відношення) через ключові поля. Полем називається стовпець таблиці, який містить значення (одиночний елемент інформації в певному записі або рядку). Запис — це логічний еквівалент рядку таблиці, тобто є набором значень полів визначеного рядка таблиці. Як правило, використовуються реляційні бази даних, які задовольняють правилам нормалізації [1, 2].

В інформаційному плані вхідними даними кожного звіту є множина X полів x_i ($i = 1, \dots, n$) таблиць, де зберігається первинна інформація або результат її обробки. Вихідними даними є множина Y полів y_j ($j = 1, \dots, m$) нової таблиці, в якій розміщується інформація, що, власне, і складає звіт. Задача ідентифікації моделі звіту полягає у визначенні зв'язків між множинами X та Y .

Як правило, немає проблеми у проектуванні звіту на основі однієї таблиці. Складнощі виникають, якщо звіт є складним і формується на основі багатьох таблиць та містить додаткові умови відбору даних.

Аналіз методів та засобів для автоматизованого формування складних звітів інформаційних систем виявив такі основні складнощі:

- вимагаються глибокі знання структури бази даних;
- необхідним є знання мови SQL;
- відсутня наочність зв'язків між вхідними даними та вихідними полями звіту;
- уповільнений процес проектування.

Таким чином, актуальним є розв'язання такої задачі: розробити метод автоматизованого проектування складних звітів реляційної бази даних інформаційної системи, який дозволив би автоматизовано синтезувати програмний код для формування звіту для заданого набору вхідних та вихідних даних та не вимагав би знань мови SQL і був оснований на наочному представленні вхідних та вихідних даних. Бажаним є автоматизований синтез моделі даних наявної реляційної бази да-

них.

Проаналізуємо відомі підходи до розв'язання поставленої задачі, розробимо основні теоретичні засади методу та формалізуємо основні операції з даними, які мають місце під час побудови звітів інформаційних систем.

2. Ідея методу

Зазвичай поставлена задача розв'язується з використанням спеціального редактора, наприклад Конструктора чи Майстра звітів в MS Access [2—4]. У цьому випадку потрібно обрати таблицю, за якою формуватиметься звіт, вибрати її поля, що містять вхідні дані, задати порядок та рівень групування звітної інформації, задати порядок її сортування, обрати один з багатьох видів макету звіту та стилів його оформлення. Результатом цього є таблиця, що містить результат вибірки даних із вхідних таблиць.

Головним недоліком цього алгоритму при формуванні складного звіту є необхідність у програмуванні мовою SQL. Запропонуємо як цього можна уникнути.

Ідея методу, що пропонується, полягає у використанні спеціалізованого програмного забезпечення, яке автоматизує рутинні операції та операції, що вимагають знання SQL, а для перетворень та оптимізації моделі інформаційного звіту використовується спеціальний апарат. Алгоритм застосування цього програмного забезпечення такий:

1. Програма автоматично будує модель даних (інформаційну модель) заданої реляційної бази даних у певному формалізованому вигляді.
2. Користувач відбирає з цієї моделі лише ту частину, яка містить вхідні дані для побудови звіту.
3. Користувач формує та оптимізує модель звіту, яка пов'язує його вхідні та вихідні дані.
4. Користувач вводить синтезовану модель звіту у програму у спеціальному редакторі.
5. Програма автоматично формує програмний код, який реалізує задану модель звіту.
6. Програма запускає синтезований програмний код та дає можливість переглянути текст звіту, забезпечуючи можливість його збереження у форматі файлів MS Word та MS Excel.

Зрозуміло, що ключовим пунктом алгоритму є третій. У разі визначення формалізованого опису інформаційної моделі звіту технічно нескладно побудувати програму для реалізації пунктів 1, 5 та 6. Розробимо теоретичні засади цього опису.

Пропонуємо синтез та оптимізацію інформаційної моделі звіту здійснювати на основі математичного апарату, який є аналогом секвенційного опису кінцевих автоматів у технічних системах.

3. Розробка основних теоретичних засад методу

Секвенція (з англ. — «sequence») — послідовність, результат, наслідок [5]. В техніці використовується апарат секвенцій для формалізації зв'язків входів та виходів кінцевих автоматів.

Розглянемо спеціальний запис у вигляді [5]:

$$\phi | - \varphi, \quad (1)$$

де ϕ та φ — функції алгебри логіки.

Запис у вигляді (1) називається секвенційним або секвенцією.

Зміст секвенційного запису полягає в тому, що його права частина (функція φ) приймає одиничне значення у випадку, якщо одиничне значення приймає ліва частина (функція ϕ). Мовою обчислення висловлень зміст секвенції (1) інтерпретується таким чином: «Висловлення φ істинне, якщо істинне висловлення ϕ ».

Проведемо аналогію секвенційного опису для звітів інформаційних систем.

Порівняння вихідних положень та понять секвенційного апарату для технічних систем з аналогічними для інформаційних систем дозволяє визначити особливості, зведені у таблицю.

Вихідні положення та поняття секвенційного апарату для технічних та їх аналоги для інформаційних систем

№ з/п	Для технічних систем	Для реляційних БД інформаційних систем
1	Є чітке співвідношення між вхідними та вихідними змінними	Є чітке співвідношення між вхідними та вихідними змінними
2	Множина вихідних змінних автомата відповідає єдиній множині вхідних змінних	Множина вихідних змінних таблиці бази даних відповідає багатьом множинам (таблицям) вхідних змінних (полів)
3	Вхідні змінні набувають значень 0 або 1	Вхідні змінні набувають різних значень відповідно до їх типів даних
4	Існує три основні операції з бінарними величинами: «Ні» «Або» «Та»	Аналог операції «Ні» наявний лише у разі порівняння змінної з фіксованим значенням: якщо, наприклад « $x = x_1$ », тоді «Ні x » — це буде « $x \neq x_1$ » Аналогом операції «Або» є вибір того або іншого поля з різних таблиць в одному запиті Аналогом операції «Та» є об'єднання різних полів однієї таблиці в одному запиті

Важливо зазначити, що в автоматах сигнали приймають бінарні значення, тобто струм у провідниках або є, або немає. А в реляційних базах даних — можуть бути довільними, в межах даного типу даних.

Таким чином, секвенційний опис моделей інформаційних систем повинен враховувати таблицю-джерело вхідної змінної T_1 та таблицю T_2 , в якій зберігається вихідна змінна. Виходячи з цього, пропонується такий формалізований опис секвенції для моделей інформаційних систем:

$$x_1 x_2 \left| \begin{array}{l} T_1 \\ T_2 \end{array} \right. y_1, \quad (2)$$

де вхідні змінні x_1 та x_2 , які є полями таблиці T_1 , відповідають (мають логічний зв'язок) вихідній змінній y_1 , яка є полем таблиці T_2 .

Як було зазначено вище, секвенційний апарат застосовується до змінних, що приймають лише два варіанти значень: 1 або 0. За аналогією, пропонується у виразах, подібних до (2), записувати

$$x_1 = x_1^*, \quad (3)$$

що означає те, що можливі лише два варіанти значень: 1 ($x_1 = x_1^*$) або 0 ($x_1 \neq x_1^*$). Такий запис забезпечує певну аналогію пропонованого секвенційного апарату, що, із секвенційним апаратом для технічних систем.

З іншого боку, варто згадати, що таблиця є кінцевою множиною полів, з яких вона складається. Саме ці поля і є змінними у виразі (2). Покажемо, яким чином ідентифікуються вхідні та вихідні змінні, а також зв'язок між ними.

Наприклад, таблиця містить такі поля:

$$T_1 = [A, B, C, I], \quad (4)$$

де A, B, C, I — деякі поля.

Для формування вибірки записів із таблиці T_1 , значення полів яких задовольняють певним умовам, застосовується мова SQL. Наприклад, вибірка значень поля A мовою SQL має вигляд:

$$T_2 = \text{SELECT } A \text{ FROM } T_1 \text{ WHERE } B = B^* \ C = C^* \ I = I^*. \quad (5)$$

Результатом цієї операції буде таблиця T_2 , яка буде містити усі значення поля A записів таблиці T_1 , в яких значення полів B, C, I задовольняють вказаним умовам. За таких умов, поля B, C, I та умови на них можна розглядати як вхідні дані, а поле A — як вихідні. Враховуючи наведені вище позначення, запит (5) можна записати у вигляді:

$$B = B^* \ C = C^* \ I = I^* \left| \begin{array}{l} T_1 \\ T_2 \end{array} \right. A. \quad (6)$$

Секвенція у вигляді (6) є типовою для опису взаємозв'язків між даними реляційних баз даних. Аналіз особливостей реляційних баз даних зумовив необхідність формалізації деяких усклад-

нень опису (6):

1. Декілька вихідних змінних, наприклад:

$$B = B^* \quad C = C^* \quad \left| \begin{array}{l} T_1 \\ T_2 \end{array} \right. \quad A \quad I. \quad (7)$$

2. Вхідні змінні є полями декількох таблиць, наприклад T_1 і T_2 , тоді секвенція для змінної вихідної таблиці T_3 може мати вигляд:

$$B = B^* \quad C = C^* \quad I = I^* \quad \left| \begin{array}{l} T_1, T_2, T_3 \\ T_2 \end{array} \right. \quad A. \quad (8)$$

3. Вхідні змінні однієї секвенції є вихідними змінними іншої, наприклад:
— має місце система секвенцій:

$$I = I^* \quad \left| \begin{array}{l} T_2 \\ T_3 \end{array} \right. \quad V, \quad (9)$$

$$B = B^* \quad C = C^* \quad \left| \begin{array}{l} T_1 \\ T_2 \end{array} \right. \quad A, \quad (10)$$

де I — це ключове поле таблиці T_2 , а V — це поле таблиці T_3 , яка містить власне звіт

$$B = B^* \quad C = C^* \quad \left| \begin{array}{l} T_1 \\ T_2 \end{array} \right. \quad I = I^* \quad \left| \begin{array}{l} T_2 \\ T_3 \end{array} \right. \quad V. \quad (11)$$

4. Іноді варто підкреслити, що результат запиту є множиною n конкретних значень вихідної змінної

$$B = B^* \quad A = A^* \quad \left| \begin{array}{l} T_1 \\ T_2 \end{array} \right. \quad I = [I_1^* \dots I_n^*]. \quad (12)$$

Такий запис є доцільним під час побудови системи взаємопов'язаних секвенцій, щоб через ці конкретні значення показати зв'язок між різними секвенціями.

Наведемо приклад секвенційного опису моделей інформаційних систем обробки даних спостережень якості вод. Розглянемо випадок, коли є дві таблиці, що містять інформацію про водний об'єкт, місце розташування створу, басейн річки та територіальну належність водного об'єкту та створу до певної адміністративної одиниці.

Запишемо вхідні дані у вигляді [6].

$$W_s = [S_s, D_s, F_s, V_s], \quad (13)$$

де W_s — відомості про розташування створу моніторингу якості поверхневих вод; S_s — номер створу; D_s — дата відбору; F_s — назва показника; V_s — значення (результат вимірювання);

$$W_u = [S_u, D_u, F_u, V_u], \quad (14)$$

де W_u — відомості про розташування створу, що контролює скиди стічних вод; S_u — номер створу; D_u — дата відбору; F_u — назва показника; V_u — значення (результат вимірювання);

$$B = [R, S], \quad (15)$$

де B — басейн річки; R — назва річки; S — номер створу, який розташований на цій річці;

$$A = [A_R, S], \quad (16)$$

де A — територіальна належність водного об'єкта (райони, область); A_R — назва річки, яка відноситься до цієї території; S — номер створу, який розташований на цій річці в цьому районі.

Розглянемо випадок, коли з таблиці W_s вибираються та аналізуються записи з даними про якість поверхневих вод V_s у конкретну дату D_s^* , за конкретним показником якості F_s^* , у конкретному створі S_s^* . Результат вибірки — значення поля V_{si}^* ($i = 1, \dots, n$) — заноситься у таблицю Z_s . Для цього випадку секвенція матиме такий вигляд:

$$D_s = D_s^* \quad F_s = F_s^* \quad S_s = S_s^* \quad \left| \frac{W_s}{Z_s} \right. \quad V_s = [V_{s1}^* \dots V_{sn}^*]. \quad (17)$$

За наявності інформації про те, що аналізуються створи, які розташовані на конкретній річці R^* або території A_R^* , мають місце такі секвенції:

$$R = R^* \quad \left| \frac{B}{Z_s} \right. \quad S = [S_1^* \dots S_m^*]; \quad (18)$$

$$A_R = A_R^* \quad \left| \frac{A}{Z_s} \right. \quad S = [S_1^* \dots S_k^*], \quad (19)$$

де S — створи, що аналізуються; Z_s — підсумкова таблиця по створах, що аналізувалися. Аналогічно щодо даних про розташування створу контролю скидів стічних вод та його параметрів маємо секвенцію:

$$D_u = D_u^* \quad F_u = F_u^* \quad S_u = S_u^* \quad \left| \frac{W_u}{Z_u} \right. \quad V_u = [V_{u1}^* \dots V_{un}^*]. \quad (20)$$

У випадку, коли дати відбору проб однакові, має місце спрощення:

$$D_u^* = D_s^* = D^*. \quad (21)$$

Введемо систему нових умовних позначень, яка дозволить перейти до класичного секвенційного опису від умов типу (7):

$$\begin{aligned} x_1 &= (D_s = D_s^*); \quad \bar{x}_1 = (D_s \neq D_s^*); & x_2 &= (F_s = F_s^*); \quad \bar{x}_2 = (F_s \neq F_s^*); \\ x_3 &= (S_s = S_s^*); \quad \bar{x}_3 = (S_s \neq S_s^*); & x_4 &= (D_u = D_u^*); \quad \bar{x}_4 = (D_u \neq D_u^*); \\ x_5 &= (F_u = F_u^*); \quad \bar{x}_5 = (F_u \neq F_u^*); & x_6 &= (S_u = S_u^*); \quad \bar{x}_6 = (S_u \neq S_u^*); \\ x_7 &= (R = R^*); \quad \bar{x}_7 = (R \neq R^*); & x_8 &= (A_R = A_R^*); \quad \bar{x}_8 = (A_R \neq A_R^*); \\ y_1 &= (V_s = V_s^*); \quad \bar{y}_1 = (V_s \neq V_s^*); & y_2 &= (V_u = V_u^*); \quad \bar{y}_2 = (V_u \neq V_u^*). \end{aligned} \quad (22)$$

Згідно з новою системою позначень секвенції (17)—(20) набудуть вигляду:

$$\begin{aligned} x_1 \ x_2 \ x_3 \ \left| \frac{W_s}{Z_1} \right. \ y_1; & \quad x_4 \ x_5 \ x_6 \ \left| \frac{W_u}{Z_2} \right. \ y_2; \\ x_7 \ \left| \frac{B}{Z_3} \right. \ s_1; & \quad x_8 \ \left| \frac{A}{Z_4} \right. \ s_2. \end{aligned} \quad (23)$$

4. Основні операції над секвенціями моделей інформаційних систем

Однією з найефективніших особливостей секвенційного апарату для технічних систем є комплекс операцій над секвенціями, який дозволяє мінімізувати кількість виразів у системі секвенцій, перейшовши до її скороченого вигляду.

Розробимо аналогічний апарат для секвенційного опису моделей інформаційних систем.

Відповідно до наведених вище позначень мають місце такі операції (див. табл.):

1. Операція «Та» полягає у формуванні секвенції з об'єднанням полів однієї таблиці:

$$x_1 \ x_2 \ x_3 \ \left| \frac{T_1}{T_2} \right. \ y_1, \quad (24)$$

тобто вихідна змінна y_1 з таблиці T_2 є результатом операції над x_1 «Та» x_2 «Та» x_3 з таблиці T_1 .

2. Операція «Або» полягає у формуванні секвенції з об'єднанням полів різних таблиць:

$$x_1 \ x_2 \ \vee \ x_3 \ x_4 \ \left| \frac{T_1, T_2}{T_3} \right. \ y_1, \quad (25)$$

тобто вихідна змінна y_1 з таблиці T_3 є результатом операції над x_1 «Та» x_2 з таблиці T_1 і x_3 «Та» x_4 з

таблиці T_2 .

За аналогією з класичним секвенційним апаратом для технічних систем наведемо алгоритм мінімізації секвенцій та формування скороченого секвенційного опису системи [5]:

1. Проаналізувати ліві частини секвенції та виявити однакові частини та частини, що повторюються.

2. Проаналізувати набір вхідних та вихідних таблиць та виявити однакові таблиці із вхідними або вихідними даними, відповідно.

3. Згрупувати вхідні дані відповідно до вхідних та вихідних таблиць зі скороченням змінних, які повторюються або ідентичні, шляхом застосування операції «Або».

Головна перевага запропонованого секвенційного апарату та алгоритму його мінімізації є можливість швидкого автоматизованого синтезу програмного коду для реалізації такої системи. Доведемо це.

Програмна реалізація моделей інформаційних систем

Побудова спрощених секвенцій дозволить написати вже оптимізовані коди мовою SQL.

Для прикладу розглянемо таку ситуацію: потрібно знайти значення якості води V_s у конкретну дату D_s^* , за конкретним показником якості F_s^* , у конкретному створі S_s^* та у заданій річці R^* . Аналізуючи вирази (13) та (18), бачимо, що вся інформація знаходиться у двох таблицях: W_s та B . Враховуючи секвенції (13) та (18), маємо таку секвенцію:

$$D_s = D_s^* \quad F_s = F_s^* \quad S_s = S_s^* \quad \vee \quad R = R^* \quad \left| \frac{W_s, B}{Z_s} \right. \quad V_s. \quad (26)$$

Використовуючи систему позначень (25), отримаємо секвенцію:

$$x_1 \quad x_2 \quad x_3 \quad \vee \quad x_7 \quad \left| \frac{T_1, T_2}{T_3} \right. \quad y_1. \quad (27)$$

Враховуючи вищезазначені особливості, секвенція мовою SQL матиме такий вигляд [7]:

$$T_3 = \text{ADDFIELD } (y_1). \quad (28)$$

$$T_3 = \text{SELECT } y_1 \text{ FROM } T_1, T_2 \text{ WHERE } T_1.x_1, T_1.x_2, (T_1.x_3 \text{ OR } T_2.x_7). \quad (29)$$

Наведемо приклади простіших випадків секвенцій:

$$x_7 \quad \left| \frac{B}{T_1} \right. \quad S_1, \quad (30)$$

$$x_8 \quad \left| \frac{A}{T_2} \right. \quad S_2. \quad (31)$$

Для (30) матимемо

$$T_1 = \text{SELECT } S_1 \text{ FROM } B \text{ WHERE } x_7 = x_7^*. \quad (32)$$

а для (31)

$$T_2 = \text{SELECT } S_2 \text{ FROM } A \text{ WHERE } x_8 = x_8^*. \quad (33)$$

Для (23) можна записати:

$$Z = \text{ADDFIELD } (y_1), \quad (34)$$

$$Z = \text{SELECT } y_1 \text{ FROM } W_s \text{ WHERE } D = D_s, F = F_s, S = S_s. \quad (35)$$

Таким чином синтезується та оптимізується інформаційна модель звіту. Конкретизуємо алгоритм застосування методу. На першому кроці програма синтезує модель бази даних (30) або (13)—(16). На четвертому кроці користувач вводить в програму модель у вигляді, подібному до (20), (23), (26) тощо. Інші кроки алгоритму зрозумілі.

Висновки

Розроблено метод автоматизованого проектування складних звітів реляційної бази даних інформаційної системи, який дозволяє автоматизовано синтезувати програмний код для формування звіту для заданого набору вхідних та вихідних даних та не вимагає знань мови SQL. Метод оснований на наочному представленні вхідних та вихідних даних у вигляді секвенційної моделі. Розроблено основні положення секвенційного опису моделей інформаційних систем та формалізовано основні операції для цих моделей. Метод ґрунтується на автоматизованому синтезі моделі даних наявної реляційної бази даних та дозволяє реалізувати або значно прискорити процес створення звітів інформаційних систем для осіб, які не володіють мовою SQL.

СПИСОК ЛІТЕРАТУРИ

1. Горев А. Эффективная работа с СУБД / А. Горев, Р. Ахаян, С. Макашарипов. — СПб.: Вильямс, 1997. — 256 с.
2. Дейт К. Введение системы баз данных, [6-е изд.] пер. с англ. / К. Дейт. — К.: М.; СПб.: Вильямс, 2000. — 848 с.
3. Романюк О. Н., Савчук Т. О. Організація баз даних і знань: [навч. посібн.] / О. Н. Романюк, Т. О. Савчук. — Вінниця: УНІВЕРСУМ–Вінниця, 2003. — 217 с.
4. Тимошок Т. В. Microsoft Access 2003 / Т. В. Тимошок. — СПб.: Вильямс, 2006. — 448 с.
5. Захаров В. Н. Автоматы с распределенной памятью / В. Н. Захаров. — М.: «Энергия», 1975. — 136 с.
6. Комп'ютеризовані регіональні системи державного моніторингу поверхневих вод: моделі, алгоритми, програми. [монографія]; під ред. В. Б. Мокіна / [В. Б. Мокін, А. Р. Яцолт та ін.] — Вінниця: УНІВЕРСУМ–Вінниця, 2005. — 315 с.
7. Дюбуа П. MySQL. [2-е изд.]; пер. с франц. / П. Дюбуа. — М.: Вильямс, 2004. — 1056 с.

Рекомендована кафедрою моделювання та моніторингу складних систем

Надійшла до редакції 11.06.08
Рекомендована до друку 19.06.08

Мокін Віталій Борисович — завідувач кафедри, **Яцолт Андрій Русланович** — аспірант.

Кафедра моделювання та моніторингу складних систем, Вінницький національний технічний університет