

ОЦІНЮВАННЯ ДОЦІЛЬНОСТІ ПРОВЕДЕННЯ РЕФАКТОРИНГУ БАЗИ ДАНИХ ІНФОРМАЦІЙНОЇ СИСТЕМИ, ЯКА ЕКСПЛУАТУЄТЬСЯ

Розглянуто основні особливості такого різновиду ІТ-проектів, як рефакторинг бази даних інформаційної системи, яка експлуатується. Виділено проблему визначення доцільності проведення робіт з рефакторингу бази даних. Запропоновано об'єктивні оцінки складності актуальної бази даних та сукупності запитів на зміни інформаційної системи, яка експлуатується. Розроблено показник обсягу змін, які слід внести під час рефакторингу бази даних, що пропонується.

1. Вступ

Рефакторинг бази даних (БД) - це зміна схеми БД з метою покращення дизайну бази даних і збереження як інформації, так і семантики поведінки. Зокрема, покращення дизайну БД має на меті створення умов для легкого розширення функціоналу ІТ-проекту. Рефакторинг є усталеною концепцією в розробці програмного забезпечення, і принципи, якими керується рефакторинг програмного забезпечення, можна також застосувати до рефакторингу БД [1]. Мета рефакторингу БД - зробити схему БД більш гнучкою, масштабованою та придатною для обслуговування з часом, мінімізуючи ризик появи помилок або втрати даних. Рефакторинг може включати ряд дій, від невеликих змін окремих таблиць або стовпців до більш масштабних змін у загальній архітектурі схеми [2].

Рефакторинг БД може бути складним і трудомістким процесом, особливо у великих або критично важливих системах. Важливо ретельно планувати та виконувати рефакторинг, а також ретельно тестувати отримані зміни, щоб переконатися, що вони не створюють нових проблем і не порушують існуючі функції.

Більшість розповсюджених методів успішного рефакторингу БД включають [2]:

- створення резервної копії або знімка БД перед внесенням будь-яких змін;
- детальне документування поточної схеми та будь-яких запропонованих змін;
- тестування змін у невикористаній середовищі перед застосуванням їх до системи, яка експлуатується;
- зведення до мінімуму обсягу та частоти змін, щоб уникнути порушення нормальної роботи;
- співпраця із зацікавленими сторонами, розробниками та адміністраторами БД, щоб переконатися, що рефакторинг відповідає бізнес-цілям і технічним вимогам.

Ефективність рефакторингу БД зазвичай пропонується визначати як мінімум змін та вибір найкращого шляху реструктуризації БД з метою підтримки розширення функцій системи.

Проведення рефакторингу у подальшому розвитку проекту буде сприяти підвищенню ремонтпридатності БД та спрощенню її структури. Це спростить додавання нових функцій інформаційної системи, виправлення помилок і адаптацію до бізнес-вимог, що змінюються, без внесення помилок або порушення нормальної роботи. Таким чином, інвестиції в проект рефакторингу БД принесуть низку переваг, які покращать результати бізнесу та знизять ризики у подальшому розвитку інформаційної системи.

Але існує дуже багато проблем, пов'язаних з ініціацією, плануванням та управлінням рефакторингом БД як різновидом ІТ-проекту модернізації інформаційної системи, що експлуатується. Причиною виникнення цих проблем є, перш за все, майже повна відсутність формального апарату, який дозволяв би описати процеси рефакторингу та особливості управління таким різновидом ІТ-проектів. Тому виникає необхідність проведення досліджень, присвячених розробці нових та вдосконаленню існуючих моделей і методів управління рефакторингом як різновидом ІТ-проекту та його окремими процесами.

2. Аналіз особливостей сучасних підходів до опису рефакторингу бази даних

Незважаючи на те, що рефакторинг БД як різновид ІТ-проектів виник вже достатньо давно, рівень формальності опису рефакторингу залишається дуже низьким. Основну увагу дослідники зосереджують на створенні прикладних методик проведення рефакторингу БД.

Як приклад такої методики розглянемо запропоновану Скоттом Амблером та його колегами у [2] методику рефакторингу БД. Вона включає в себе набір процедур та практик, які допомагають безпечно та ефективно проводити рефакторинг БД у рамках проекту розробки програмного забезпечення. Ця методика пропонує підхід до рефакторингу, який дозволяє поступово покращувати якість БД та знижувати ризики, пов'язані зі зміною БД [2].

Методика рефакторингу БД представлена С. Амблером як процес, наведений на рис. 1 [3].

Переваги використання методики рефакторингу БД включають [2, 3]:

- поліпшення якості БД;
- зниження ризиків, пов'язаних зі зміною БД;
- прискорення процесу розробки БД;
- збільшення гнучкості системи;
- поліпшення розуміння структури БД.

Як недоліки зазначеної методики слід вказати:

- значну трудомісткість та складність для невеликих проектів або невеликих змін у БД;
- неможливість легкого застосування у випадках, коли програмне забезпечення має високий ступінь залежності від БД;
- можливість виникнення потреби у використанні спеціалізованих інструментів, таких як інструменти для автоматичного тестування БД або інструменти міграції даних, що призводить до різкого збільшення витрат на виконання рефакторингу;
- можливість використання тільки для поліпшення якості БД.

Ще одним підходом до організації і проведення рефакторингу БД є модель управління якістю даних (Data Quality Management Model), розроблена The Data Warehousing Institute. Вона надає докладний план дій для поліпшення якості даних і включає процеси, пов'язані з рефакторингом БД. Ця модель складається з таких елементів, що описують процес управління якістю даних [4]:

- визначення вимог щодо якості даних - визначення критеріїв якості даних та створення механізму для оцінки відповідності даних цим критеріям;
- аналіз якості даних - оцінка якості даних з використанням певних критеріїв якості та визначення причин помилок у даних;
- очищення даних - видалення або коригування некоректних чи неповних даних з метою покращення якості даних;

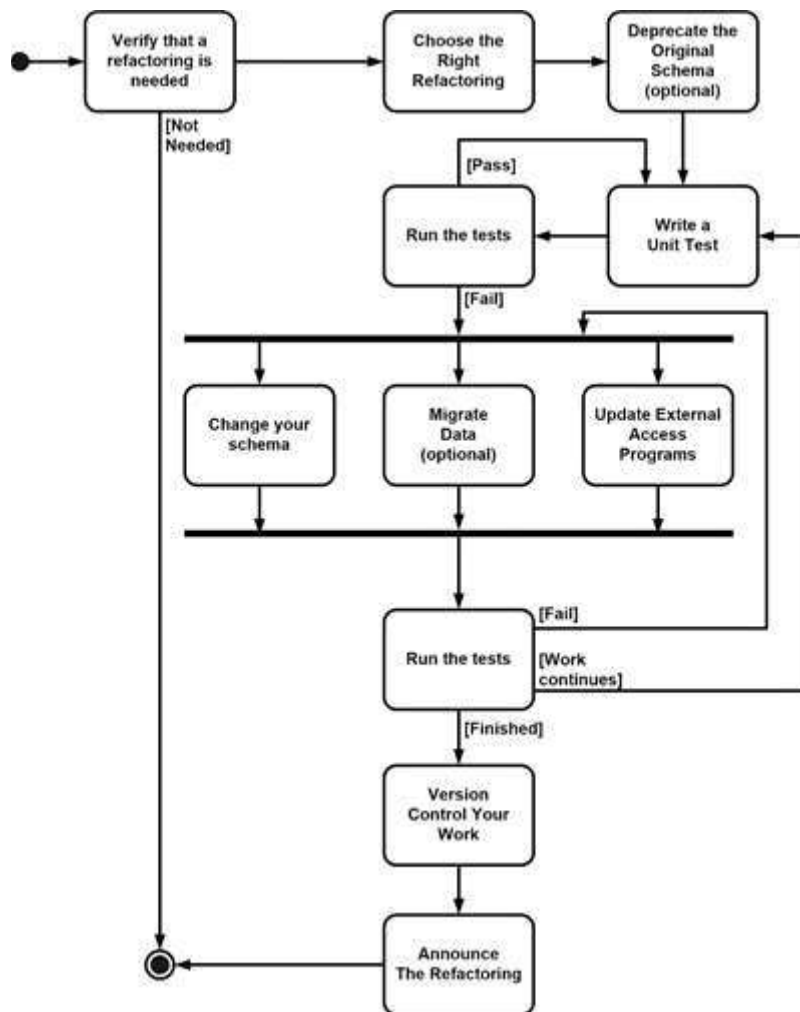


Рис. 1. Схема процесу рефакторингу бази даних

- управління якістю даних - встановлення процедур та механізмів для безперервного контролю якості даних та забезпечення їх відповідності встановленим критеріям якості;
- моніторинг якості даних - оцінка та контроль якості даних для забезпечення їх відповідності правилам та стандартам.

Використання моделі управління якістю даних для проведення рефакторингу БД дозволяє надати команді виконавців відповідного ІТ-проєкту такі можливості:

- можливість поліпшення розуміння структури БД;
- можливість поліпшення якості БД;
- можливість зниження ризиків, пов'язаних зі зміною БД;
- можливість збільшення гнучкості системи для введення нового функціоналу інформаційної системи.

Головним недоліком моделі управління якістю даних є її декларативність та обмеженість концептуальним рівнем опису. Перехід від задекларованих у цій моделі елементів управління якістю БД вимагає великих витрат часу і ресурсів.

В цілому можна стверджувати, що існуючі підходи до опису рефакторингу БД не проявляють достатньо уваги до формальних описів моделей задач, які виникають під час такого рефакторингу, та методів вирішення цих задач. І перш за все це стосується процесу "Verify that a refactoring is needed" (рис. 1). Головною метою цього процесу є об'єктивне визнання необхідності проведення рефакторингу БД до початку його проведення. Виконання цього процесу дозволяє вчасно відмовитися від проведення рефакторингу БД, якщо обмеження відповідного ІТ-проєкту перевищують очікуваний прибуток. Тому в даному дослідженні основну увагу буде приділено рішенням задачі кількісного оцінювання доцільності проведення рефакторингу БД інформаційної системи, яка експлуатується.

3. Мета і задачі дослідження

Метою даного дослідження є розробка показника кількісного оцінювання обсягу змін, які слід внести під час рефакторингу БД, що пропонується, за результатами порівняльного аналізу БД інформаційної системи, яка експлуатується, та запитів на зміну елементів цієї системи. Досягнення цієї мети дозволить своєчасно прийняти рішення про доцільність проведення рефакторингу БД або про необхідність виконання реінжинірингу БД замість запропонованого рефакторингу.

Для досягнення даної мети в статті пропонується вирішити такі задачі:

- розробити кількісні показники складності актуальної БД та сукупності запитів на зміни інформаційної системи, яка експлуатується;
- розробити показник кількісного оцінювання обсягу змін, які слід внести під час рефакторингу БД, що пропонується.

4. Моделі, які використовуються для формального опису бази даних та запитів на зміни

І для опису обсягу змін, що вносяться, і для опису актуальної БД (тобто БД інформаційної системи, яка експлуатується, до проведення рефакторингу) необхідно використовувати однакокий математичний апарат. Як такий математичний апарат пропонується використати теоретико-множинні описи діаграми "сутність - зв'язок" (Entity - Relation Diagram (ERD)), запропоновані у [5]. Даний апарат обраний тому, що ERD є найпоширенішою формою представлення проєктованої чи змінюваної БД, яка практично не залежить від конкретної моделі даних. Крім того, ERD використовується як засіб опису структур даних у класичній моделі вимог до системи, запропонованій у методології SSADM [6]. Ця класична модель вимог до системи може використовуватися і для опису запитів на зміни інформаційної системи, яка експлуатується.

Будь-яку БД можна описати кортежем [5]

$$ERD_{DB} = \langle E, R, D \rangle, \quad (1)$$

де $E = \{E_i\}$ - множина сутностей ERD, $i=1,t$, $R = \{R_j\}$ - множина зв'язків, визначених на сутностях множини E та елементах цих сутностей, $i=1,n$, $D = \{D_l\}$ - множина доменів, визначених в ERD, $l=1,p$.

Кожна сутність з множини E_i може бути також описана кортежем вигляду [5]

$$E_i = \langle n_{E_i}, Tit_{E_i}, B_{E_i} \rangle, \quad (2)$$

де n_{E_i} - ім'я сутності E_i ; Tit_{E_i} - заголовок сутності E_i ; $B_{E_i} = \{e_{E_i}^k\}, k = 1, 2, \dots$ - тіло сутності E_i ; $e_{E_i}^k$ - k -й екземпляр сутності E_i .

Заголовок Tit_{E_i} сутності E_i , в свою чергу, може бути описаний кортежем вигляду [5]

$$Tit_{E_i} = \langle atr_{E_i}^j \rangle \subseteq Atr_E, \quad (3)$$

де Atr_E - множина атрибутів, які використовуються для формування заголовків усієї множини сутностей E ; $atr_{E_i}^j$ - j -й атрибут, який використовується для опису заголовку сутності E_i .

Кожен атрибут $atr_{E_i}^j$ сутності E_i може бути описаний кортежем вигляду [5]

$$atr_{E_i}^j = \langle n_{E_i}^j, D_{E_i}^j \rangle, \quad (4)$$

де $n_{E_i}^j$ - ім'я атрибута $atr_{E_i}^j$; $D_{E_i}^j$ - домен атрибута $atr_{E_i}^j$, причому $D_{E_i}^j \in D$.

Необхідно відзначити, що будь-який атрибут в ERD може приймати тільки атомарні, неподільні (для ERD) значення. Виходячи з цього, кожен екземпляр сутності $e_{E_i}^k$ може бути описаний кортежем вигляду [5]

$$e_{E_i}^k = \langle val_{E_i}^{kj} \rangle, \quad (5)$$

де $val_{E_i}^{kj}$ - значення, яке приймає в екземплярі сутності $e_{E_i}^k$ атрибут $atr_{E_i}^j$, який використовується для опису заголовку сутності E_i . При цьому $val_{E_i}^{kj} \in D_{E_i}^j$.

Зв'язком $R_i \in R$ в ERD прийнято називати іменовану значну асоціацію між двома сутностями чи сутності із собою. У загальному випадку зв'язок $R_i \in R$ між сутностями E_m та E_n може бути описаний кортежем вигляду [5]

$$R_i = \langle n_{R_i}, Atr_{E_m}^{R_i}, Atr_{E_n}^{R_i}, Pow_{E_m}^{R_i}, Pow_{E_n}^{R_i}, S_{E_m}^{R_i}, S_{E_n}^{R_i} \rangle, \quad (6)$$

де n_{R_i} - ім'я зв'язку R_i ; $Atr_{E_m}^{R_i}$ - підмножина атрибутів сутності E_m , що беруть участь в утворенні зв'язку R_i ; $Atr_{E_n}^{R_i}$ - підмножина атрибутів сутності E_n , що беруть участь в утворенні зв'язку R_i ; $Pow_{E_m}^{R_i}$ - потужність зв'язку R_i для сутності E_m (кількість екземплярів $e_{E_m}^k \in B_{E_m}$, що беруть участь в утворенні зв'язку R_i); $Pow_{E_n}^{R_i}$ - потужність зв'язку R_i для сутності E_n (кількість екземплярів $e_{E_n}^k \in B_{E_n}$, що беруть участь в утворенні зв'язку R_i); $S_{E_m}^{R_i}$ - ступінь участі сутності E_m у зв'язку R_i , яка визначає обов'язковість участі

екземплярів $e_{E_m}^k \in B_{E_m}$ у зв'язку R_i ; $S_{E_n}^{R_i}$ - ступінь участі сутності E_n у зв'язку R_i , яка визначає обов'язковість участі екземплярів $e_{E_n}^k \in B_{E_n}$ у зв'язку R_i .

Аналізуючи множину зв'язків $R = \{R_i\}$ за ступенем обов'язковості участі сутностей в утворенні даних зв'язків, можна розглянути цю множину як сукупність підмножин [5]

$$R = R^{00} \cup R^{10} \cup R^{11}, \quad (7)$$

де R^{00} - підмножина зв'язків, необов'язкових з боку сутностей E_m та E_n ; R^{10} - підмножина зв'язків, обов'язкових з боку сутності E_n і необов'язкових з боку сутності E_m ; R^{11} - підмножина зв'язків, обов'язкових з боку сутностей E_m и E_n .

Тоді умови належності зв'язку R_i до однієї з розглянутих вище підмножин, виділених за ступенем обов'язковості участі сутностей в утворенні даного зв'язку, визначаються так [5]

$$R \in R^{00} \text{ за умови } \begin{cases} S_{E_m}^{R_i} = 0; \\ S_{E_n}^{R_i} = 0; \end{cases}; \quad R \in R^{10} \text{ за умови } \begin{cases} S_{E_m}^{R_i} = 1; \\ S_{E_n}^{R_i} = 0; \end{cases} \quad (8)$$

$$R \in R^{11} \text{ за умови } \begin{cases} S_{E_m}^{R_i} = 1; \\ S_{E_n}^{R_i} = 1; \end{cases}, \quad (9)$$

причому

$$S_{E_m}^{R_i} = \begin{cases} 1 \text{ якщо } \exists e_{B_k}^i \in B_{E_k}; \\ 0 \text{ в протилежному випадку.} \end{cases} \quad (10)$$

Аналізуючи найпоширеніші типи зв'язків, що утворюють множину $R = \{R_i\}$, виділені за кількістю сутностей, що беруть участь в утворенні екземплярів, дану множину можна описати так

$$R = R^{om} \cup R^{mm}, \quad (11)$$

де R^{om} - підмножина зв'язків типу "один - до багатьох"; R^{mm} - підмножина зв'язків типу "багато - до багатьох". Зв'язок типу "один - до одного" розглядається як окремий випадок зв'язку типу "один - до багатьох".

Тоді умови належності зв'язку R_i до однієї з розглянутих вище підмножин, виділених за кількістю сутностей, що беруть участь в утворенні екземплярів; визначаються так [5]:

$$R_i \in R^{om} \text{ за умови } \begin{cases} Pow_{E_m}^{R_i} = 1; \\ Pow_{E_n}^{R_i} = k, k = 1, p; \end{cases} \quad (12)$$

$$R_i \in R^{mm} \text{ за умови } \begin{cases} Pow_{E_m}^{R_i} = k, k = 1, p; \\ Pow_{E_n}^{R_i} = l, l = 1, p. \end{cases} \quad (13)$$

Для ERD у нотації IDEF1X немає можливості позначати ступінь участі сутності з боку "багато", і при переході від логічної моделі даних до фізичної підмножина зв'язків R^{mm} відображається в підмножину зв'язків R^{om} .

Для того, щоб отримати детальний опис зв'язків типу "один - до багатьох", введемо поняття первинного ключа PK_{E_i} для сутності E_i . Такий ключ описуватиметься виразом [5]

$$PK_{E_i} = Atr_{E_i}^{PK} \subseteq Atr_{E_i}, \quad (14)$$

де $Atr_{E_i}^{PK} = \langle atr_{E_i}^{jPK} \rangle$ - підмножина атрибутів, що утворюють первинний ключ для сутності E_i . Елементи даної підмножини виділяються таким чином [5]

$$\begin{cases} val_{E_i}^{kPK} = \{val_{E_i}^{kjPK}\} \subseteq \langle val_{E_i}^{kj} \rangle; val_{E_i}^{kjPK} \neq \emptyset; r(Atr_{E_i}^{PK}) \rightarrow \min; \\ val_{E_i}^{kPK} \neq val_{E_i}^{lPK}, \text{ якщо } k \neq l; \\ val_{E_i}^{kPK} = val_{E_i}^{lPK}, \text{ якщо } k = l. \end{cases} \quad (15)$$

де $val_{E_i}^{kPK}$ - значення первинного ключа для екземпляра сутності $e_{E_i}^k$; $val_{E_i}^{kjPK}$ - значення j-го атрибуту, який бере участь в утворенні первинного ключа, що є присутнім в екземплярі сутності $e_{E_i}^k$; $r(Atr_{E_i}^{PK})$ - функція, що визначає потужність підмножини атрибутів $Atr_{E_i}^{PK}$, які утворюють первинний ключ сутності E_i .

За аналогією з визначенням первинного ключа, введемо поняття зовнішнього ключа FK_{E_m} для сутності E_m . Такий ключ можна описати виразом [5]

$$FK_{E_m} = Atr_{E_m}^{FK} \subseteq Atr_{E_m}, \quad (16)$$

де $Atr_{E_m}^{FK} = \langle atr_{E_m}^{jFK} \rangle$ - підмножина атрибутів, що утворюють первинний ключ для сутності E_m . Елементи даної підмножини виділяються таким чином [5]

$$\begin{cases} \exists E_n, \text{ причому для } e_{E_n}^k \in B_{E_n} \exists e_{E_m}^l \in B_{E_m}; \\ val_{E_n}^{kPK} = val_{E_m}^{lFK}; val_{E_n}^{kPK} = \{val_{E_n}^{kjPK}\}; val_{E_m}^{lFK} = \{val_{E_m}^{ljFK}\}; \\ val_{E_m}^{lFK} \in D^j; val_{E_n}^{kPK} \in D^j; D^j \in D, j = 1, t; \end{cases} \quad (17)$$

де $val_{E_n}^{kPK}$ - значення первинного ключа для екземпляра сутності $e_{E_n}^k$; $val_{E_m}^{lFK}$ - значення вторинного ключа для екземпляра сутності $e_{E_m}^l$; $val_{E_n}^{kjPK}$ - значення j-го атрибуту, який бере участь в утворенні первинного ключа, що є присутнім в екземплярі сутності $e_{E_n}^k$; $val_{E_m}^{ljFK}$ - значення j-го атрибуту, який бере участь в утворенні зовнішнього ключа, що є присутнім в екземплярі сутності $e_{E_m}^l$; D^j - домен атрибутів $atr_{E_n}^{jPK}$ та $atr_{E_m}^{jFK}$.

Тоді (6) для елементів підмножини R^{om} можна перетворити так [5]

$$R_i^{0m} = \langle n_{R_i}, FK_{E_m}^{R_i}, PK_{E_n}^{R_i}, Pow_{E_m}^{R_i}, Pow_{E_n}^{R_i}, S_{E_m}^{R_i}, S_{E_n}^{R_i} \rangle \quad (18)$$

за умови

$$\begin{cases} Pow_{E_m}^{R_i} \geq 1; \\ Pow_{E_n}^{R_i} = 1. \end{cases} \quad (19)$$

Зв'язок R_i^{0m} буде обов'язковим з боку сутності E_n , якщо виконується умова $S_{E_n}^{R_i} = 1$.

Зв'язок R_i^{0m} буде ідентифікуючим, якщо вірна умова [5]

$$\begin{cases} FK_{E_m}^{R_i} \subseteq PK_{E_n}^{R_i}; \\ S_{E_n}^{R_i} = 1. \end{cases} \quad (20)$$

При цьому сутність E_m є слабкою сутністю (її первинний ключ може бути визначений без участі екземпляра сутності E_n), а E_n - сильною сутністю (за умови, що вона не є слабкою сутністю в іншому зв'язку).

Використання запропонованого математичного апарату дозволяє вирішувати задачі кількісного оцінювання складності актуальної БД, складності сукупності запитів на зміни інформаційної системи, яка експлуатується, а також обсягу змін, які слід внести під час рефакторингу БД, що пропонується, шляхом підрахунків кількості атрибутів, які беруть участь в утворенні сутностей та зв'язків, виділених у описах відповідних БД та сукупності запитів на зміни.

5. Результати розробки кількісних показників складності актуальної бази даних та сукупності запитів на зміни інформаційної системи, яка експлуатується

Виходячи із запропонованого у [5] теоретико-множинного апарату опису ERD, складність актуальної БД можна оцінити через кількість окремих об'єктів, з яких складається ця БД. У загальному випадку, виходячи з (1), цю кількість можна підрахувати таким чином

$$C_{DB} = r(E) + r(R) + r(D). \quad (21)$$

де $r(\bullet)$ - функція, що визначає потужність відповідних множин.

Однак, у (21) не враховані особливості проведення рефакторингу. Зокрема, передбачається, що для загального домену БД під час рефакторингу, викликаного найчастіше додаванням і зміною окремих послуг, зміни не очікуються. Тому підраховувати кількість описів елементів домену БД як таких, що впливають на складність робіт з рефакторингу цієї БД, загалом недоцільно. Крім того, у (21) відсутня можливість уніфікованого опису сутностей та зв'язків актуальної БД.

Така можливість уніфікованого опису виникає, коли під час кількісного оцінювання складності актуальної БД відбувається перехід від підрахунків окремих сутностей та зв'язків цієї БД до підрахунків окремих атрибутів, що беруть участь в утворенні відповідних сутностей та зв'язків. Так кількісну оцінку складності сутностей, що утворюють актуальну БД, виходячи з (2) та (3), пропонується визначити як кількість атрибутів, які утворюють заголовок кожної конкретної сутності, яка входить до опису актуальної БД. Цю кількість атрибутів можна підрахувати за виразом

$$C_{DBE} = \sum_{i=1}^l r(\langle atr_{E_i}^j \rangle). \quad (22)$$

За аналогією з (22), кількісну оцінку складності зв'язків, наявних між сутностями актуальної БД, пропонується визначити як кількість атрибутів, які беруть участь в утворенні кожного конкретного зв'язку між батьківською сутністю E_m та дочірньою сутністю E_n . Цю кількість атрибутів можна підрахувати за виразом:

$$C_{DBR} = \sum_{i=1}^z r(Attr_{E_m}^{R_i}) + \sum_{i=1}^z r(Attr_{E_n}^{R_i}), \quad (23)$$

де z - в даному випадку кількість зв'язків, визначених на сутностях множини E актуальної БД та елементах цих сутностей.

Тоді кількісну оцінку складності актуальної БД (21) можна, з урахуванням (22) та (23), представити таким чином

$$C_{DB} = \sum_{i=1}^z r(< atr_{E_i}^j >) + \left(\sum_{i=1}^z r(Atr_{E_m}^{R_i}) + \sum_{i=1}^z r(Atr_{E_n}^{R_i}) \right). \quad (24)$$

Для кількісної оцінки складності сукупності запитів на зміни інформаційної системи, яка експлуатується, визначимо можливість опису таких запитів на зміну класичною моделлю вимог, запропонованою у методології SSADM [6]. За цією моделлю кожний запит на зміну може бути описаний відповідною ERD. Тому, виходячи з (1), множина ERD, які описують усю сукупність запитів на зміну, може бути описана таким чином

$$ERD_{RFC} = \bigcup_{r=1}^q ERD_{RFC_r} = \bigcup_{r=1}^q < E_r, R_r, D_r >, \quad (25)$$

де r - ідентифікатор ERD, яка використовується для опису r -го запиту на зміну; q - кількість запитів на зміну у сукупності, на основі якої робиться оцінювання можливості проведення рефакторингу БД.

Виходячи з (21), в загальному випадку кількість окремих об'єктів, з яких складаються ERD, що описують сукупність запитів на зміни, можна підрахувати наступним чином

$$C_{RFC} = \bigcup_{r=1}^q r(E_r) + \bigcup_{r=1}^q r(R_r) + \bigcup_{r=1}^q r(D_r). \quad (26)$$

Однак ця формула не враховує особливості проведення рефакторингу БД. Як показано вище, для загального домену БД під час рефакторингу зміни не очікуються. Тому врахування останнього доданку у (26) перестає бути необхідним. Крім того, у (26) не врахована можливість використання для опису кожного конкретного запиту на зміну таких атрибутів, які будуть залишатися незмінними під час рефакторингу БД. Це означає, що для кількісного оцінювання складності запитів на зміни інформаційної системи, яка експлуатується, треба враховувати тільки ті атрибути, описи яких у цих запитах повністю або частково відрізняються від описів аналогічних атрибутів у актуальній БД. Виходячи з цього положення, визначимо оцінку складності сукупності запитів на зміни як оцінку кількості елементів у множинах, які є результатами різниці множини описів атрибутів з ERD, яка описує кожен запит на зміну, та множини описів атрибутів з ERD актуальної БД.

Для розрахунку такої оцінки введемо такі підмножини:

$$Atr_{E_{DB}} = \bigcup_{i=1}^t < atr_{E_i}^j >; \quad Atr_{E_{RFC}} = \bigcup_{r=1}^q \bigcup_{i=1}^t < atr_{E_i}^j >. \quad (27)$$

Ці підмножини описують, відповідно, підмножину атрибутів актуальної БД та підмножину атрибутів сукупності запитів на зміни інформаційної системи, яка експлуатується.

Тоді оцінку складності сутностей, що утворюють ERD сукупності запитів на зміни, можна розрахувати за формулою

$$C_{E_{RFC}} = r(Atr_{E_{RFC}} \setminus Atr_{E_{DB}}). \quad (28)$$

Для оцінювання складності зв'язків, що утворюють ERD сукупності запитів на зміни, введемо такі підмножини:

$$Atr_{E_m}^{DB} = \bigcup_{i=1}^t < atr_{E_m}^{R_i} >; \quad Atr_{E_n}^{DB} = \bigcup_{i=1}^t < atr_{E_n}^{R_i} >; \quad (29)$$

$$Atr_{E_m}^{RFC} = \bigcup_{r=1}^q \bigcup_{i=1}^t < atr_{E_m}^{R_{ir}} >; \quad Atr_{E_n}^{RFC} = \bigcup_{r=1}^q \bigcup_{i=1}^t < atr_{E_n}^{R_{ir}} >. \quad (30)$$

Ці підмножини описують сукупності атрибутів, які беруть участь в утворенні зв'язків між сутностями ERD актуальної БД (вираз (29)) та сутностями сукупності ERD запитів на зміни (вираз (30)).

Тоді оцінку складності зв'язків, що утворюють ERD сукупності запитів на зміни, можна розрахувати за формулою

$$C_{RFCR} = r\left(Atr_{E_m}^{RFC} \setminus Atr_{E_m}^{DB}\right) + r\left(Atr_{E_n}^{RFC} \setminus Atr_{E_n}^{DB}\right). \quad (31)$$

Виходячи з отриманих результатів, оцінку (26) складності сукупності запитів на зміни інформаційної системи, яка експлуатується, можна описати виразом

$$C_{RFC} = r\left(Atr_{ERFC} \setminus Atr_{EDB}\right) + \left(r\left(Atr_{E_m}^{RFC} \setminus Atr_{E_m}^{DB}\right) + r\left(Atr_{E_n}^{RFC} \setminus Atr_{E_n}^{DB}\right)\right). \quad (32)$$

6. Результати розробки показника кількісного оцінювання обсягу змін, які слід внести під час рефакторингу бази даних

Отримані у вигляді (24) та (32) оцінки складності актуальної БД та сукупності запитів на зміни інформаційної системи, яка експлуатується, пропонується використати для кількісного оцінювання обсягу змін, які слід внести під час рефакторингу БД. Але, на відміну від абсолютних кількісних оцінок (24) та (32), показник, який пропонується використовувати для кількісного оцінювання такого обсягу змін, слід розробити відносним. Це дозволить спростити прийняття рішення щодо верхньої межі доцільності проведення рефакторингу БД, за якою процес власне рефакторингу БД переходить у процес повного реінжинірингу цієї ж БД.

В загальному випадку цей показник буде мати вигляд

$$C_{REF} = C_{RFC} / C_{DB}. \quad (33)$$

З урахуванням (24) та (32), вираз (33) матиме вигляд

$$C_{REF} = \frac{r\left(Atr_{ERFC} \setminus Atr_{EDB}\right) + \left(r\left(Atr_{E_m}^{RFC} \setminus Atr_{E_m}^{DB}\right) + r\left(Atr_{E_n}^{RFC} \setminus Atr_{E_n}^{DB}\right)\right)}{\sum_{i=1}^l r(< atr_{E_i}^j >) + \left(\sum_{i=1}^{\bar{z}} r\left(Atr_{E_m}^{R_i}\right) + \sum_{i=1}^{\bar{z}} r\left(Atr_{E_n}^{R_i}\right)\right)}. \quad (34)$$

Для спрощення технології розрахунку значень показника (34) визнаємо як вірні такі рівняння

$$r\left(Atr_{EDB}\right) = \sum_{i=1}^l r(< atr_{E_i}^j >); \quad r\left(Atr_{E_m}^{DB}\right) = \sum_{i=1}^{\bar{z}} r\left(Atr_{E_m}^{R_i}\right); \quad r\left(Atr_{E_n}^{DB}\right) = \sum_{i=1}^{\bar{z}} r\left(Atr_{E_n}^{R_i}\right). \quad (35)$$

Тоді правила розрахунку значень за (35) технологічно спростяться за рахунок уніфікації процедур формування окремих підмножин, а сам вираз (35) матиме вигляд

$$C_{REF} = \frac{r\left(Atr_{ERFC} \setminus Atr_{EDB}\right) + \left(r\left(Atr_{E_m}^{RFC} \setminus Atr_{E_m}^{DB}\right) + r\left(Atr_{E_n}^{RFC} \setminus Atr_{E_n}^{DB}\right)\right)}{r\left(Atr_{EDB}\right) + \left(r\left(Atr_{E_m}^{DB}\right) + r\left(Atr_{E_n}^{DB}\right)\right)}. \quad (36)$$

Використання запропонованого показника (36) дозволяє визначити верхню межу доцільності проведення рефакторингу БД. Виходячи з практичного досвіду, тут і в подальшому пропонується визначити процес рефакторингу доцільним, якщо значення показника (36) не буде перевищувати 0,5. Якщо показник (36) під час визначення доцільності проведення рефакторингу БД прийматиме значення, більші за 0,5 (тобто при виконанні запитів на зміни потрібно змінити більше половини актуальної БД), то слід визнати відповідні роботи роботами з реінжинірингу БД. Але цей висновок слід вважати попереднім і таким, що потребує підтвердження чи спростування шляхом подальших досліджень.

7. Висновки і перспективи подальших досліджень

Розроблені формальні описи оцінок складності актуальної БД та сукупності запитів на зміни інформаційної системи, яка експлуатується, дозволяють оцінити кількість елементів, з

яких складаються ERD, що описують актуальну БД та кожен із сукупності запитів на зміни. В основу запропонованих оцінок покладено теоретико-множинний апарат, який описує, у тому числі, ERD через кортежі та підмножини атрибутів. Використання такого уніфікованого формального опису дозволяє в подальшому використовувати для автоматизованого розрахунку значень цих оцінок одні й ті ж процедури.

На основі розроблених формальних оцінок було запропоновано показник кількісного оцінювання обсягу змін, які слід внести під час рефакторингу БД. Визначено два основних варіанти розрахунку цього показника за виразами (34) та (36) відповідно. Описано особливості використання значень цього показника під час процесу перевірки доцільності проведення робіт з рефакторингу БД.

Слід зазначити, що отримані результати дозволяють визначити складність та обсяг змін, які слід внести під час рефакторингу БД. Але ці результати не дозволяють оцінити доцільність проведення робіт з рефакторингу БД як окремого різновиду ІТ-проектів. Тому найближча перспектива подальших досліджень полягає у модифікації існуючих моделей оцінювання ІТ-проектів (зокрема, параметричних моделей СОСОМО II) для оцінювання трудовитрат, витрат часу та потреби у персоналі ІТ-проекту рефакторингу БД на основі отриманих оцінок.

Список літератури: 1. *Мірошниченко Д.О.* Дослідження моделей і методів управління ІТ-проектом рефакторингу бази даних Інтернет-провайдера задля підтримки гнучкого створення замовлень. 27-й Міжнародний молодіжний форум "Радіоелектроніка та молодь XXI століття". Зб. матеріалів форуму. Т.6. Конференція "Інформаційні інтелектуальні системи". Харків: ХНУРЕ. 2023. С.258-259. 2. *Ambler S. W., Sadalage P.* Refactoring Databases: Evolutionary Database Design. Addison-Wesley Longman, Incorporated, 2006. 384 с. 3. *The Process of Database Refactoring: Strategies for Improving Database Quality.* Agile Data. URL: <http://agiledata.org/essays/databaseRefactoring.html> (дата звернення: 07.04.2023 р.). 4. *Data Quality Management Model (2015 Update) - Retired* // AHIMA's HIM Body of Knowledge. URL: <https://library.ahima.org/PB/DataQualityModel#.ZDBZmXvP2Uk> (дата звернення: 07.04.2023 р.). 5. *Левыкин В.М., Евланов М.В., Сугробов В.С.* Параллельное проектирование информационного и программного комплексов информационной системы. Радиотехника. 2006. Вып. 146. С. 89-98. 6. *Петров Э.Г., Чайников С.И., Овезгельдыев А.О.* Методология структурного системного анализа и проектирования крупномасштабных ИУС. Концепции и методы. Харьков: Рубикон, 1997. Часть 1. 140 с.

Надійшла до редколегії 17.11.2022

Євланов Максим Вікторович, доктор технічних наук, доцент, професор кафедри ІУС ХНУРЕ. Наукові інтереси: методи, моделі та інформаційні технології інженерії вимог до інформаційних систем. Адреса: Харків, 61166, пр. Науки, 14. Контактний телефон: +38(057)7021451.

Юр'єв Іван Олексійович, кандидат технічних наук, доцент, доцент кафедри ІУС ХНУРЕ. Наукові інтереси: методи, моделі та інформаційні технології управління інформаційними системами та ІТ-сервісами. Адреса: Харків, 61166, пр. Науки, 14. Контактний телефон: +38(057)7021451.

Мірошниченко Дмитро Олександрович, здобувач вищої освіти, група УППТм-21-1. Наукові інтереси: дослідження моделей і методів управління ІТ-проектом рефакторингу бази даних Internet-провайдера. Адреса: Харків, 61166, пр. Науки, 14. Контактний телефон: +38(057)7021451.