

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ОДЕСЬКА НАЦІОНАЛЬНА АКАДЕМІЯ ХАРЧОВИХ ТЕХНОЛОГІЙ

**ХІ МІЖНАРОДНА
НАУКОВО-ПРАКТИЧНА
КОНФЕРЕНЦІЯ**

**ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ І
АВТОМАТИЗАЦІЯ – 2018**

Збірник доповідей

Частина II

Одеса,
4-5 жовтня 2018

ЗМІСТ

<i>МОРОЗ А. Н.</i>	3
<i>НОЖКО Т.Г.</i>	4
<i>УЕНОРОВ В.В., РОНЛЕВИНА Н.О.</i>	6
<i>РОМАНЮК О.Н., ЛИСЕНКО Є.С., ВОЙТ Б.Л.</i>	7
<i>РОМАНЮК С. О., НЕЧИПОРУК М. Л.</i>	10
<i>РОМАНЮК О. Н., ПАНФІЛОВА Ю. О., ЧАН А. Л. В.</i>	13
<i>РИБАЛКО І. І., БОГДАНОВА Л. М., АНОСОВ В. Л.</i>	16
<i>СКАКОВСЬКИЙ Ю.М., БАБКОВ А.В.</i>	17
<i>СТАНОВЬКА Т.П., СПРОМЛЯ С.Г., БОЛТАЧ С.В.</i>	20
<i>СУЛІМА Ю.Ю., СУЛІМА Ю.Є.</i>	22
<i>ТРАЧ Н.Р., ВОЛКОВ В.Э.</i>	24
<i>ЮРЧЕНКО В. В., БОГДАНОВА Л. М., АНОСОВ В. Л.</i>	25
<i>УАНАКОВ В.Р.</i>	27
<i>ГНАТЕНКО В.Ю., СТУПЕНЬ П.В.</i>	29
<i>ЛЕОНТЬЄВА І.О., ХОБІН В.А.</i>	31
<i>КОРНІЄНКО Ю.К., БОЙЦОВА О.С., ШАМРАЙ О.А.</i>	33
<i>КОРНІЄНКО Ю.К., КОТЛИК С.В., БОЙЦОВА О.С., ШАМРАЙ О.А.</i>	35
<i>ІВАНОВА А.Г., ОЛЬШЕВСЬКА О.В.</i>	38
<i>ШЕРШУН О.О., ОЛЬШЕВСЬКА О.В.</i>	40
<i>ВОЛКОВА А.Ю., ПРУС В.В., ОЛЬШЕВСЬКА О.В.</i>	42
<i>ХАРАШ К.М., ОЛЬШЕВСЬКА О.В.</i>	43
<i>БОГДАНОВ А.С., КОРНІЄНКО Ю.К.</i>	45
<i>СКАЛІЙ Д.О., ОЛЬШЕВСЬКА О.В.</i>	47
<i>ДЖИДЖУЛА М.В., КОРНІЄНКО Ю.К.</i>	48
<i>ЄПІФАНОВА А.О., КОРЖАН В.С., ОЛЬШЕВСЬКА О.В., ЛОМОВЦЕВ П.Б.</i>	49

УНІВЕРСАЛЬНІ МОДЕЛІ ДАНИХ

Існує декілька реалізацій специфічної моделі даних, яку прийнято називати універсальною. Така модель складається з ієрархічно пов'язаного набору таблиць, що представляють метадані та дані додатка. Універсальна схема зберігає будь-яке число віртуальних схем, заповнених даними. Типова схема УМД має таку структуру: таблиці «Схема» що складається з стовбців «ідентифікатор схеми» та «Назва схеми», таблиці «Таблиця» що складається з стовбців «ідентифікатор схеми», «ідентифікатор таблиці» та «назва таблиці», таблиці «Стовбець» яка складається з стовбців «ідентифікатор схеми», «ідентифікатор таблиці», «ідентифікатор стовбця» та «назва стовбця», а також таблиці «Дані» яка містить у собі такі стовбці як «ідентифікатор схеми», «ідентифікатор таблиці», «ідентифікатор стовбця» та «значення». У ряді реалізацій дані зберігаються в кількох таблицях, кожна з яких містить дані одного типу, наприклад, все числові дані. Зазвичай замість запиту до віртуальної схеми створюється вручну складний еквівалентний запит до схеми УМД з сполуками всіх або майже всіх реальних таблиць УМД. Раніше вважалося, що універсальна модель даних може послужити основою для реалізації будь-яких баз даних. Таке припущення на-вряд чи виправдано тому що зміна структури даних під час роботи програмного продукту з базою даних стається дуже рідко. Але існують два типи систем, в яких схема бази даних постійно перебу-дується. Це інформаційні системи с даними, структури яких неможливо передбачити підчас проектування системи, типовий приклад – медичні дані. А також засоби розробки інформаційних систем.

Основними недоліками універсальної моделі даних є: дуже складні SQL-запити, низька швидкість опрацювання запитів системою управління базою даних, а також відсутність в багатьох реалізаціях обмежень цілісності, індексів, користувачів, ролей та уявлень.

Складність SQL-запитів була вирішена створенням ретрансляторів з мови Query-by-Example яка використовується для запитів до віртуальної бази даних. Ретранслятори у свою чергу перетворюють запити для віртуальної бази даних у запити до реальних таблиць системи управління базами даних. Але цей підхід не вирішує проблему з низькою швидкістю обробки запитів системою управління базою даних. Є другий варіант рішення цієї проблеми, використання методу Т. Кайта для транспонування рядків у стовбці. Згенерований SQL-запит після використання методу Т. Кайта оброблюється системою управління базою даних набагато швидше. Також можливо покращити швидкість завдяки створенню індексів на таблиці даних [1, 2].

Використовуючи універсальну схему, можливо створювати моделі даних, відмінні від моделей реляційного типу. Розглянемо розширення універсальної моделі даних для об'єктно-реляційних моделей. У користувача повинна з'явитися можливість визначати свої типи даних і використовувати їх в якості типів стовбців і таблиць. Об'єкт може мати атрибути і методи. При цьому атрибут також може мати об'єктний тип. Об'єктний тип може бути присвоєний стовбцю або усій таблиці. На основі таблиць словника до універсальної моделі даних додається набір таблиць і зв'язків для реалізації описаних властивостей. Тіло методу може бути виконано за допомогою динамічного SQL. Об'єктна модель відрізняється від об'єктно-реляційної, зокрема, тим, що для деяких полів об'єктів можуть бути задані рівні доступу, відмінні від «public», які мають всі поля об'єктно-реляційної моделі. Для цього можна додати до таблиць «Стовпець» і «Метод» поля, значеннями яких будуть уровни доступу «public» та «private». Для роботи з такими стовпцями і методами доведеться змінити інтерпретацію і дозволити користувачам звертатися безпосередньо тільки до тих полів, які мають рівень доступу «public»[3].

Атрибути у традиційних базах даних завжди повинні використовуватися. Полуструктуровані дані відрізняються тим, що деякі атрибути присутні тільки в окремих примірниках сутностей. Такі атрибути називаються необов'язковими. Набір обов'язкових атрибутів визначає так званий мінімальний DataGuide. Він грає ту ж роль, що і схеми звичайних таблиць. Набір обов'язкових і необов'язкових атрибутів становить максимальний DataGuide. Він дозволяє визначити, чи використовується стовпець в перший раз, або ж фігурував раніше.

Для реалізації полуструктурованої моделі можна використовувати метод заснований не на розширенні схеми УМД, а на її новій інтерпретації і зміні логіки роботи транслятора. Будемо вважати, що стовпці, описані в таблиці «Стовпець» є обов'язковими, тобто входять до мінімального DataGuide. Щоб дати користувачеві можливість вставляти в значення в віртуальні поля таблиці «Дані», що не входять в мінімальний DataGuide, необхідно відключити перевірку існування стовпців в таблиці. Можна додати в таблицю «Дані» стовпець «тип_даних», в який записується тип даних необов'язкових атрибутів, який визначається транслятором. Також можна пропонувати користувачеві після вставки визначати тип значень необов'язкових стовпців що вставляються.

Універсальні моделі даних оптимальне рішення при створенні бази даних для інформаційних систем у яких завжди відбувається зміна схем, тобто логічні та архітектурні зв'язки між даними. Хоча при використанні УМД є незначна проблема зі швидкістю виконання запитів системою управління базами даних, це повністю нівелюється можливістю створення стабільної бази даних, а також можливість легко та прозоро змінювати структуру даних у моделі.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. *Универсальные модели данных.* [Online]. Available: <https://rsdn.org/article/db/51-55-model.xml>. [Accessed: 27-Sep-2018].
 2. *Секреты поваров компьютерной кухни или ПК: решение проблем.* [Online]. Available: <http://citforum.ck.ua/seminars/cbd2002/111.shtml>. [Accessed: 27-Sep-2018].
- “Сетевая модель данных,” *Wikipedia*, 16-Nov-2017. [Online]. Available: https://ru.m.wikipedia.org/wiki/Сетевая_модель_данных. [Accessed: 27-Sep-2018].

XI МІЖНАРОДНА НАУКОВО-ПРАКТИЧНА КОНФЕРЕНЦІЯ

ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ І АВТОМАТИЗАЦІЯ – 2018

ОДЕСА
4 – 5 ЖОВТНЯ, 2018

Збірник включає доповіді учасників XI Міжнародної науково-практичної конференції «Інформаційні технології і автоматизація – 2018»

Редакційна колегія: Котлик С.В., Хобін В.А.

Комп'ютерний набір і верстка: Шамрай О.А.

Відповідальний за випуск: Котлик С.В.

НТТБ ОНАХТ

