Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя
(повне найменування вищого навчального закладу)
Факультет комп'ютерно-інформаційних систем і програмної інженерії
(назва факультету)
Кафедра комп'ютерних наук
(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

Магістр
(назва освітнього ступеня)

на тему: Математичне і програмне забезпечення СППР на основі вітрин великих даних

Виконав: студент 6 курсу, групи СНнм-61
спеціальності 122 Комп'ютерні науки
(шифр і назва спеціальності)

_____ Юрчак В.В.
(підпис) (прізвище та ініціали)

Керівник _____ проф. Литвиненко Я.В.
(підпис) (прізвище та ініціали)

Нормоконтроль _____ доц. Дуда О.М.
(підпис) (прізвище та ініціали)

Завідувач кафедри _____ доц. Боднарчук І.О.
(підпис) (прізвище та ініціали)

Рецензент _____ _____
(підпис) (прізвище та ініціали)

м. Тернопіль – 2025

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра комп'ютерних наук
(повна назва кафедри)

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ доц. Боднарчук І.О.
(підпис) (прізвище та ініціали)
« » 20___ р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня магістр
(назва освітнього ступеня)

за спеціальністю 122 Комп'ютерні науки
(шифр і назва спеціальності)

студенту Юрчаку Валерію Володимировичу

1. Тема роботи Математичне і програмне забезпечення СППР на основі вітрин великих даних

Керівник роботи Литвиненко Ярослав Володимирович, д.т.н, проф. каф. КН
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом по університету від «__29__» _листопада_ 2023 року №__4/7-1139_

2. Термін подання студентом роботи 29.05.2025

3. Вихідні дані до роботи наукові літературні джерела

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)
1 Аналіз предметної області.
2. Теоретична частина. 3. Практична частина.
4. Охорона праці та безпека в надзвичайних ситуаціях

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)
1. Тема роботи. 2. Актуальність. 3. Мета, задачі дослідження. 4. Об'єкт, предмет дослідження наукова новизна, практичне значення роботи. 5. Моделі СППР на основі вітрин даних
6. Порівняння методів побудови вітрин даних.
7. Технології побудови BI -систем для аналізу великих обсягів даних
8. Діаграма діяльності процесу проектування ПЗ BI-системи. 9. Паттерни проектування.
10. Діаграма варіантів використання BI-системи.
11. Метамодель BI-системи на основі логічної вітрини даних. 12. Кортеж для шаблону даних
13. Логічна модель даних логічної вітрини даних кредитної організації
14. Програмна архітектура СППР на базі логічної вітрини даних
15. Порівняльний аналіз характеристик BI –систем.
16. Оцінка ефективності BI-системи. 17. Основні результати дослідження.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці	Сенчишин В.С., доцент каф. МТ		
Безпека в надзвичайних			
ситуаціях	Клепчик В.М., проректор з АГРБ		

7. Дата видачі завдання _____ 2024 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Термін виконання етапів роботи	Примітка
1	Затвердження теми кваліфікаційної роботи	29.11.24	Виконано
2	Аналіз літературних джерел	30.11.24–10.12.24	Виконано
3	Обґрунтування актуальності дослідження	11.12.24–20.12.24	Виконано
4	Аналіз предмету дослідження та предметної області	21.12.24–31.12.24	Виконано
5	Проведення дослідження методів та засобів аналітичного опрацювання даних	01.01.25 –20.01.25	Виконано
6	Оформлення розділу «Аналіз предметної області»	21.01.25–10.02.25	Виконано
7	Оформлення розділу «Теоретична частина»	11.02.25–13.03.25	Виконано
8	Оформлення розділу «Практична частина»	14.03.25–15.04.25	Виконано
9	Оформлення розділу «Охорона праці та безпека в надзвичайних ситуаціях»	16.04.25–22.04.25	Виконано
10	Нормоконтроль	15.04.25–30.04.25	Виконано
11	Перевірка на плагіат	01.05.25–10.05.25	Виконано
12	Попередній захист роботи	14.05.25	Виконано
13	Захист кваліфікаційної роботи	30.05.25	

Студент _____ Юрчак В.В.
(підпис) (прізвище та ініціали)

Керівник роботи _____ Литвиненко Я.В.
(підпис) (прізвище та ініціали)

АНОТАЦІЯ

Математичне і програмне забезпечення СППР на основі вітрин великих даних // Кваліфікаційна робота освітнього рівня «Магістр» // Юрчак Валерій Володимирович // Тернопільський національний технічний університетімені Івана Пулюя, факультет комп'ютерно-інформаційних систем та програмної інженерії, кафедра комп'ютерних наук, група СНнм–61 // Тернопіль, 2025 // С. – 70, рис. – 27, табл.– 7, слайдів – 17, додат. – 1, бібліогр. – 46.

Ключові слова: великі дані, вітрина даних, сховище даних, СППР, BI-система, OLAP

У першому розділі дано аналіз сучасного стану досліджень у галузі розробки математичного та програмного забезпечення для СППР в управлінні на базі вітрин даних. Розглянуто моделі СППР на базі вітрин даних та аналізу великих даних.

Другий розділ присвячено аналізу методологій та технологій побудови систем підтримки прийняття рішень на основі вітрин великих даних.

Третій розділ присвячений розробці математичного забезпечення СППР на основі вітрин великих даних. Розглянуто процес логічного моделювання СППР із застосуванням вітрин великих даних. Описано процес побудови логічної моделі даних СППР із використанням вітрин великих даних. Також продемонстровано послідовність створення програмного забезпечення такої СППР. Наведено результати апробації обраних проектних рішень та оцінки ефективності BI-системи.

ANNOTATION

Mathematical and Software Support for Decision Support Systems Based on Big Data Marts // Yurchak Valerii // Ternopil Ivan Pul'uj National Technical University, Faculty of Computer Information Systems and Software Engineering, Department of Computer Science, SNnm-61 group // Ternopil, 2025 // P. - 70, Fig. - 27, Table – 7, Slide - 17, References - 46.

Keywords: big data, data mart, data warehouse, DSS, BI system, OLAP

The first chapter provides an analysis of the current state of research in the field of developing mathematical and software for DSS in management based on data marts. DSS models based on data marts and big data analysis are considered.

The second chapter is devoted to the analysis of methodologies and technologies for building decision support systems based on big data marts.

The third chapter is devoted to the development of mathematical DSS software based on big data marts. The process of logical modeling of DSS using big data marts is considered. The process of building a logical DSS data model using big data marts is described. The sequence of creating software for such DSS is also demonstrated. The results of testing selected design solutions and assessing the effectiveness of the BI system are presented.

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ СКОРОЧЕНЬ І ТЕРМІНІВ

BA/BI (Business Anlytics/Business Intelligence) –бізнес-аналітика.

ETL (Extract, Transform, Load - Витяг, Перетворення, Завантаження) – процес, який використовується в базах даних та, особливо, у сховищах даних та у засобах BI для забезпечення їх роботи для підтримки прийняття рішень

MDA (Model Driven Architecture) – різновид модельно-орієнтованого підходу до розробки програмного забезпечення.

LSA (Layered Scalable Architecture) – архітектура листового пирога.

OLAP (online analytical processing) – аналітична обробка у реальному часі.

ROLAP (Relational OLAP) – реляційна онлайн-аналітична обробка.

RUP (Rational Unified Process) – ітеративний процес розробки програмного забезпечення.

UML (Unified Modeling Language) – уніфікована мова моделювання.

БД – база даних.

ВВД – вітрини великих даних.

Вітрина даних (Data Mart) - це структура або шаблон доступу, який визначає правила отримання даних із сховища даних для певних груп користувачів або цілей.

ІС – інформаційна система.

МПЗ – математичне та програмне забезпечення.

МПЗ ІС - сукупність математичних методів, моделей, алгоритмів та програм для реалізації цілей та завдань ІС, а також нормального функціонування комплексу технічних засобів

Озеро даних (Date Lake) - це централізоване сховище, яке дозволяє зберігати структуровані та неструктуровані дані у будь-якому масштабі.

ПЗ – програмне забезпечення.

СУБД – система управління базами даних.

СППР (Decision Support System, DSS) – система підтримки прийняття ріщень.

Сховище даних (data warehouse) — предметно орієнтований, інтегрований, незмінний набір даних, що підтримує хронологію і здатний бути комплексним джерелом достовірної інформації для оперативного аналізу та прийняття рішень.

Технологія In-Memory – технологія виконання комп'ютерних обчислень повністю в пам'яті комп'ютера, наприклад, в оперативному пристрої.

ЦСД – централізоване (корпоративне) сховище даних.

ЗМІСТ

Вступ 10
1 Аналіз предметної області 12
1.1 Основні визначення та поняття 12
1.2 Моделі СППР на основі вітрин даних 14
1.3 Моделі СППР на основі бізнес- аналізу великих даних 18
1.4 Висновки до першого розділу 21
2 Теоретична частина 23
2.1 Методи побудови вітрин даних 23
2.1.1 Методи побудови залежних вітрин даних 23
2.1.2 Методи побудови незалежних вітрин даних 26
2.1.3 Методи побудови логічних вітрин даних 28
2.2 Технології побудови BI -систем для аналізу великих обсягів даних 30
2.3 Технологія проектування програмного забезпечення BI- системи 33
2.4 Методології проектування програмного забезпечення BI-системи 36
2.3.1 Методологія об'єктно-орієнтованого аналізу та проектування 36
2.3.2 Методологія об'єктно-структурного моделювання 38
2.5 Висновки до другого розділу 41
3 Практична частина 42
3.1 Математичне забезпечення СППР на базі ВВД 42
3.1.1 Логічне моделювання СППР на базі ВВД 42
3.1.2 Розробка логічної моделі даних СППР на базі ВВД 47
3.2 Програмне забезпечення СППР на базі ВВД 49
3.2.1 BI -система MS Power BI 51
3.2.2 BI -система Tableu 52
3.2.3 BI -система QlikView 53
3.3 Оцінка ефективності BI-системи 55
3.4 Висновки до третього розділу 56
4 Охорона праці та безпека в надзвичайних ситуаціях 58

4.1 Режими праці і відпочинку при роботі з ЕОМ 58
4.2 Вплив електромагнітного імпульсу ядерного вибуху на елементи виробництва та заходи захисту 60
4.3 Висновки до четвертого розділу 63
Висновки 64
Перелік джерел 65
Додатки

ВСТУП

Актуальність теми. Успішне функціонування фірми чи підприємства прямо залежить від швидкості реагування менеджменту на зміни кон'юнктури ринку із метою збільшення доходу і забезпечення конкурентоспроможності. Менеджери компанії повинні відстежувати ринкові тенденції, ідентифікувати конкурентів і різнорідні загрози, бути здатними оцінювати ризики та свої ресурс, і, як наслідок, приймати обґрунтовані управлінські рішення. Інформація і є, властиво, тим необхідним виробничим ресурсом, котрий вимагається для прийняття рішень, котрі будуть ефективними [1].

З метою розв’язання завдань, котрі перераховані вище, застосовуються аналітичні СППР, котрі належать до категорії BA/BI - систем.

Теперішні предметно-орієнтовані BI – системи побудовані на базі концепції вітрин даних [2]. Вітрина даних є простою формою сховища даних, орієнтованою на визначений напрямок бізнесу чи складову частину компанії. Використовуючи можливості вітрин даних фахівці здатні швидше одержувати доступ до даних, так як їм вони не витрачають час на відшукання складнішого сховища чи на процес ручного агрегування даних із різноманітних джерел [3].

Зараз фірми накопичили значні об’єми даних і володіють доступом до величезних обсягів зовнішніх даних, котрі дають змогу розглядати їх як Big Data.

Тут виникає проблема проектування BI - систем, компонентами яких будуть вітрини великих структурованих масивів даних. Цілком очевидно, що висока ефективність таких систем може бути досягнута за рахунок застосування при їх розробці спеціалізованого МПЗ.

Розробка такого МПЗ, котре дозволяє підвищити ефективність СППР в управління, побудованих із застосуванням ВВД, є надзвичайно цікавим із науково-практичного боку.

Мета дослідження: дослідження та розробка МПЗ СППР, створеної на базі ВВД.

Задачі дослідження:

– виконати аналіз наявного стану предметної області дослідження;

– проаналізувати методологічні підходи до побудови СППР в управління на основі ВВД;

– розробити МПЗ таких систем;

– виконати апробацію запропонованих проектних рішень та оцінити їх ефективність.

Об’єкт дослідження: СППР.

Предмет дослідження: МПЗ СППР.

Методи дослідження: системний аналіз, методології та технології проектування BA/BI – систем, методи та технології аналітичної обробки великих даних.

Наукова новизна роботи:

– запропоновано МПЗ, котре здатне підвищення ефективність СППР, побудованої на основі ВВД.

Практичне значення одержаних результатів. Результати, одержані в дослідженні, дають змогу реального застосування розробленого МПЗ для СППР у підприємствах та компаніях.

Апробація. Деякі результати роботи були апробовані як тези на науково-технічних конференціях:

– Юрчак В. В. Технології побудови BI-систем для аналізу великих даних // Інформаційні моделі, системи та технології: Праці XII наук.-техн. конф. Тернопіль, 2024. с. 113;

– Юрчак В. В. Процес розробки метамоделі об'єктів BI-системи // VIII Міжнародна студентська науково-технічна конференція «Природничі та гуманітарні науки. Актуальні питання». Тернопіль, 2025. с. 226.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Основні визначення та поняття

СППР є комп'ютерною автоматизованою системою, метою котрої є допомога людям, які приймають рішення у складних умовах для абсолютного та об'єктивного аналізування предметної діяльності [4].

Згідно з концепцією компанії Gartner, у технології BI виділяються такі напрямки:

– BI -сервіси (BI services) – це пропозиції для проектування, розроблення і розгортання, властиво, корпоративних процесів, а також для інтегрування, підтримання і керування визначеними платформами і програмами. Рішення включають такі галузі, як корпоративне управління продуктивністю та аналітика, на додачу до традиційної платформи BI, сховищ даних / інфраструктури даних та областей якості даних [5];

– BI -платформи (BI platforms), які дозволяють підприємствам створювати програми бізнес-аналітики, надаючи можливості у трьох категоріях: аналіз (OLAP), доставка інформації (звіти та інформаційні панелі) та платформна інтеграція (управління метаданими бізнес-аналітики та середовище розробки).

Основні вимоги, які пред'являються до сучасних BI-систем:

– оперативний бізнес-аналіз (в реальному часі). Сьогодні через конкурентний тиск підприємств зросла потреба в BA, близькій до реального часу, яка називається операційною BA. Мета операційної BA - зменшити затримку між часом аналізу даних та часом збору даних. Зменшення часу відгуку дозволяє системі робити відповідні дії при виникненні події. За допомогою оперативної реалізації BA компанії можуть виявляти закономірності чи тимчасові тенденції у потоці операційних даних;

– ситуаційний бізнес-аналіз, що забезпечує ситуаційну обізнаність. У компаніях позиціонування BA є важливим там, де швидка зміна позицій,

зазвичай зовнішні тенденції в бізнесі, вплинули на бізнес. Однак ці зовнішні дані, які в основному надходять із внутрішньої мережі компанії, зовнішнього постачальника чи Інтернету, не структуровані. Більше того, ці неструктуровані дані мають бути об'єднані з іншими структурованими даними з локального сховища даних компанії для підтримки маркування рішень у реальному часі. Наприклад, компанія може побажати дізнатися, чи залишають її користувачі та клієнти позитивні чи негативні коментарі про свої нові продукти. Аналізуючи ці коментарі, компанії можуть негайно надсилати свої коментарі команді розробників, щоб зробити продукт більш конкурентоспроможним та кваліфікованим. Інший приклад – для компанії важливо знати, чи вплинули стихійні лиха на їхніх контрактних постачальників. Це створює можливість для керівників компаній вживати відповідних заходів щодо зниження втрат;

– самостійна ВА, що дозволяє кінцевим користувачам створювати аналізи та аналітичні запити без участі IT-відділу. У такій BI -системі інтерфейс додатків повинен бути простим у використанні і інтуїтивно зрозумілим, що не вимагає від користувача технічних знань в області сховищ даних. Крім того, користувач повинен мати доступ або розширення джерел даних, організованих IT-відділом, а також нетрадиційних джерел.

Згідно з цією ж концепцією, одним із ключових компонентів BI- систем є сховище та вітрина даних.

Сховище даних є архітектурою зберігання, призначеною для зберігання даних, вилучених із систем транзакцій, сховищ операційних даних та зовнішніх джерел Сховище об'єднує ці дані в агреговану зведену форму, придатну для аналізу даних у масштабах підприємства та створення звітів для попередньо визначених бізнес-потреб.

Вітрина даних містить дані, які аналогічно змінюються у часі та є предметно-орієнтованими, але зі зв'язками, котрі використовують дані у вимірюваннях, при цьому факти чітко відокремлені від даних вимірювань, що робить їх більш придатними для окремих категорій аналізу [2].

Відмінні характеристики вітрини даних:

– орієнтована лише на запити користувачів, які стосуються конкретної бізнес-мети;

– зазвичай містять оперативні дані;

– містять менші обсяги даних у порівнянні зі сховищами даних та їх легше реалізувати.

Принципи побудови BI -додатків на основі вітрин даних викладені в роботах Р. Кімбалла [6].

1.2 Моделі СППР на основі вітрин даних

Розглянемо моделі систем підтримки ухвалення рішень, побудовані на основі вітрин даних.

У роботі [7] описана структура типового BI-рішення, побудованого на основі вітрин даних, представлена на рис. 1.1.

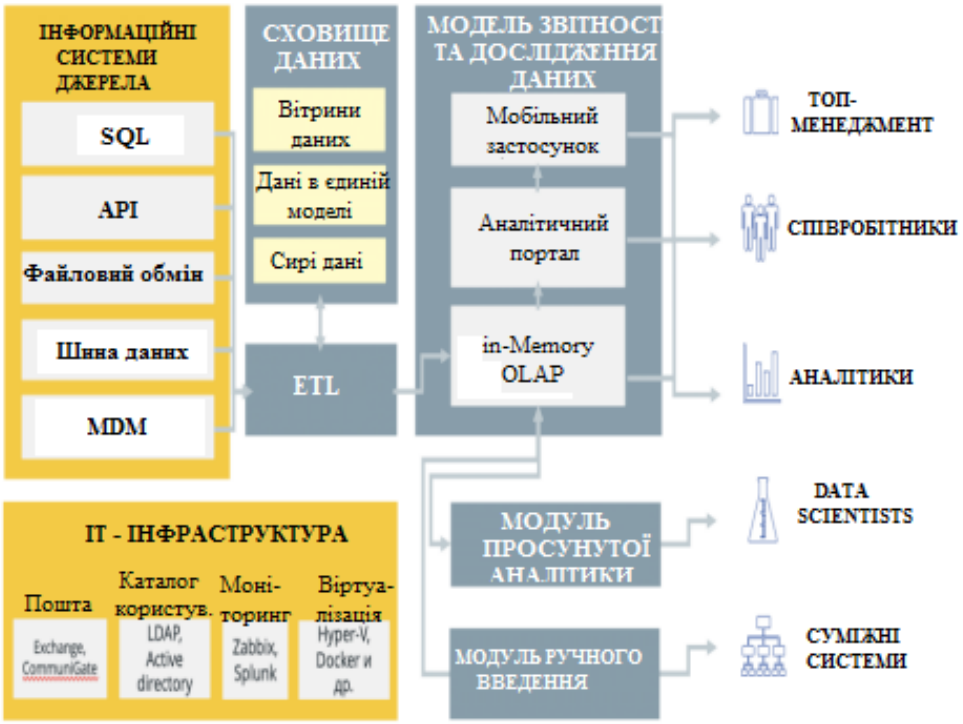

Рисунок 1.1 – Типова структура BI -рішення на основі вітрин даних

Дані, необхідні для роботи BI -платформи, витягуються з різних IT-

систем за допомогою ETL- інструментів і надходять до спеціальних сховищ даних. Перетворення даних на рівні сховища даних, як правило, оркеструються ETL -інструментом. Кінцевим результатом перетворення даних на рівні сховищ даних є вітрини даних - очищені набори даних, яких користувачі отримують прямий доступ за допомогою OLAP інструментів [8].

У роботі [9] описано інформаційно-аналітичну систему управління андеррайтингом, в якій використовується вітрина страхових даних (Insurance data mart).

Структурна схема системи показана на рис. 1.2.

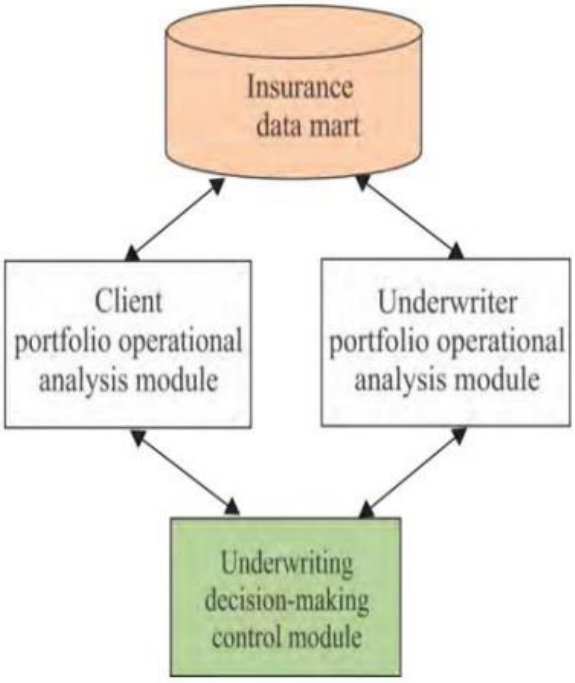

Рисунок 1.2 – Структурна схема інформаційно-аналітичної системи управління андеррайтингом

Вітрина страхових даних побудована на основі технології реляційних OLAP (ROLAP), яка використовує схему "зірка", як показано на рис. 1.3.

Така модель забезпечує найкращу продуктивність запитів за підтримки прийняття рішень

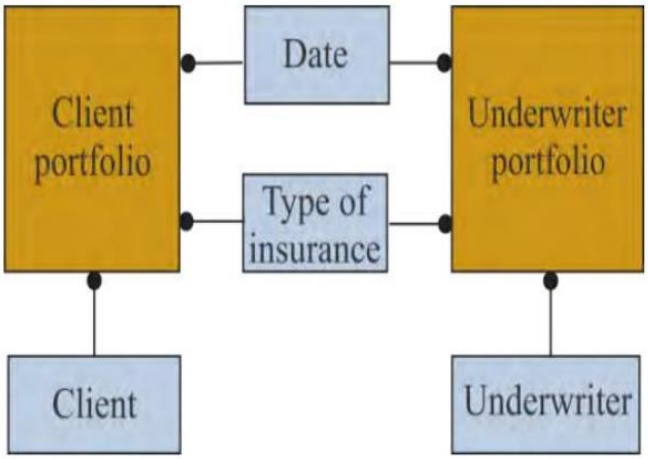

Рисунок 1.3 – Архітектура вітрини страхових даних (Портфель клієнта та Портфель Андеррайтера - таблиці фактів; Дата, Клієнт, Андеррайтер, Тип страхування - таблиці вимірів)

У роботі [10] представлені та обговорюються переваги розробки та реалізації СППР на основі вітрини даних.

Пропонується архітектура ефективної СППР телекомунікаційної компанії, побудованої на основі СУБД MS SQL Server (рис. 1.4).

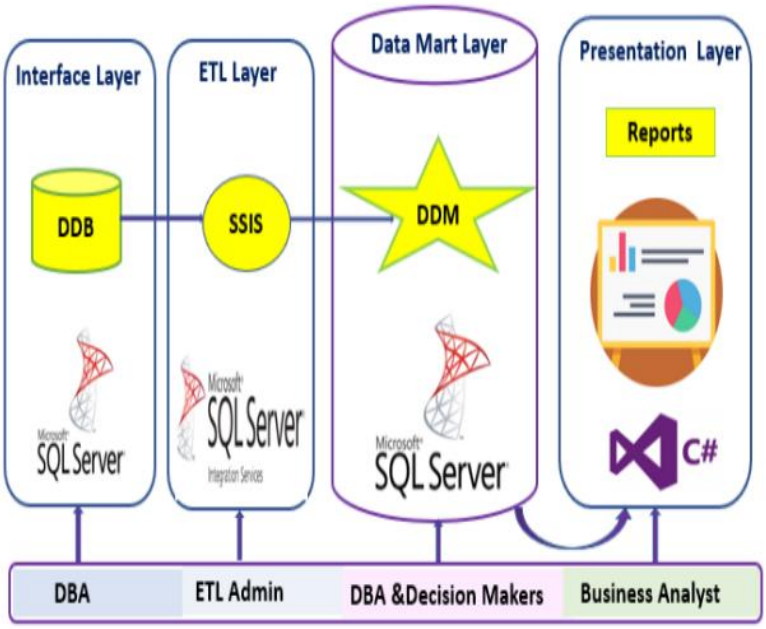

Рисунок 1.4 – Архітектура ефективної СППР для телекомунікаційної компанії на основі вітрини даних

В роботі [11] обговорюються проблеми, пов'язані з системами сховища даних (DW)/ вітрини даних (DM), і пропонується нова архітектура, заснована на мультиагентній технології, для отримання інформації та знань з розподілених джерел даних для підтримки прийняття рішень. Така структура використана у життєвому циклі реального проекту.

Архітектура СППР на основі мультиагентної технології наведена на рис. 1.5.

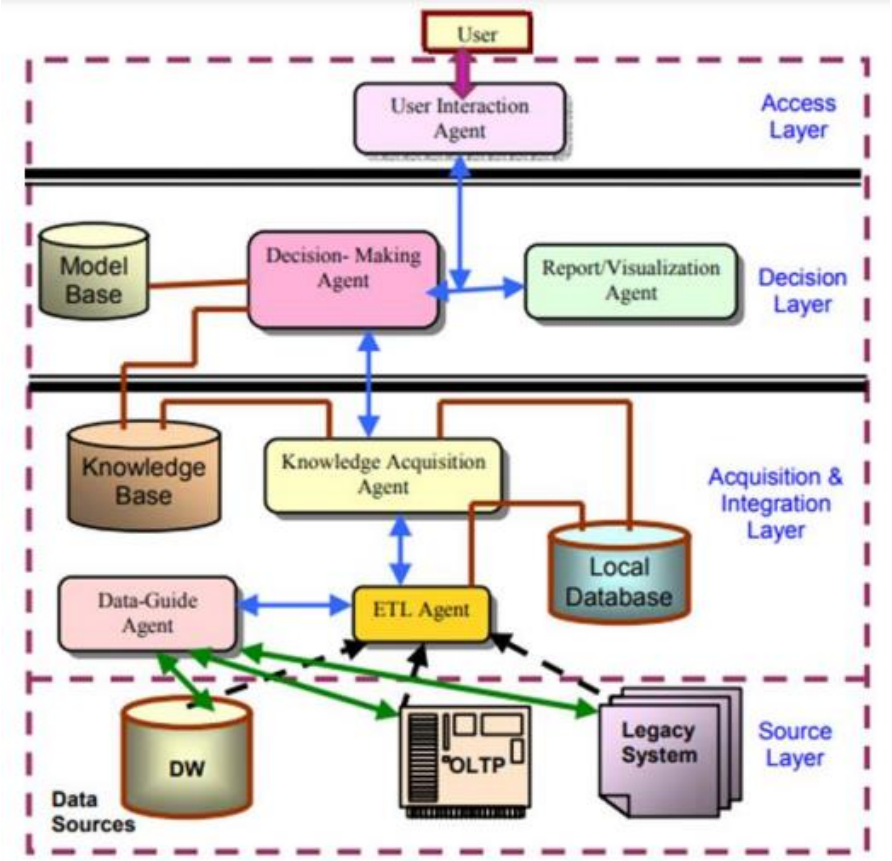

Рисунок 1.5 – Архітектура СППР на основі мультиагентної технології

У запропонованому рішенні сховище даних DW складається із набору залежних вітрин даних.

Діяльність [12] представлена СППР будівельників (забудовників) під час виборів ділянки будівництва житлового будинку.

Концептуальна модель СППР відображена на рис. 1.6.

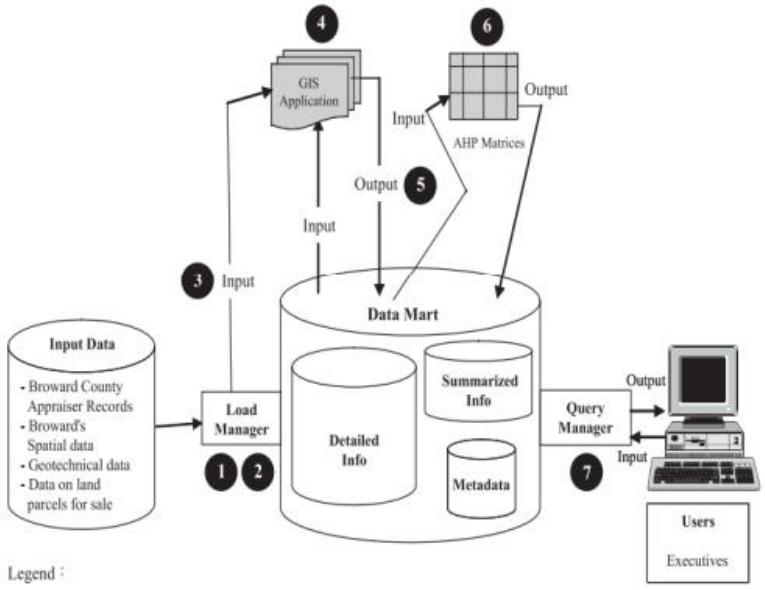

Рисунок 1.6 – Концептуальна модель СППР при виборі ділянки для будівництва житлового будинку

Спеціалізовний застосунок експортує список ділянок та їх макетів у вітрину даних (Data mart) для подальшого аналізу та прийняття рішення.

У рішенні використовується підхід до створення сховища даних, у якому окремі вітрини даних, що призначаються окремим бізнес одиницям або процесам, розробляються і пізніше інтегруються в сховище даних у масштабі підприємства.

Головна перевага цього підходу - відносна простота реалізації.

Слід зазначити, що розглянуті рішення не призначені для аналізу великих масивів даних.

1.3 Моделі СППР на основі бізнес- аналізу великих даних

У теорії BI виділено такі проблеми традиційних BI-рішень у контексті застосування для аналізу великих даних [13]:

– відсутність можливостей аналізу на вимогу: досвідченим користувачам BI сьогодні не потрібно чекати відповідей на складніші бізнес-завдання. Нові користувачі потребують можливостей самообслуговування для зв'язування та аналізу конкретних наборів даних залежно від їхнього власного розуміння, для

будь-яких цілей та у будь-який час;

– необхідний прогнозний аналіз: можливості історичних звітів дозволяють отримати лише одну частину бізнес-аналізу – розуміння того, що відбувалося у минулому. Компанії звертаються до прогнозної аналітики чи розуміння майбутнього, щоб спрогнозувати майбутнє і справді керуватися даними. За допомогою прогнозних моделей компанії можуть використовувати закономірності та прогнози, щоб отримати наступні практичні кроки, використовуючи свої дані;

– аналіз змішаних типів даних: традиційні платформи BI в основному орієнтовані на структуровані дані, але сьогодні користувачам потрібна можливість перегляду та аналізу напівструктурованих, неструктурованих даних та сторонніх даних. В останні роки величезна кількість виробленої інформації збільшилася, частково через нові технології інтелектуального аналізу даних, Інтернету речей (IoT), поширення давачів даних та засобів автоматизованого збору даних. Тепер досвідченим користувачам BI та фахівцям з обробки даних необхідний доступ до невикористаних даних у різних форматах, щоб змішувати типи даних та створювати власні алгоритми, в яких на запит доступні аналітичні дані для прийняття точних і швидких рішень.

Так, у роботі [14] звертається увага на необхідність розробки інструментів, що поєднують функціональну міць BI-систем (а краще - перевершують її) зі здатністю до обробки великих обсягів інформації. Як один із способів отримати такий інструмент пропонується створення логічної вітрини даних.

Структурна схема BI-системи з урахуванням логічної вітрини даних представлена на рис. 1.7.

У запропонованому рішенні вітрина даних не тільки компонує і пов'язує інформацію з різних джерел, а й робить логічні висновки у ній відповідно до заданих правил. Автоматизація отримання логічних висновків – одна з головних практичних переваг семантики.

Рисунок 1.7 – Структурна схема BI-системи на основі логічної вітрини даних

Необхідно наголосити, що одним із найпопулярніших методів побудови аналітичних платформ для великих даних передбачає використання концепції озер даних.

Дані у озері даних можна зберігати у вихідному вигляді без попереднього структурування та запускати різні типи аналітики - від панелей моніторингу та візуалізацій до обробки великих даних, аналітики в реальному часі та машинного навчання для прийняття більш ефективних рішень [15, 16].

Слід врахувати, що озера даних власними силами мало виконують автоматичні операції очищення і передачі, дозволяючи приймати дані ефективніше, але вони перекладають основну відповідальність за підготовку та аналіз даних на бізнес-користувачів.

Як показує практика, підхід на основі озера даних може запропонувати недороге рішення завдяки застосуванню розподіленої файлової системи (як варіант - HDFS) для ефективного зберігання різних типів даних та аналізу їх у вихідній структурі. Водночас озеро даних потребує додаткових ресурсів для реалізації та високого рівня експертизи - по-справжньому отримати з них користь можуть лише висококваліфіковані аналітики.

Для вирішення цієї проблеми використовуються вітрини даних, що являють собою середовища зберігання даних вузькоспеціалізованої інформації,

орієнтовані на запити працівників певного відділу, вектор роботи організації.

Авторами представлено дослідження форматів зберігання великих даних на платформі Apache Hadoop, які використовуються створення вітрин даних.

На рис. 1.8 представлена схема обчислювального експерименту дослідження форматів зберігання великих даних на платформі Apache Hadoop.

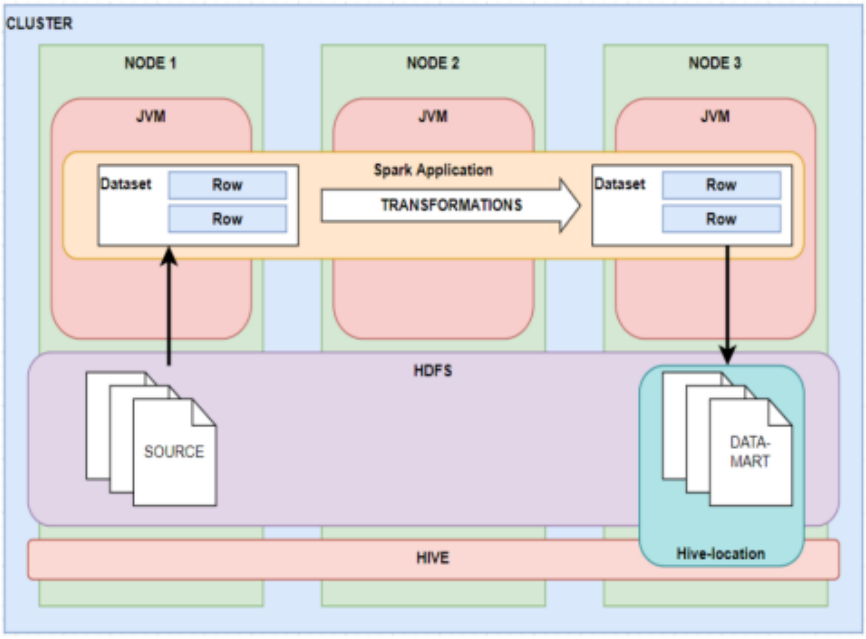

Рисунок 1.8 – Схема обчислювального експерименту для дослідження форматів зберігання великих даних на платформі Apache Hadoop

Як основні критерії ефективності авторами вибрані такі характеристики, як час, необхідний створення вітрини даних, і навіть її обсяг. Експеримент показав, що найефективнішим є формат Apache Parquet.

Разом з тим, необхідно констатувати недостатність робіт, присвячених проблемі розробки BI-систем на основі сховищ і ВВД.

1.4 Висновки до першого розділу

Наведено аналіз сучасного стану досліджень у галузі розробки МПЗ систем прийняття управлінських рішень на основі вітрин даних.

Результати виконаної роботи дозволили зробити такі висновки:

– згідно з концепцією компанії Gartner, одним із ключових компонентів BI -систем є сховище даних;

– вітрина даних - це невелика колекція даних зі сховища даних, яка використовується для бізнес-аналізу в одному підрозділі, тоді як сховище даних є набором всіх даних компанії, котрі застосовуються в цілому для аналізу;

– в деяких роботах автори звертають увагу на необхідність розробки інструментів, що поєднують функціональну міць BI – систем із здатністю до обробки більших обсягів даних. Для цього було запропоновано використовувати різні технології, що базуються на застосуванні вітрин даних.

Разом з тим, проведений аналіз дозволив констатувати недостатність робіт, присвячених проблемі розробки BI на основі сховищ та ВВД, що підтверджує актуальність теми цього дослідження.

2 ТЕОРЕТИЧНА ЧАСТИНА

2.1 Методи побудови вітрин даних

Виділяють такі типи вітрин даних:

– незалежні вітрини даних;

– залежні вітрини даних і оперативне сховище даних;

– логічні вітрини даних.

Розглянемо методи побудови кожного з перелічених вище типів вітрин даних.

2.1.1 Методи побудови залежних вітрин даних

Залежна вітрина даних отримує дані із централізованого сховища даних. Це покращує централізацію даних. Крім того, якщо потрібно розробити одну або кілька фізичних вітрин даних, можна налаштувати їх як залежні [17].

Щоб сформувати сховище даних, певний набір даних агрегується зі сховища даних, реструктурується, а потім завантажується у вітрину даних, де його можна запросити.

Залежна вітрина даних переважно є комбінацією корпоративного (централізованого) сховища даних (ЦСД) і узгоджених даних.

ЦСД (інтегроване) надає кінцевим користувачам інформацію підтримки прийняття рішень, необхідних для втілення.

Залежна вітрина даних та архітектура сховища операційних даних називається підходом «концентратор і промінь», в якому ЦІСД є концентратором, а системи вихідних даних і вітрина даних є вхідними та вихідними периферійними пристроями.

Концепція ЦСД ґрунтується на підході Б. Інмона, згідно з яким сховище даних є централізованим сховищем усіх корпоративних даних.

За такого підходу організація спочатку створює єдине ЦСД. Потім

створюються залежні вітрини розмірних даних з урахуванням моделі сховища.

Це низхідний підхід до побудови сховища даних (рис. 2.1).

Рисунок 2.1 – Архітектура ЦСД

На основі ЦСД побудовано архітектуру LSA [18]. Реально LSA втілює логічне розподілення структур із даними на декілька функціональних шарів. Дані піддаються копіюванню із рівня на рівень, трансформуючись при цьому, щоб врешті-решт стати погодженою інформацію, котра придатна для аналізу вподальшому.

LSA -архітектура сховища даних представлена на рис. 2.2.

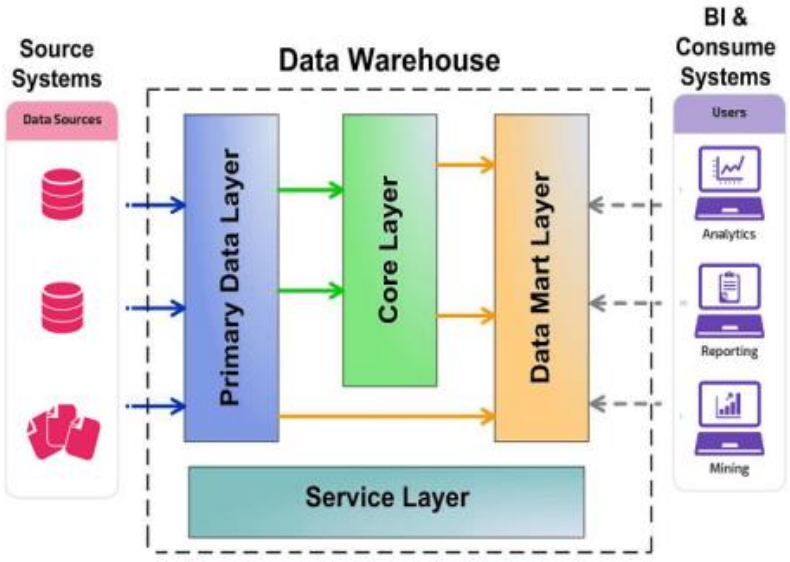

Рисунок 2.2 – LSA -архітектура сховища даних

Стандартно LSA втілюється через такі рівні:

– операційний шар первинних даних, власне на якому і відбувається завантаження інформації і зберігання всієї історії змін. Саме тут і проходить абстрагування чергових шарів сховища від фізичної будови джерел, способів їх збирання і методів визначення змін даних;

– ядро сховища, є центральним компонентом, котрий консолідує дані із різноманітних джерел, зводячи їх до властиво єдиних структур і ключів. Якраз на цьому рівні і проходить основна робота з якістю даних та відбуваються загальні трансформації, з метою абстрагування споживачів від показників логічної будови джерел даних і, власне, потреби їх взаємного співставлення. Таким чином розв'язується завдання дотримання цілісності і якості даних [18];

– аналітичні вітрини, в яких дані зводяться до структур, котрі є зручними для аналізу та застосування у BI –дашбордах або інших системах-споживачів. Якщо вітрини отримують дані із ядра, тоді вони носять назву регулярних. Коли для швидкого вирішення локальних задач не вимагається консолідація даних, тоді вітрина здатна брати первинні дані із операційного шару і вона, відповідно, носить назву операційної. На додачу сніють вторинні вітрини, котрі застосовуються для демонстрації результатів складних обчислень та нетипових змін;

– сервісний шар втілює керування всіма рівнями, які були описані вище. Тут бізнес-дані відсутні, проте використовуються метадані і інші структури для роботи з якістю даних, даючи змогу проводити наскрізний аудит даних, застосовувати загальні критерії для виділення різниці змін і керування завантаженням. Додатково наявні засоби моніторингу і діагностики помилок для пришвидшення вирішення проблем [18].

У LSA -архітектурі реалізовано озеро даних BI -системи для аналізу великих даних у банку (рис. 2.3).

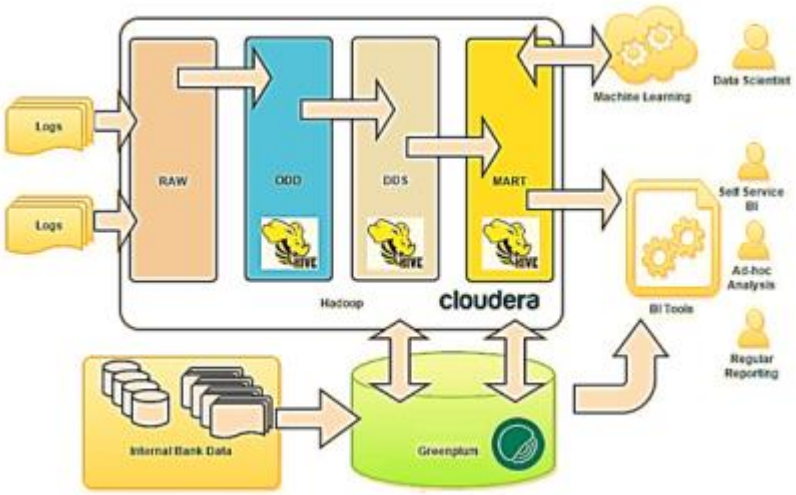

Рисунок 2.3 – LSA-архітектура корпоративного Data Lake в банку

Слід зазначити, що ця технологія широко застосовується у рішеннях корпорації SAP [19].

Залежні вітрини даних містять усі переваги вітрин даних, а також дозволяють здійснювати фізичний контроль даних у міру їх вилучення зі сховища даних.

Оскільки залежні вітрини використовують сховище даних як основу, вони зазвичай вважаються кращим рішенням, ніж незалежні вітрини, але вони займають більше часу на обробку запитів та дорожчі в реалізації.

2.1.2 Методи побудови незалежних вітрин даних

Незалежна вітрина даних – це автономна система, створена без використання сховища даних, яка фокусується на одній предметній галузі або бізнес-функції.

Дані витягуються з внутрішніх або зовнішніх джерел даних, обробляються, потім завантажуються в репозиторій вітрини даних, де зберігаються до того часу, як вони не знадобляться бізнес-аналітики (рис. 2.4).

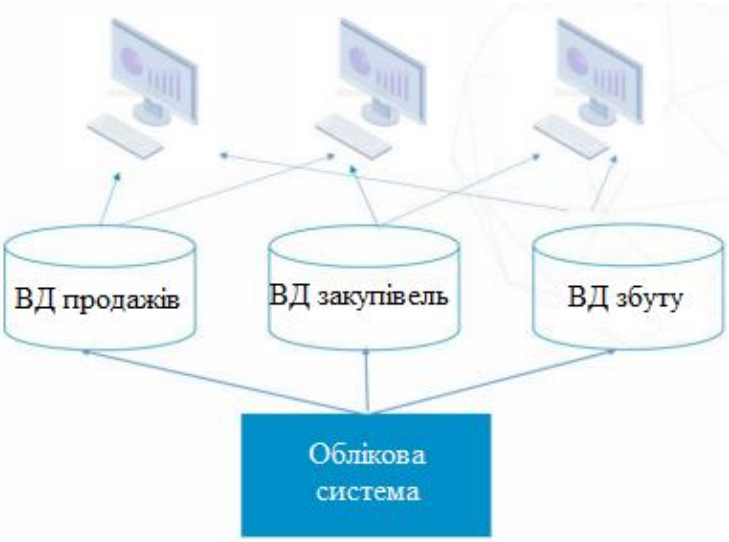

Рисунок 2.4 – Архітектура незалежних вітрин даних

Концепція незалежних вітрин даних належить Р. Кімбаллу. Вона спирається на висхідний підхід до побудови сховищ даних.

При такому підході спочатку створюються вітрини даних, щоб надати звіти та аналітичні можливості для конкретних бізнес- процесів. Потім ці вітрини даних можна інтегрувати для створення комплексного сховища даних. У результаті створюється багатовимірне сховище даних, що починається з критично важливих вітрин даних, які швидко налаштовуються обслуговування аналітичних потреб відділів чи напрямів бізнесу.

При такому підході сховище даних є об'єднанням вітрин даних, однак єдиного джерела достовірної інформації не існує, оскільки дані не інтегруються перед складанням звітів.

Незалежні вітрини даних, як правило, найпростіші і найшвидші в реалізації, і їхня окупність може бути майже миттєвою. Деякі корпорації починають із кількох вітрин даних, перш ніж ухвалити рішення про створення справжнього сховища даних.

Однак цей підхід має кілька проблем:

– хоча незалежні вітрини даних мають очевидну цінність, вони не є справжнім рішенням для всього підприємства і згодом можуть стати дуже дорогими у міру збільшення їх обсягів;

– незалежні вітрини даних не забезпечують історичну глибину справжнього сховища даних.

Оскільки вітрини даних призначені для обробки певних типів запитів від певного типу користувачів, часто не підходять для аналізів типу «що, якщо», як, наприклад, сховища даних.

Незалежні вітрини даних застосовуються, наприклад, у компаніях, що мають розгалужену мережу філій.

2.1.3 Методи побудови логічних вітрин даних

Логічне сховище даних (Logical data warehouse, LDW) - це архітектура управління даними, у якій архітектурний рівень розташовується поверх традиційного сховища даних, забезпечуючи доступ до множини різноманітних джерел даних, надаючи користувачам одне логічне джерело даних.

За своєю суттю, це аналітична архітектура даних, котра оптимізує як традиційні джерела даних (БД, корпоративні сховища даних, озера даних тощо), так і інші джерела даних (додатки, файли великих даних, веб-служби та хмари) задоволення будь-яких аналітичних завдань.

Цей термін було введено в обіг у 2009 році і він продовжує набирати обертів на ринку, оскільки обсяги та складність даних стає проблемою, що зростає, для багатьох компаній [20].

Логічна вітрина даних (Logical data mart, LDM) – це форма залежної вітрини даних, що створюється віртуально з фізичного сховища даних.

Дані видаються користувачам вітрини з використанням ряду уявлень SQL, які справляють враження, що в основі даних, доступних для аналізу, лежить фізична вітрина даних.

Інакше кажучи, логічні вітрини даних - це фізично розділені БД, а різні реляційні уявлення сховища даних. Дані переміщуються до сховища даних, а чи не в окрему проміжну область.

Нову вітрину даних можна створити швидко.

Логічні вітрини даних завжди у актуальному стані.

На думку фахівців компанії Teradata, «можливо, ідеальний підхід до впровадження концепції вітрини даних у сховище даних – це створення однієї чи кількох залежних логічних чи віртуальних вітрин даних» [21].

Архітектура логічної вітрини даних у концепції компанії Teradata показано на рис. 2.5.

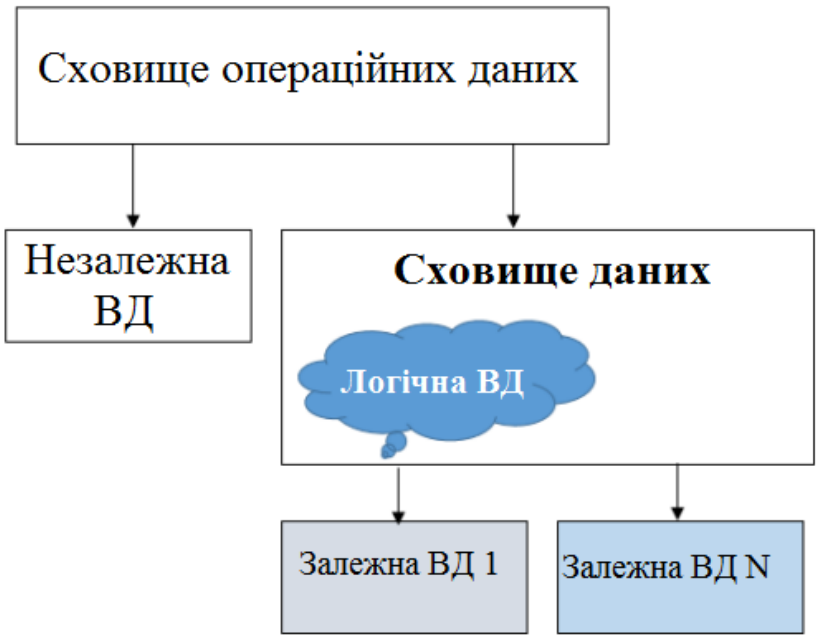

Рисунок 2.5 – Архітектура логічної вітрини даних

Використовуючи систему ретельно сконструйованих уявлень про деталізовані дані сховища, можна створити кілька віртуальних вітрин даних для конкретних користувачів або відділів, які будуть надавати таку ж ретельно підібрану інформацію, як і фізичні вітрини даних, без необхідності великих завантажень даних, очищення та інших необхідних перетворень.

Логічні вітрини даних долають більшість обмежень незалежних вітрин даних. Дані не мають історичної межі, і запити типу «що, якщо» цілком здійсненні.

Головний недолік логічних вітрин даних – відсутність фізичного контролю за даними. Оскільки дані в сховищі не агреговані та не розподілені за розмірами, продуктивність порівняно з логічним магазином зазвичай не така

хороша, як у порівнянні з незалежним магазином. Однак використання паралелізму в логічній вітрині може подолати деякі обмеження неперетворених даних.

Слід також зазначити, що логічні вітрини повинні використовуватись у добре спроектованому сховищі даних.

Для порівняння вищеописаних методів щодо застосування при розробці вітрин великих даних складемо табл. 2.1.

Критерії оцінювання:

0 - повна невідповідність вимогам;

1 - значна невідповідність вимогам;

2 - незначна невідповідність вимогам;

3 - повна відповідність вимогам.

Таблиця 2.1 - Порівняння методів побудови вітрин даних

Характеристики	**Залежні вітрини даних**	**Незалежні вітрини даних**	**Логічні вітрини даних**
Витрати на реалізацію для великих обсягів даних	2	3	3
Надмірність даних	2	4	3
Великі обсяги історичних даних	5	1	4
Швидкість обробки запитів	2	4	3
Разом	11	12	13

Таким чином, при розробці ВВД є доцільним використання методів побудови логічних вітрин даних.

2.2 Технології побудови BI -систем для аналізу великих обсягів даних

Як показав аналіз відомих рішень, при побудові BI- систем для аналізу

великих даних для різних завдань, зазвичай використовується технологія ROLAP – це форма OLAP, яка виконує динамічний багатовимірний аналіз даних, котрі зберігаються в реляційній БД, а не в багатовимірній БД [22].

Принцип роботи такої моделі досить простий: при активізації аналітичної функції формується SQL -запит, який йде на той чи інший бекенд. При BI - система може бути інтегрована з будь-якою СУБД, що підтримує мову SQL (наприклад, реляційні СУБД MySQL, PostgreSQL та інші) [23].

Модель інтеграції BI -системи з реляційною СУБД представлена на рис. 2.6.

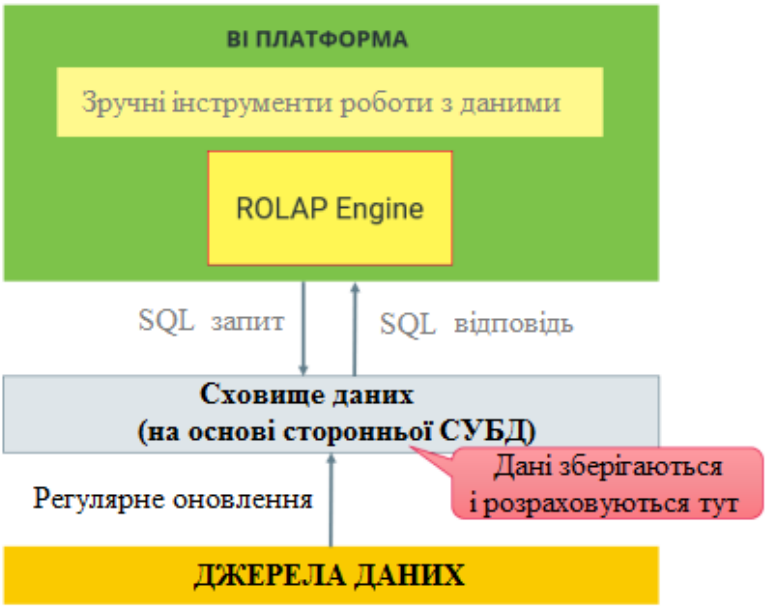

Рисунок 2.6 – Модель інтеграції BI-системи з реляційною СУБД

Перевага даної технології у тому, що для роботи BI- системи немає обмежень за обсягом даних. Недолік - відносно невисока швидкість виконання запитів, що негативно впливає на продуктивність BI -системи.

Для вирішення цієї проблеми використовуються різні підходи, у тому числі застосування спеціалізованих колонкових СУБД, наприклад ClickHouse або Vertica. Однак колонкові СУБД використовують власний зрізаний діалект SQL, що суттєво знижує функціональні можливості такого рішення.

Більш перспективним є рішення, засноване на застосуванні СУБД, в яких

реалізована технологія In-Memory. Цей термін, зазвичай, має на увазі великомасштабні складні обчислення, які вимагають спеціального системного ПЗ для виконання обчислень на комп'ютерах, що працюють разом у кластері. Як кластер комп'ютери об'єднують свою оперативну пам'ять, тому розрахунок по суті виконується між комп'ютерами і використовує загальний простір ОЗП всіх комп'ютерів разом. Обробка в пам'яті усуває всі повільні звернення до даних і покладається виключно дані, що зберігаються в ОЗП [24]. Загальна продуктивність обробки не знижується через затримку, яка зазвичай спостерігається при доступі до жорстких дисків або твердотільних накопичувачів. Обробка в пам'яті часто виконується за допомогою технології відомої як сітки даних у пам'яті.

Такі СУБД мають перевагу у швидкості порівняно з традиційними дисковими СУБД. Багато існуючих реляційних СУБД підтримують механізм In Memory, наприклад, пізні версії MS SQL Server, Oracle Database та інші [23].

Модель BI -системи, що використовує рушій In-Memory OLAP, представлена на рис.2.7.

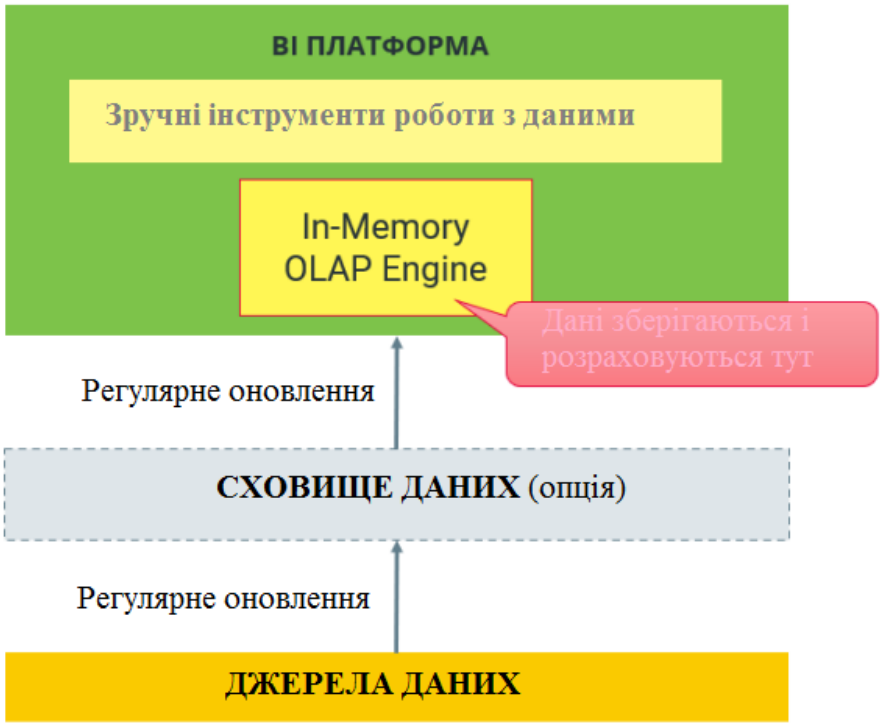

Рисунок 2.7 – Модель BI -системи, що використовує рушій In-Memory OLAP

За результатами представленого аналізу вибираємо як технологію побудови BI -системи для аналізу великих даних технологію, засновану на застосуванні реляційної СУБД, що підтримує механізм In-Memory [23].

2.3 Технологія проектування програмного забезпечення BI- системи

Між життєвим циклом транзакційних систем та життєвим циклом BI-систем є деякі суттєві відмінності, які залежать від характеристик СППР, але для розробки використовуються одні й ті ж традиційні методи та етапи [25]:

Діаграма діяльності процесу проектування ПЗ BI-системи продемонстрована на рис. 2.8.

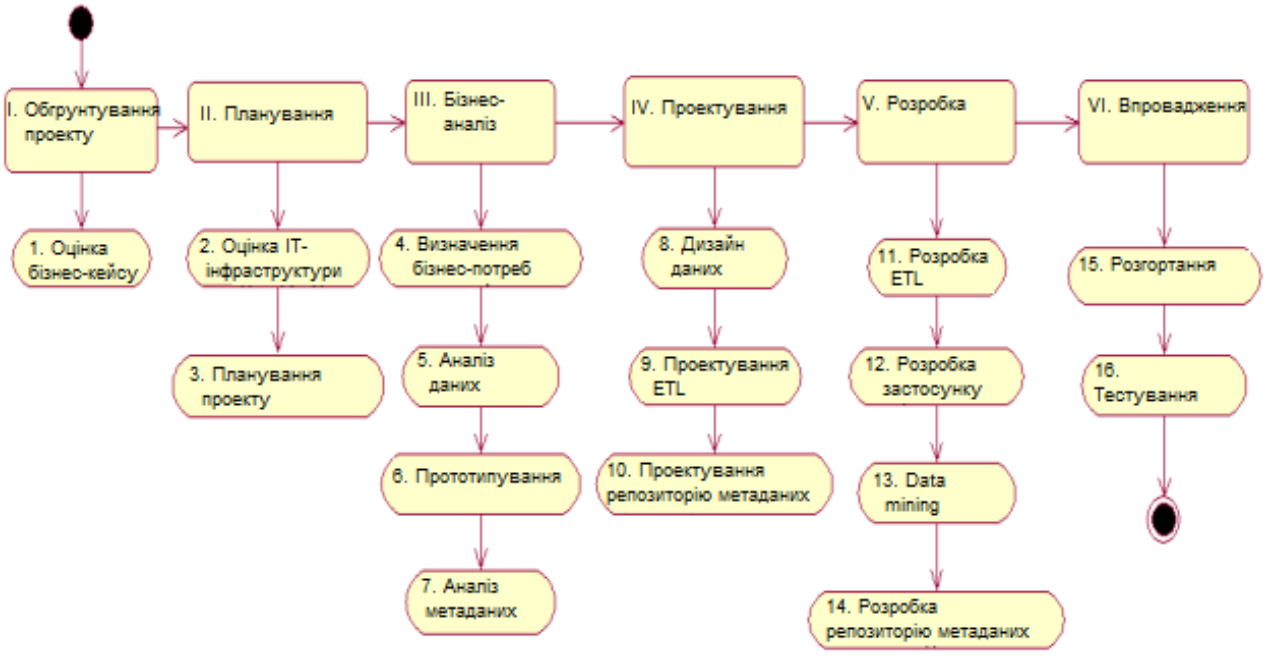

Рисунок 2.8 – Діаграма діяльності процесу проектування ПЗ BI-системи

Процес проектування BI-системи складається із 6 стадій, що включають 16 етапів.

Розглянемо стадії цього процесу.

Стадія I: Обгрунтування. Ця стадія містить такі етапи.

Етап 1: Оцінка бізнес-кейсу. Визначаються бізнес-потреби та можливості організації. Потім команда розробників пропонує початкове рішення,

виправдане з урахуванням витрат та вигод. Складається попередній звіт.

Стадія II: Планування. Нижче наведено етапи цієї стадії.

Етап 2: Оцінка інфраструктури підприємства. На цьому етапі оцінюються можливості організації підтримки та виконання проекту BI-системи з точки зору IT-інфраструктури, компонентів, пристроїв, мережі, а також майбутніх потреб в обладнанні. На цьому етапі будується IT-інфраструктура організації.

Етап 3: Планування проекту. Даний етап включає динамічне планування проекту, яке призводить до швидких змін у технологіях, організаційних та бізнес потребах, людських ресурсах та команді розробників. План проекту докладний, прогресивний, на кожному етапі та кроці є контрольні точки, тестові документи та звіти.

Стадія III: Бізнес-аналіз. Етапи стадії:

Етап 4: Визначення бізнес-потреб. Тут організовуються співбесіди та зустрічі з керівниками та менеджерами, а також визначаються бізнес-потреби та функціональні вимоги.

Етап 5: Аналіз даних. Цей етап включає визначення та проектування джерел даних, розробку докладних діаграм «сутність-зв'язок» (ER-діаграм) з атрибутами і посиланнями між даними. Розробляється логічна модель даних BI -системи.

Етап 6. Створення прототипу програми. Початковий прототип створюється та тестується для відповідності потребам бізнесу. Після тестування результати оцінюються з урахуванням позитивних та негативних аспектів.

Етап 7: Аналіз метаданих. На даному етапі створюються метадані, джерела даних відображаються у структурі метаданих. CABE-засоби використовуються для процесу розробки та відображення.

Стадія IV: Проектування системи, котра містить наступні етапи.

Етап 8: Дизайн даних. На цьому етапі деталізується та уточнюється логічна модель даних, а також проектується фізична модель даних. Модель даних для обробки та зберігання вибирається з наступних варіантів: реляційна, об'єктно-орієнтована та багатовимірна модель.

Етап 9: Проектування процесу ETL. Цей етап є найскладнішим у всьому циклі та залежить від якості джерел даних. Рекомендується, щоб процес був побудований в єдиному середовищі, що об'єднує всі модулі організації, а не окремо для кожного відділу. Правило має бути таким: використовуйте один скоординований процес ETL.

Етап 10: Проектування репозиторію метаданих сховища або вітрини даних. Якщо використовується зумовлене рішення для репозиторію метаданих, то цьому етапі воно налаштовується відповідно до вимогами проекту, інакше репозиторій метаданих розробляється в термінах логічної моделі метаданих залежно від моделі даних: реляційної, об'єктно-орієнтований чи багатовимірний [25]. Також є важливим та складним етапом.

Стадія V: Розробка. Ця стадія складається із таких етапів.

Етап 11: Розробка ETL. Для побудови ETL використовуються інструменти, процедури та оператори фільтрації. Фільтрування та перетворення даних залежить від якості джерел даних. Ці джерела є різними: файли, БД, електронна пошта, Інтернет, нетрадиційні джерела.

Етап 12: Розробка програми. Після перевірки прототипу створення остаточної програми може бути простим процесом. Перебудовуються шаблони процедур та інтерфейси. Надаються права та привілеї користувачам.

Етап 13: Data mining. Виконавчі системи повинні реалізувати можливості Data mining, щоб досягти успіху та виконати вимоги менеджера. Цей етап включає перевірку алгоритмів, методів Data mining, таких як кластеризація, методи прогнозування та організації.

Етап 14: Розробка репозиторію метаданих. Якщо репозиторій метаданих має бути створений, то розробляються словник метаданих та доступ до даних через інтерфейси.

Стадія VI: Використання системи, яка складається з етапів.

Етап 15: Розгортання системи. Це процес доставки, під час якого команда розробників організує тренінги для менеджерів, готує остаточну документацію та технічну підтримку, завершує процес завантаження даних та налаштування

програми.

Етап 16: Тестування системи. Після впровадження системи робляться попередні висновки та оцінюються витрати. Команда розробників складає остаточний звіт, в якому описуються характеристики системи, а також деякі частини, які потрібно покращити або перебудувати.

2.4 Методології проектування програмного забезпечення BI-системи

2.4.1 Методологія об'єктно-орієнтованого аналізу та проектування

Як показав аналіз, при проектуванні BI-систем широко застосовується методологія об'єктно-орієнтованого аналізу та проектування, заснована на принципах об'єктно-орієнтованого підходу (ООП) та мові візуального моделювання UML.

Ця методологія використовується на стадії логічного проектування BI-систем.

Логічне проектування визначає функції та особливості інформаційної системи та відносини між її компонентами. Таке проектування включає вихідні дані, які мають бути зроблені ІС, вхідні дані, необхідні для системи, та процес, який має бути зроблений системою, незалежно від того, як завдання виконуватимуться фізично. Логічне проектування визначає те, що має відбуватися в ІС, а не те, як це має бути виконане. Інакше кажучи, логічні проекти не стосуються реальних методів реалізації. При підготовці логічного проекту ІС системний аналітик визначає потреби користувача на рівні деталізації, який фактично визначає потік інформації в систему та з неї та необхідні джерела даних. Результатом логічного проектування є логічна модель та програмна архітектура ІС.

Логічне моделювання – це обов'язковий процес перед написанням програмного коду. Таке моделювання є невід'ємною частиною великих програмних проектів, а також корисним для середніх і навіть невеликих

проектів.

Поняття логічної моделі ІС тісно пов'язане з її логічною архітектурою. Слід додати, що з основних завдань логічного моделювання є розробка логічної моделі даних ІС.

У процесі логічного моделювання широко використовуються патерни (шаблони) проектування. Одним з таких шаблонів є патерн "Модель-Подання-Контролер" (Model-View-Controller, MVC) [26]. MVC здебільшого застосовується для інтерактивних програмних систем. Компонент Model містить основні функції та дані, компонент View надає інформацію користувачеві. Компонент контролера обробляє введення даних за допомогою перевірки. View і Controller спільно є користувацьким інтерфейсом. Механізм розповсюдження змін надає узгодженість між, властиво, інтерфейсом користувача та моделлю. Архітектура патерну MVC представлена на рис. 2.9.

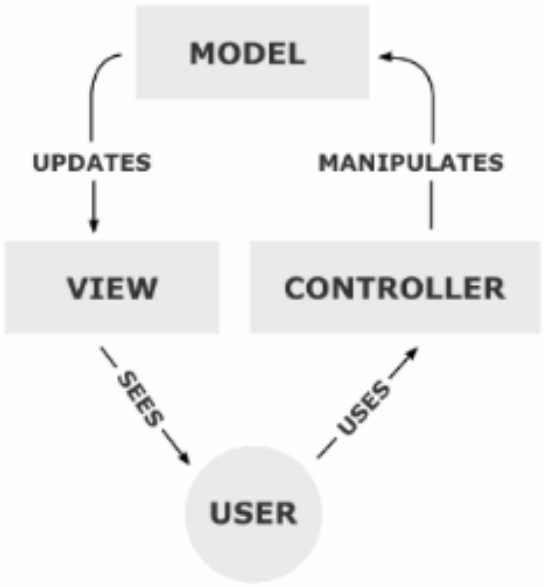

Рисунок 2.9 – Архітектура патерну MVC

На думку авторів роботи [27], застосування MVC при розробці ПЗ BI-системи забезпечить такі переваги:

– ефективність. Системи BI, які використовують MVC, дуже популярні та ефективні з точки зору зберігання різнорідних даних у сховищі даних та синхронізації уявлень;

– простота. Поділ завдання на незалежні компоненти, а саме на модель, представлення та контролер, знижує складність архітектури BI-системи;

– масштабованість. Бізнес-програми стають добре масштабованими, так що нові компоненти бізнес-додатків можна додати чи видалити без шкоди для функціональності BI-системи;

– гетерогенність. BI1-системи неоднорідні, тому можуть працювати у різних обчислювальних середовищах;

– підтримка. BI-системи підтримують синхронізовані уявлення та узгодженість даних, коли вони реалізовані з використанням архітектурного патерну MVC [27].

Оскільки архітектура MVC відокремлює модель від представлення, для однієї моделі можна створити кілька представлень.

При реалізації архітектури BI для будь-якої організації потрібна наявність кількох уявлень для представлення даних у бажаній формі (наприклад, у вигляді графіків, кругових діаграм тощо).

Різним типам зацікавлених сторін (менеджерам, аналітикам, розробникам) можуть знадобитися різні подання даних у різних форматах.

Архітектура MVC допомагає створювати кілька уявлень, зберігаючи незмінними модель та контролер.

2.4.2 Методологія об'єктно-структурного моделювання

Якщо розглядати системи BI як різновид систем управління процесом навчання, то при виборі методологічного підходу до їх побудови слід використовувати методологію, засновану на інтеграції різних методологічних підходів.

Однією з таких методологій є методологія об'єктно-структурного підходу.

У роботі [9] пропонується методологія об'єктно-структурного підходу до побудови ІС управлінського обліку, організованих за моделлю Workflow, що складається з наступних стадій.

1. Об'єктно-структурне моделювання. Об'єктно-структурна модель (ОСМ)

системи N -етапної обробки облікової інформації має вигляд орієнтованого графа O(W,S,D), зображеного на рис. 2.10.

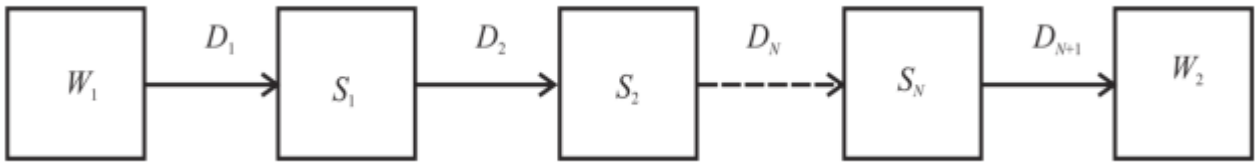

Рисунок 2.10 – ОСМ системи N-етапної обробки облікової інформації

Використано такі позначення: W = {W_1, W_2} - вузли, що позначають об'єкти класу "Склад"; S = {S_1, S_2 ,..., S_N } - множина вузлів, що позначають об'єкти класу "Етап (Склад + Агрегат + Контролер)"; D = {D_1, D_2 ,..., D_{N+1} } - множина дуг, навантажених елементами інформаційного потоку

Фізичним уявленням ОСМ є бізнес-транзакція.

2. Формалізація елементів ОСМ. Полягає у поданні ОСМ як системи взаємодіючих автоматів, управляючих статусом оброблюваного елемента матеріального (інформаційного) потоку, який описується як кінцевий автомат (КА) відповідно до його життєвого циклу, представленого у табл. 2.2 [9].

Таблиця 2.2 - Життєвий цикл продукту у багатоетапному виробництві

Статус	Опис
1	Сировина
2	Напівфабрикат
3	Готова продукція

3. Розробка UML патернів проектування. Паттерни проектування будуються в нотації UML на основі об'єктних моделей кінцевого автомата, згрупованих за суперкласами – класами технологічної онтології конкретної предметної галузі.

Приклади патернів проектування представлені на рис. 2.11.

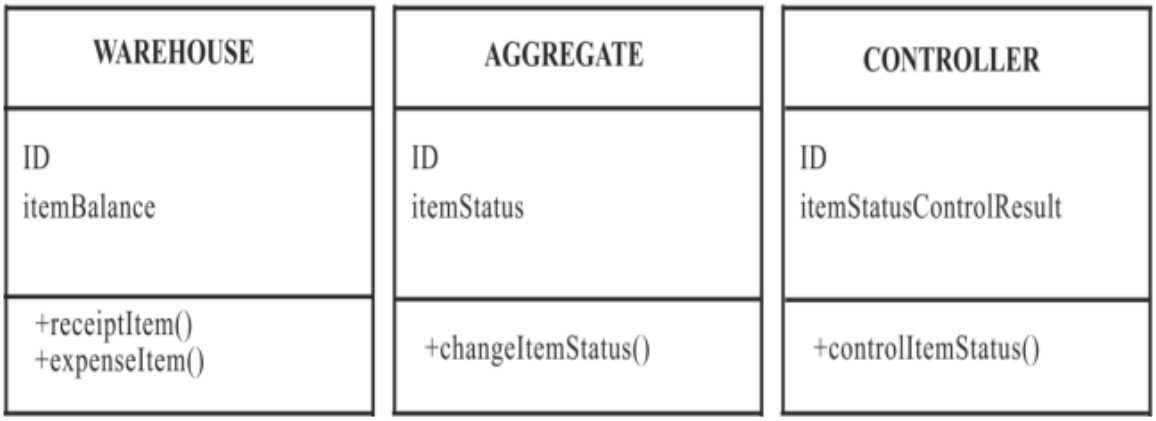

Рисунок 2.11 – Паттерни проектування

Перевагами запропонованої методології проектування є:

– універсальність об'єктно-структурних моделей;

– простота адаптації ОСМ до специфіки управлінського обліку у конкретній організації;

– простота інтеграції до корпоративної інформаційної системи (КІС) підприємства.

Для зберігання агрегованих даних та результатів аналізу інформації (наприклад, страхових резервів, збитковості клієнтів чи фінансових результатів агентів) створюються таблиці фактів, які пов'язуються за схемою «зірка» чи «сніжинка» з таблицями-класифікаторами чи довідниками БД ІС.

Приклад діаграми класів типової підсистеми аналізу страхової інформації наведено на рис. 2.12.

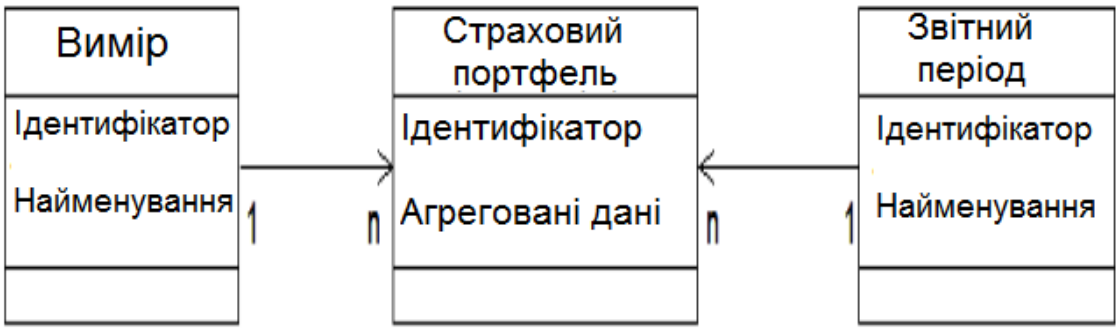

Рисунок 2.12 – Діаграма класів типової підсистеми аналізу страхової інформації

Ключовим об'єктом представленої діаграми є об'єкт класу «Страховий

портфель», що є репозиторієм даних.

На основі відповідного шаблону проектування розробляються програмні об'єкти - інформаційні експерти, які призначені для акумулювання та розрахунку аналітичних даних у СППР [28].

Застосування цього шаблону під час проектування вітрин даних забезпечить суттєве зниження витрат.

2.5 Висновки до другого розділу

У цьому розділі подано аналіз методологій та технологій побудови СППР на основі ВВД.

Результати виконаної роботи дозволили зробити такі висновки:

– як показав аналіз, для розробки ВВД є доцільним використання методів побудови логічних вітрин даних;

– найбільш перспективною для побудови BI -систем для аналізування великих даних вважається технологія, яка ґрунтується на використанні реляційної СУБД, котра працює з механізмом In Memory OLAP;

– технологія проектування BI -системи складається з 6 стадій, що включають 16 етапів. Найбільш трудомісткими вважаються етапи проектування ETL та репозиторію метаданих сховища або вітрини даних.

Застосування шаблону "Репозиторій даних" об'єктно-структурного підходу при проектуванні вітрин даних забезпечить суттєве зниження витрат.

3 ПРАКТИЧНА ЧАСТИНА

3.1 Математичне забезпечення СППР на базі ВВД

3.1.1 Логічне моделювання СППР на базі ВВД

Для розробки математичного забезпечення СППР варто використати два аспекти проектування ІС: ООП і об'єктно-структурний.

Для створення логічної моделі ІС використовується ООП методологія

Варто стверджувати, що при втіленні проектів розроблення МПЗ на базі ООП самі кращі результати отримано за використання методології RUP і CASE -засобу Rational Rose. Основною метою RUP є написання ПЗ високої якості із прогнозованим кошторисом і часовими межами.

Необхідно зауважити, що із сформованої логічної моделі ІВ потрібно створити базові діаграми мови UML, котрі відображають різноманітні аспекти ІС.

З метою встановлення функціональних вимог треба використати UML -діаграму варіантів використання [30]. Вона дає змогу описувати складні процеси простими способами, робить їх абсолютно зрозумілими для усіх осіб. Така діаграма наочно відображає взаємодію між головними сервісами, властиво бізнес-прецедентами, котрі надає власне досліджуваний бізнес-процес, і тими особами, для кого ці сервіси представлені (властиво бізнес-суб'єкти чи актори).

На етапі керування вимогами RUP потрібно, щоб всі прецеденти і особи були чітко встановлені, було створено більшість описів наявних прецедентів.

Отже, акторами BI -системи будуть ETL, Джерело великих даних, Особа, що приймає рішення (ОПР), Інструмент BI, OLAP -система.

Варіанти використання (прецеденти) представлені у табл. 3.1 – 3.4.

Таблиця 3.1 – Опис прецеденту: Вилучення, трансформація та завантаження даних

Прецедент: Вилучення, трансформація та завантаження даних
ID: 1
«Короткий опис: Вилучення, трансформація та завантаження даних у сховище даних
Головний актор: ETL
Другий актор: Джерело великих даних
Передумова: ні
Основний потік: ETL отримує дані з Джерела великих даних, перетворює їх і завантажує в сховище даних »
Післяумова: ні
Альтернативні потоки: ні

Таблиця 3.2 – Опис прецеденту: Створення логічної вітрини даних

Прецедент: Створення логічної вітрини даних
ID: 2
«Короткий опис: створення логічної вітрини даних
Головний актор: OLAP-система
Другий актор: ні
Передумова: ні
Основний потік: OLAP-система формує логічної вітрини даних із сховища даних
Післяумова: ні
Альтернативні потоки: ні

Таблиця 3.3 – Опис прецеденту: Аналіз даних

Прецедент: Аналіз даних
ID: 3
«Короткий опис: Виконання аналізу даних
Головний актор: Інструмент BI
Другий актор: ні
Передумова: створення логічної вітрини даних
Основний потік: Інструмент BI виконує аналіз даних
Післяумова: подання результатів аналізу
Альтернативні потоки: ні

Таблиця 3.4 – Опис прецеденту: Прийняття управлінського рішення

Прецедент: Прийняття управлінського рішення
ID: 4
«Короткий опис: ухвалення управлінського рішення
Головний актор: ОПР
Другий актор: Інструмент BI
Передумова: представлення результатів аналізу
Основний потік: ОПР приймає управлінське рішення на основі результатів аналізу
Післяумова: ні
Альтернативні потоки: ні

Діаграма варіантів використання властиво BI-системи на базі власне логічної вітрини даних наведена на рис. 3.1.

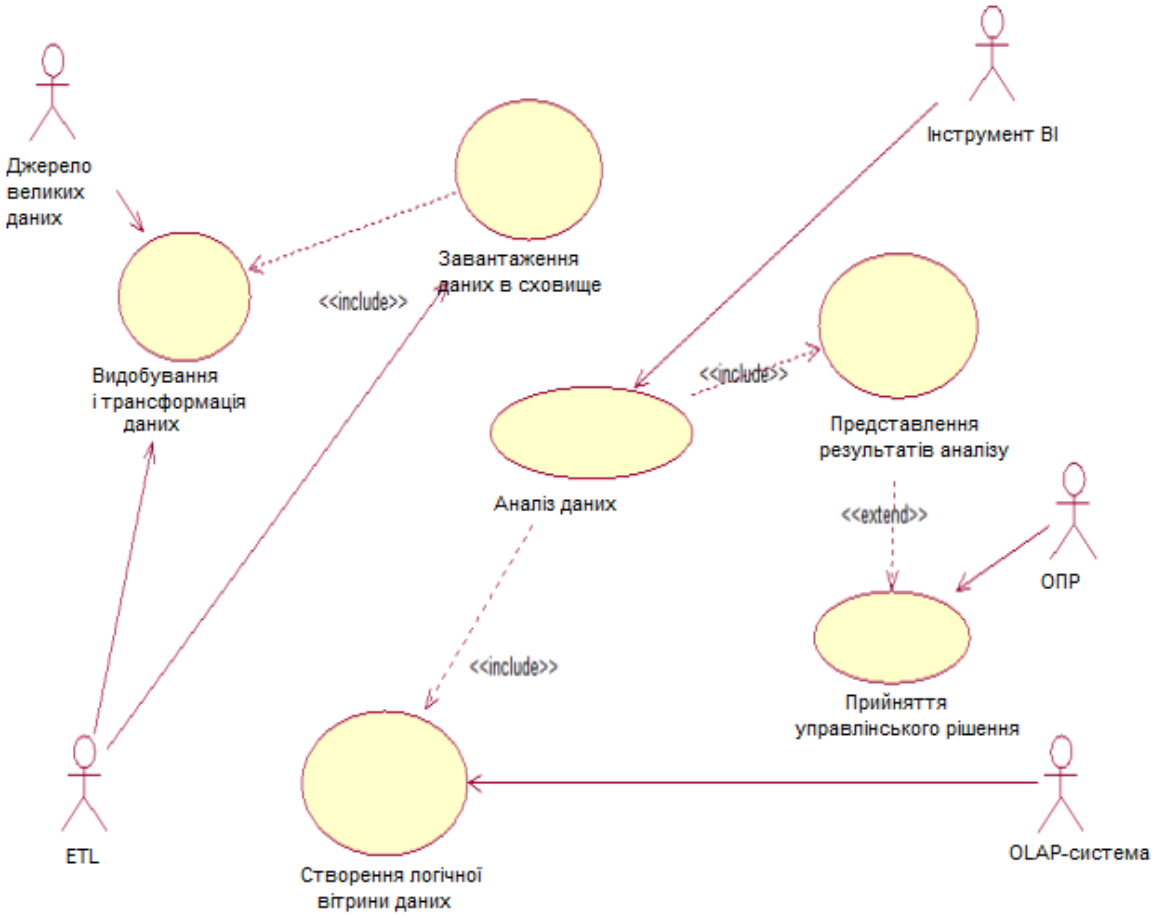

Рисунок 3.1 – Діаграма варіантів використання BI-системи

Розглянемо процес розробки метамоделі об'єктів BI-системи.

Метамодель - це модель, котра описує іншу якусь модель або транзитивне відношення між двома моделями. Для побудови метамоделей варто застосувати підхід MDA [31].

Виділяються три основні категорії об'єктів BI-системи [32]:

– репозиторії даних, що становлять елементи архітектури сховища даних;

– об'єкти даних, що становлять модель даних певного репозиторію;

– об'єкти-презентації, наприклад, статичний звіт чи дашборд.

Для розробки метамоделі об'єктів BI-системи використовуємо діаграми класів. Діаграми класів описують внутрішню структуру класів та відносини між класами. Сюди входять відносини класів, а також атрибути та поведінки, пов'язані з кожним класом. Діаграми класів чудово ілюструють успадкування та складові відносини [33].

На діаграмі класів UML представлені:

– класи в об'єктно-орієнтованій системі;

– атрибути та методи класів;

– зв'язок між класами.

Не представлені на діаграмі класів UML:

– деталі того, як класи взаємодіють один з одним;

– алгоритмічні деталі, тобто як конкретна поведінка класу реалізована.

Діаграми класів відмінно підходять для:

– виявлення пов'язаних даних та атрибутів;

– отримання швидкого уявлення про важливі об'єкти у системі;

– з'ясування достатності чи недостатності класів;

– визначення, чи є відносини між об'єктами надто складними чи простими;

– виявлення залежностей між класами чи об'єктами.

Разом з тим діаграми класів не дозволяють вирішити такі питання, як виявлення алгоритмічної (не керованої даними) поведінки та розуміння загального потоку управління додатком [33].

Метамодель BI-системи на основі логічної вітрини даних – рис. 3.2.

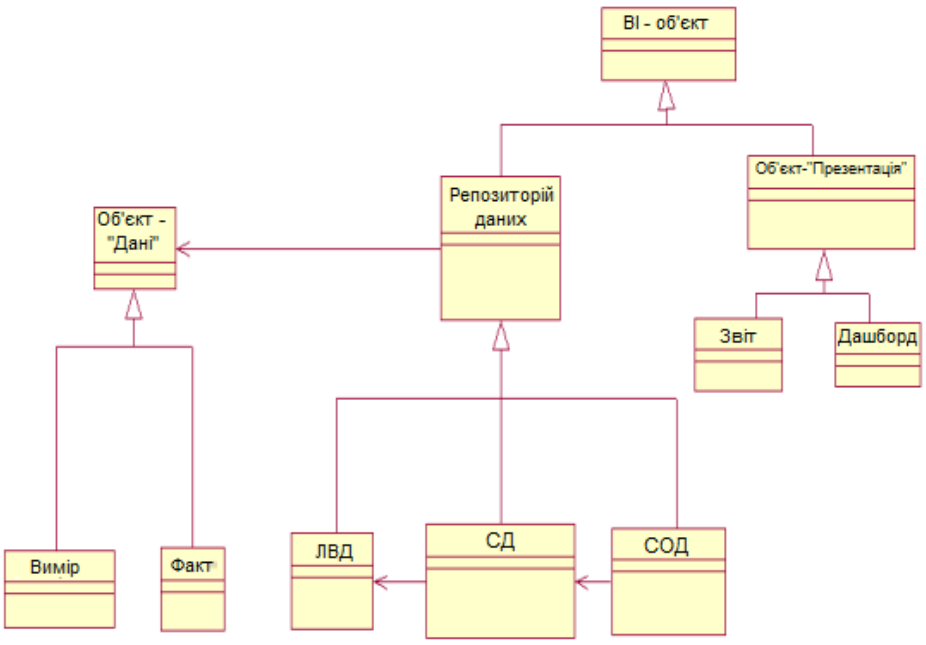

Рисунок 3.2 – Метамодель BI-системи на основі логічної вітрини даних

Тут: СД – сховище даних; СОД - сховище операційних даних; ЛВД - логічна вітрина даних.

Як випливає з діаграми, об'єкти BI, вибрані для моделі, залежать від цільової аудиторії та рівня деталізації моделі.

В оглядовій моделі бізнес-процесу, що підходить для менеджерів, можна показати СД або окремі вітрини даних загалом. У більш докладній метамоделі для розробників підпроцеси можна описати як доступ до окремих сутностей та фактів. Крім того, ОПР, часто отримують відповідні дані у формі звітів, наприклад, звіту про продаж за минулий фінансовий рік, який також може стосуватися моделювання бізнес-процесів.

3.1.2 Розробка логічної моделі даних СППР на базі ВВД

Моделювання даних аналізує об'єкти даних та визначає відносини між ними. Воно генерує теоретичне представлення об'єктів даних - постачальників чи клієнтів у БД і навіть те, як зберігати об'єкти у системі, визначаючи правила відносин між таблицями. Моделювання даних визначає правила перетворення даних та готує дані для аналізу великих даних.

ETL-процес застосовує ці правила та перевіряє їх на наявність аномалій. Це гарантує, що дані готові до аналізу. Потім він завантажує ці дані у сховище даних.

В епоху великих даних для точного аналізу даних аналітики повинні розуміти взаємозв'язок між частинами інформації та тим, як дані передаються в системі.

Отже, моделювання даних BI-системи має вирішальне значення з метою агрегування власне даних між різнорідними, властиво, інструментами і платформами.

Потрібно розглянути процес моделювання даних логічної вітрини даних.

Вона створена на основі ROLAP-моделі, із використанням схеми «сніжинка» [34].

Для побудови таблиці фактів застосовано шаблон «Репозиторій даних».

Власне репозиторій даних описується за допомогою кортежу:

$$SP = <GP, XP, YP, ZP, zp_0, vp, fp> , \quad (3.1)$$

де GP – потік управління; XP, YP - вхідний та вихідний потоки строго структурованих та впорядкованих послідовностей значень атрибутів даних; ZP - множина станів репозиторію; zp_0 - початковий стан репозиторію; $vp \in VP$ – оператор переходів репозиторію; $fp \in FP$ – оператор виходів репозиторію.

Зовнішні події, що викликають зміну стану репозиторію ZP, ініціюються рухами потоків XP (додавання даних) та YP (видалення даних). Внутрішні події репозиторію ініціюються керуючим потоком GP (наприклад, зміна умов страхування чи кредитування, припинення дії договору тощо) [28].

Приклад логічної моделі даних віртуальної вітрини даних кредитної організації, створеної з допомогою представлення даних (View) у сховищі великих даних, показано на рис. 3.3.

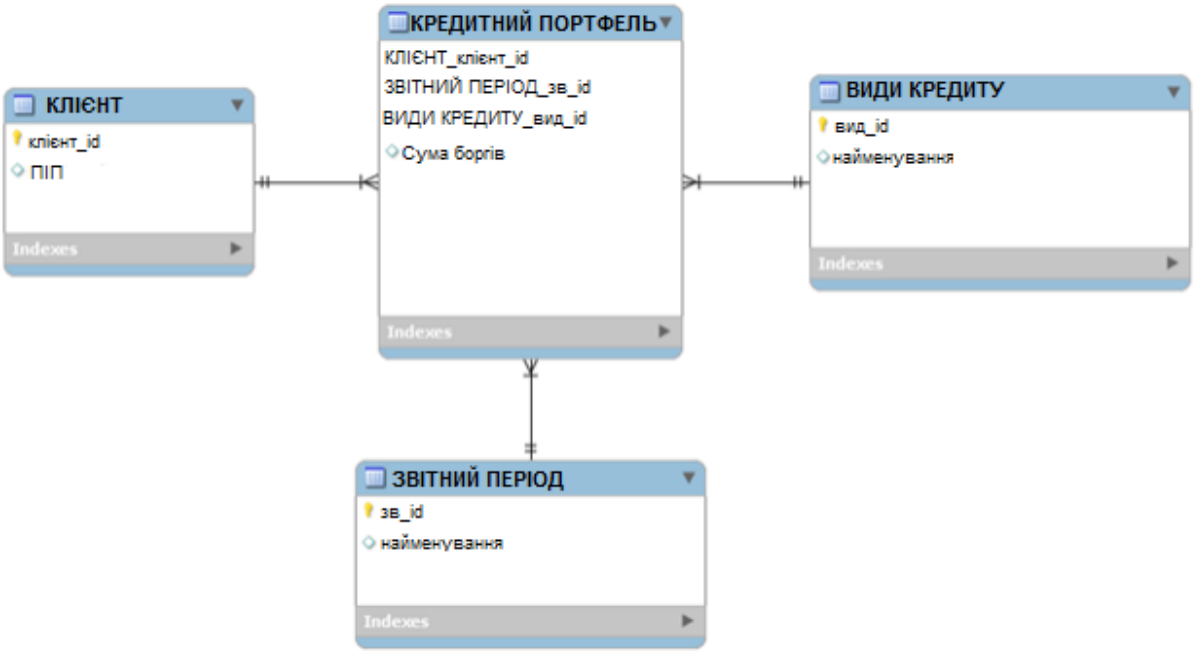

Рисунок 3.3 – Логічна модель даних логічної вітрини даних кредитної організації

Таблиця фактів моделі "Кредитний портфель" створена на основі вищеописаного шаблону "Репозиторій даних".

3.2 Програмне забезпечення СППР на базі ВВД

Для розробки програмної архітектури СППР використовуємо діаграму компонентів UML. Діаграми компонентів використовуються для візуалізації організації та відносин між компонентами у системі. Ці діаграми також використовуються для створення систем, що виконуються.

Діаграми компонентів можна описати як статичне представлення системи. Статична реалізація є організацією компонентів у певний момент.

Діаграма одного компонента неспроможна уявити всю систему, але набір діаграм використовується для представлення цілого [29].

На рис. 3.4 наведена програмна архітектура СППР на основі логічної (віртуальної) вітрини даних.

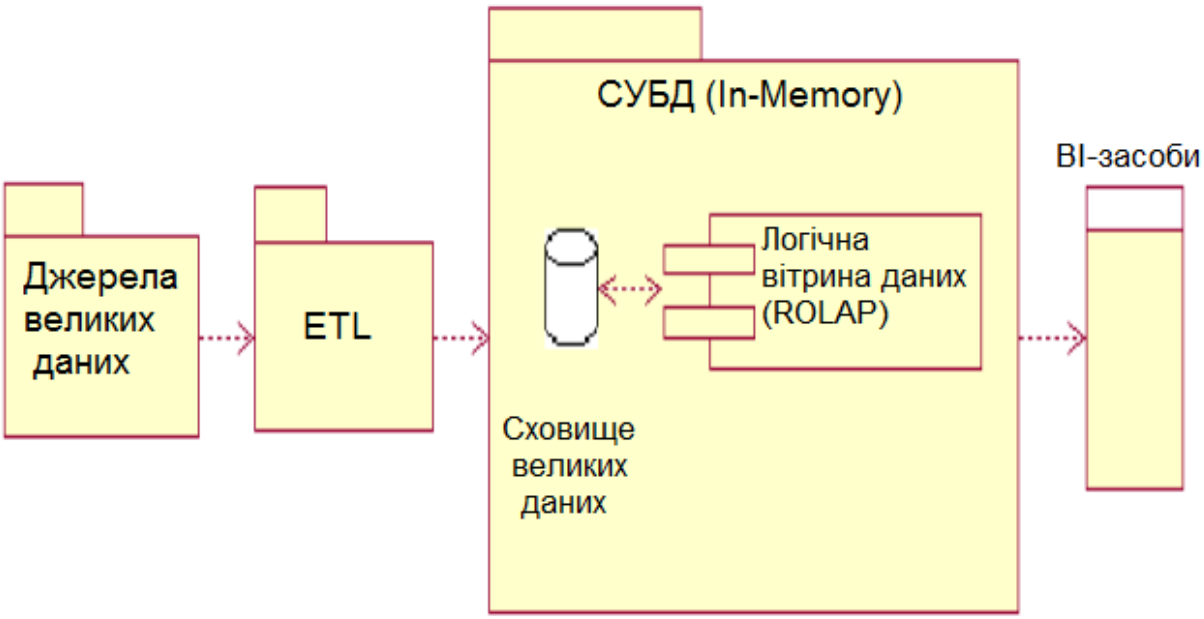

Рисунок 3.4 – Програмна архітектура СППР на базі логічної вітрини даних

Зараз загальновизнаним є підхід до проектування ПЗ BI -систем, котрий ґрунтується на використанні готових IT-рішень.

Основними перевагами цього підходу є [35]:

– менші витрати. Існує множина готових BI- систем для аналізу великих даних, найбільш популярними з яких сьогодні вважаються MS Power BI, Qlik і Tableau, а також різні безкоштовні і комерційні аналоги в тому числі комплексні DaaS- сервіси. Їх можна використовувати як локально, так і у хмарному варіанті за підписною моделлю;

– висока продуктивність. Готові рішення призначені для масового використання, вони містять набори алгоритмів для обробки та аналізу великих даних, які найбільш затребувані на практиці;

– простота розуміння. Вбудовані модулі візуалізації готових BI- та DaaS -рішень наочно демонструють найбільш значущі для бізнесу показники, такі як кількість відвідувань, конверсія, рівень продажів, сукупні витрати та доходи, а також у розрізі окремих статей. Інтерфейс зрозумілий не тільки IT-фахівцю, а й керівнику та різним фахівцям;

– практичність. BI -системи і DaaS -рішення для аналітики великих даних не тільки наочно відображають найважливіші бізнес-показники, але і автоматизують безліч процесів бек-офісу, необхідних для створення зрозумілого уявлення. При цьому більшість готових рішень розширюються, надаючи API або візуальні редактори для створення власних функцій, скриптів та інших унікальних налаштувань;

– доступність. Крім того, що готові BI- і DaaS -рішення не вимагають вкладення суттєвих інвестицій, для їх використання необов'язкова наявність команди Data -професіоналів (інженер, аналітик, архітектор, розробник, Data Scientist), а також повноцінної Big Data інфраструктури.

Таким чином, впровадження BI/DaaS можна розглядати як початковий етап цифровізації без глибокого реінжинірингу корпоративної діяльності, який реалізує так звану еволюційну стратегію поступово поліпшення.

Розглянемо і порівняємо найбільш популярні рішення BI -систем щодо застосування як СППР на основі ВВД.

3.2.1 BI -система MS Power BI

Це набір програмних служб, застосунків і сполучників, котрі співдіють між собою з метою перетворення розрізнених джерела даних на власне узгоджені, які занурюють користувачів у віртуальний світ і інтерактивні аналітичні дані. Користувацькі дані у Power BI здатні представлятися таблицями Excel чи колекціями хмарних або локальних гібридних сховищ даних.

Інструмент дає змогу легко під'єднюватися до джерел даних, здійснювати візуалізацію і виділяти найбільш важливу інформацію та ділити її між визначеними чи усіма користувачами.

Рішення складається з кількох елементів, які працюють разом:

– класична програма Windows - Power BI Desktop;
– веб-служба SaaS (ПЗ як послуга) – служба Power BI;
– мобільні програми Power BI для Windows, iOS та Android» [36].

На рис. 3.5 представлена функціональна архітектура системи Power BI.

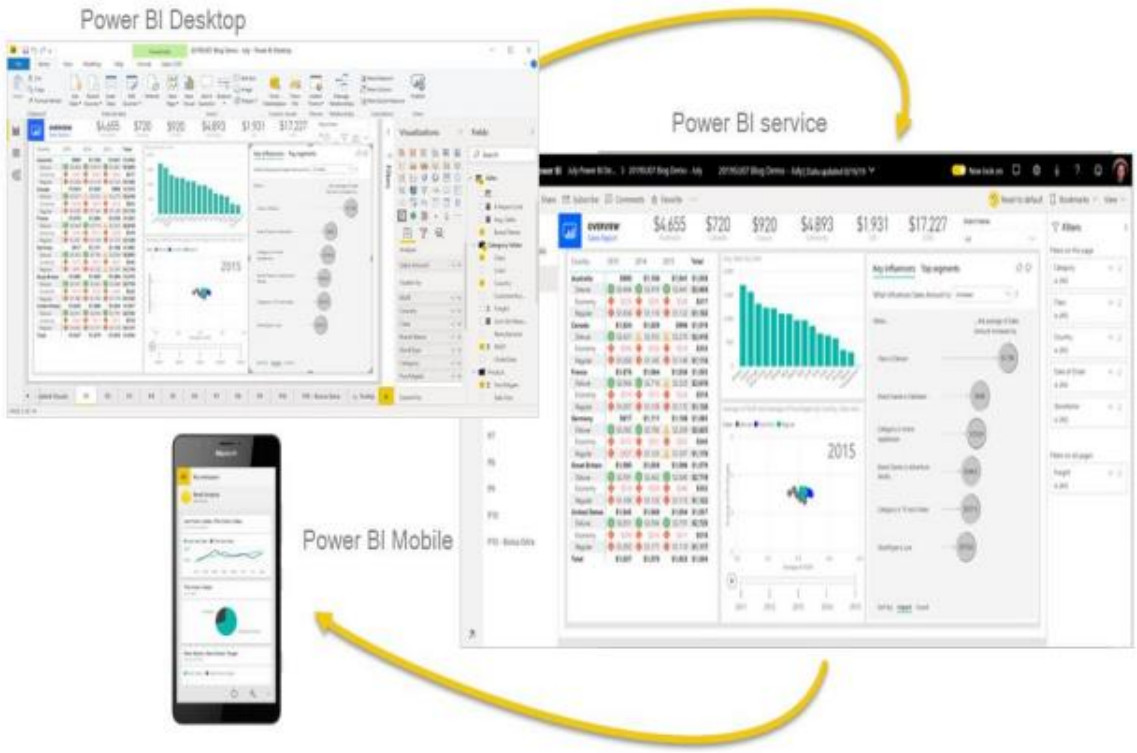

Рисунок 3.5 – Функціональна архітектура системи Power BI

На думку вендора, основними перевагами Power BI є:

– простота під’єднання, моделювання та візуалізації даних, можливість створювати звіти, які запам'ятовуються, персоналізовані з урахуванням ключових показників ефективності та бренду компанії;

– отримання максимальної віддачі від інвестицій у великі дані, під’єднавшись до всіх джерел даних за допомогою шкали для аналізу, обміну та поширення інформації організації, зберігаючи при цьому точність, узгодженість та безпеку даних.

– простота інтеграції з програмами Microsoft Office, такими як Microsoft Teams та Excel.

Power BI дозволяє отримати швидкі відповіді на питання, пов'язані з бізнесом клієнта за допомогою штучного інтелекту.

3.2.2 BI -система Tableu

Є найширшою і найглибшою платформою для наскрізних даних та аналітики. Tableau забезпечує відповідальне використання даних та покращує бізнес-результати за допомогою повністю інтегрованого управління даними, візуальної аналітики та оповідання даних, а також спільної роботи – і все це за допомогою провідного в галузі рішення Einstein від Salesforce [37].

Tableau застосовує природну людську властивість швидко визначати візуальні закономірності (рис. 3.6).

Tableau пропонує широкий вибір і гнучкість, щоб продовжувати рости разом з компанією-власником у міру розвитку стратегії компанії в галузі технологій і даних. Система дозволяє під’єднюватися до даних локально або у хмарі – як до великих даних, БД SQL, електронних таблиць чи хмар, так і до Google Analytics та Salesforce.

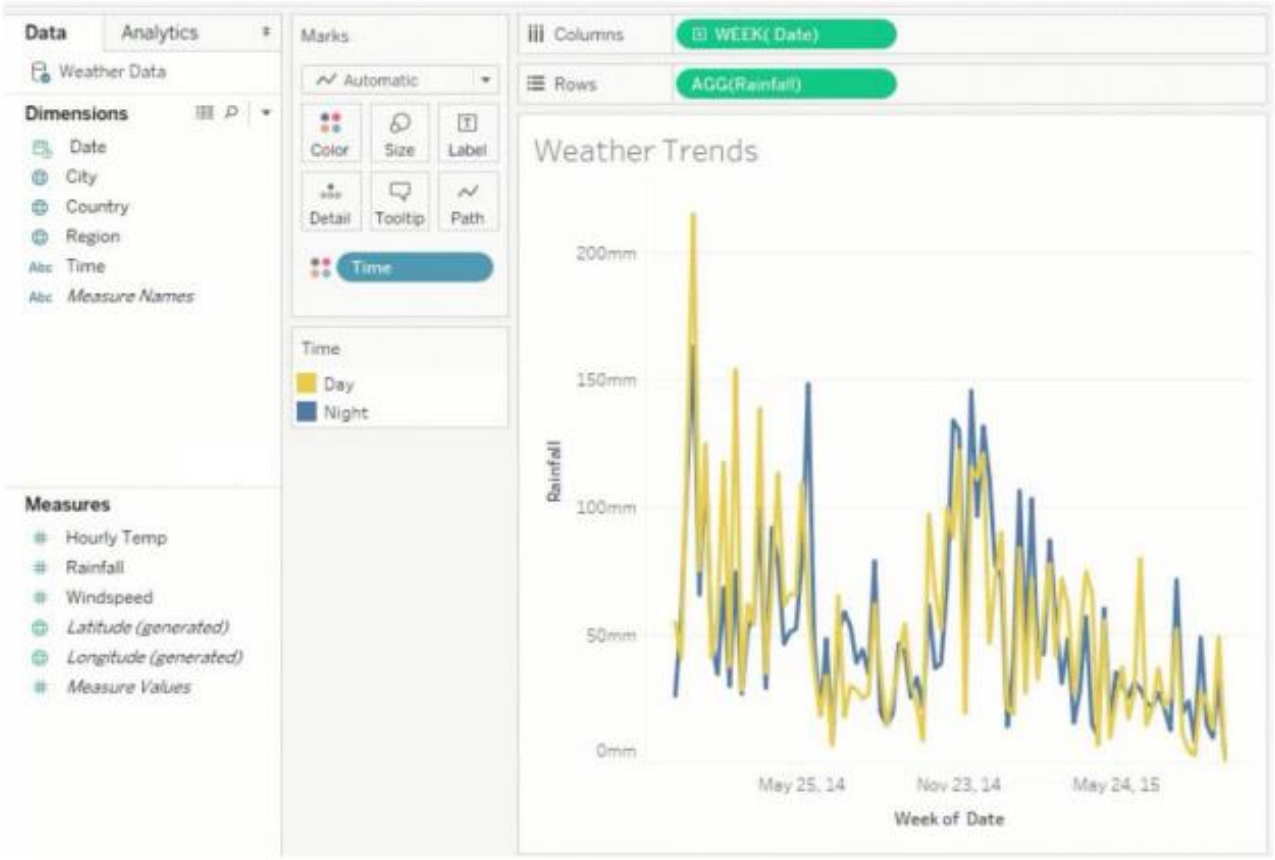

Рисунок 3.6 – Скріншот візуалізації результатів аналізу в Tableau

Переваги рішення:

– аналіз великих даних реальному часі, як In-Memory;
– безпечне спільне використання та спільна робота;
– можливість підключення мобільних пристроїв;
– постійне удосконалення рішення.

Забезпечує доступ та об'єднання розрізнених даних без створення програми.

3.2.3 BI -система QlikView

Інноваційна BI-система QlikView – це платформа для просунутого бізнес-аналізу, що дозволяє створити джерело «єдиної правди» для ухвалення виважених, точних та оперативних управлінських рішень. Це аналітичне ПЗ з категорії BI, яке дозволяє користувачам, які не мають спеціальних технічних

навичок, формувати звіти та аналізувати інформацію про діяльність свого підприємства [38].

Архітектура системи представлена на рис. 3.7.

Стратегія компанії Qlik щодо архітектури та платформ дозволяє створювати автоматизовані конвеєри даних та безпечно надавати їх аналітику у гнучкому масштабі. QlikView володіє потужним ETL-інструментом, який має більше 200 вбудованих функцій для фільтрації, об'єднання та здійснення складних обчислень щодо даних з метою їх завантаження прямо із різноманітних джерел [38].

Платформа QlikView легко адаптується до вимог користувача завдяки гнучкій масштабованості для задоволення пікових вимог постійно зростаючого обсягу даних та складності сценаріїв використання.

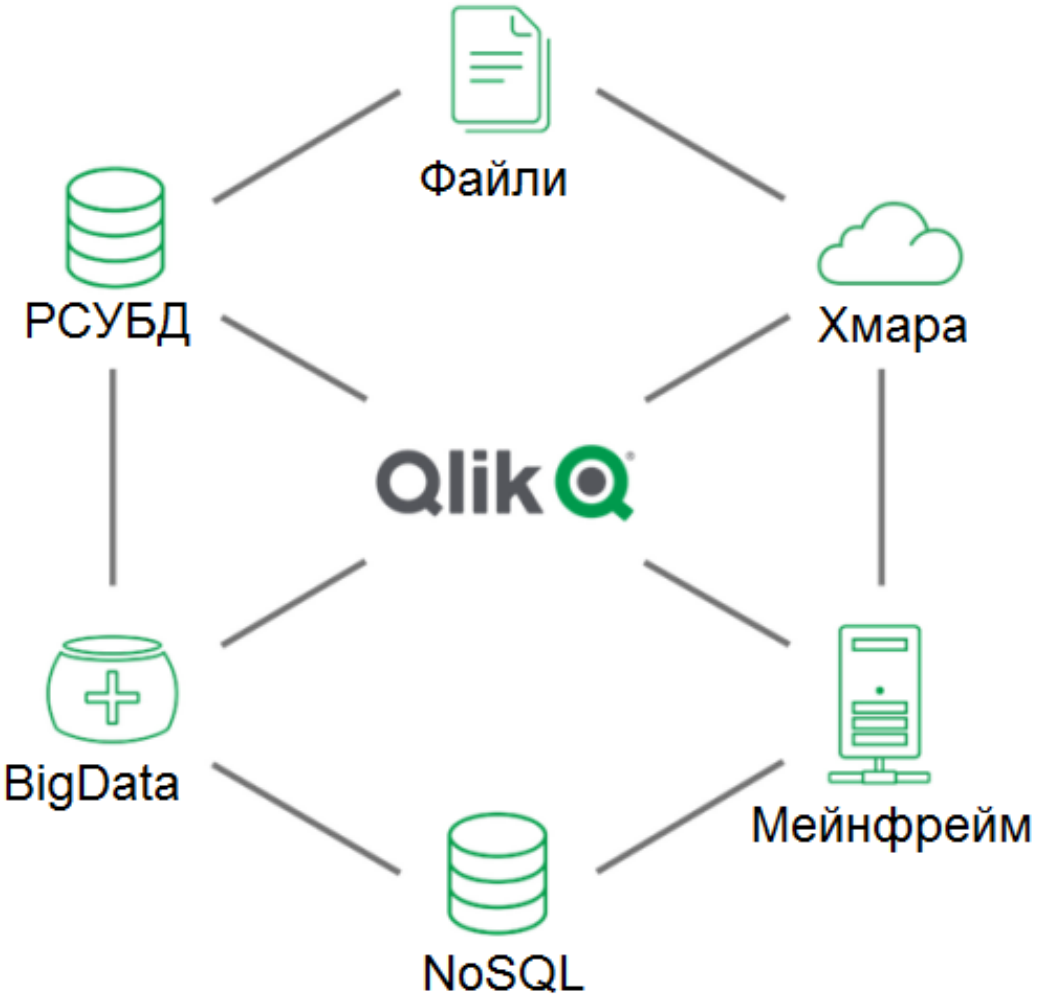

Рисунок 3.7 – Архітектура BI-системи QlikView

QlikView пропонує сучасну хмарну аналітику на основі штучного інтелекту, яка дає всім співробітникам компаній робити важливіші аналізи та приймати більш обґрунтовані рішення. Слід зазначити, що рішення QlikView широко представлені на ринку ПЗ для фінансової сфери.

Для вибору рішення використовуємо таблицю порівняння показників BI-систем (табл. 3.5).

Таблиця 3.5 - Порівняльний аналіз характеристик BI -систем

Характеристики	Power BI	Tableau	QlikView
Продуктивність аналізу великих обсягів даних	2	2	2
Візуалізація даних	2	3	2
Доступ до даних	3	3	2
Наявність безкоштовної версії	3	1	0
Разом	10	9	6

Як готове IT-рішення для аналізу на основі великих даних вибираємо безпосередньо BI - систему Power BI.

3.3 Оцінка ефективності BI-системи

Для розрахунку оцінки ефективності BI -системи використовуємо показник ефективності управління ІВ.

Ефективність управління - це багатопланове поняття, що включає такі компоненти:

– цільову ефективність управління;
– економічну ефективність управління;
– функціональну ефективність управління.

Величина показника функціональної ефективності управління $К_{еу}$ може обчислюватися за допомогою наступної формули:

$$К_{еу} = \frac{\sum_{i=1}^{n} P_{yi}}{n}, \quad (3.2)$$

де n - кількість функцій управління, що реалізуються ІС; P_{yi} - ймовірність вироблення ІС ефективного керуючого впливу при реалізації *i*-ї функції управління [40].

Для ІС з високою ефективністю управління величина цього показника має перевищувати значення 0,5.

Оцінимо ефективність BI-системи з урахуванням показника функціональної ефективності управління.

В даному випадку BI-системи підтримує такі функції:

– створення представлення вітрини даних;
– формування аналітичного звіту;
– аналіз даних.

Як показує практика, лише на першу функцію може негативно вплинути людський чинник.

Нехай ймовірність вироблення ефективного керуючого впливу для цієї функції дорівнює 0,5.

Тоді значення показника функціональної ефективності управління дорівнюватиме $К_{еу}$ = 2,5/3 = 0,833.

Таким чином, коефіцієнт ефективності управління BI-системи $К_{еу}$>0.5, що свідчить про її високу функціональну ефективність управління.

3.4 Висновки до третього розділу

В цьому розділі представлено розробку МПЗ СППР на основі ВВД.

Результати виконаної роботи дозволили зробити такі висновки:

– для формування логічної моделі СППР треба побудувати базові UML -діаграми, котрі відображають різні сторони системи;

– з метою встановлення функціональних вимог використовується діаграма варіантів використання;

– при розробці метамоделей потрібно застосувати MDA -підхід;

– щоб агрегувати дані між різнорідними засобами і платформами, встановлено, що вирішальне значення має моделювання даних BI -системи;

– тепер популярним вважається підхід до проектування ПЗ BI -систем, котрий базується на використанні наявних IT-рішень;

– найкращими параметрами для проведення аналізу на базі ВВД володіє BI -система Power BI.

Коефіцієнт ефективності управління BI -системи $К_{еу}>0.5$, що свідчить про її високу функціональну ефективність управління.

4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

4.1 Режими праці і відпочинку при роботі з ЕОМ

Нервово-емоційне напруження, втома очей, гіподинамія, підвищене навантаження на кисті верхніх кінцівок та хребет – усе це негативний вплив на організм людини при роботі з комп'ютером.

В даному випадку для збереження здоров'я працюючих, запобігання професійним захворюванням і підтримки працездатності передбачаються внутрішньозмінні регламентовані перерви для відпочинку.

Основним нормативно-правовим документом, який регламентує всі питання, пов'язані із охороною праці, в т.ч. і при роботі з ЕОМ, є [43].

Вимоги при роботі з ЕОМ визначають Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин - ДСанПІН 3.3.2.007-98.

Внутрішньозмінні режими праці і відпочинку містять додаткові нетривалі перерви в періоди, що передують появі стомлення і зниження працездатності.

При виконанні робіт, що належать до різних видів трудової діяльності, за основну роботу з візуальними дисплейними терміналами (ВДТ) слід вважати таку, що займає не менше 50% робочого часу. Впродовж робочої зміни мають передбачатися: перерви для відпочинку і вживання їжі (обідні перерви); перерви для відпочинку і особистих потреб (згідно з трудовими нормами); додаткові перерви, що вводяться для окремих професій з урахуванням особливостей трудової діяльності.

Тривалість обідньої перерви визначається чинним законодавством про працю і правилами внутрішнього трудового розпорядку підприємства, організації чи установи. Як правило, тривалість такої перерви становить 40 – 60 хвилин. Тривалість та кількість інших внутрішньозмінних регламентованих перерв залежить від характеру трудової діяльності, напруженості і важності праці і визначається диференційовано для кожної професії.

За характером трудової діяльності розрізняють три професійні групи, згідно з діючим класифікатором професій [44]:

– розробники програм (інженери-програмісти) виконують роботу переважно з відеотерміналом та документацією при необхідності інтенсивного обміну інформацією з ЕОМ і високою частотою прийняття рішень;

– оператори ЕОМ виконують роботу, пов'язану з обліком інформації, одержаної з ВДТ за попереднім запитом, або тієї, що надходить з нього;

– оператор комп'ютерного набору виконує одноманітні за характером роботи з документацією та клавіатурою і нечастими нетривалими переключеннями погляду на екран дисплея, з введенням даних з високою швидкістю.

Правилами встановлюються такі внутрішньозмінні режими праці та відпочинку при роботі з ЕОМ при 8-годинній денній робочій зміні в залежності від характеру праці:

– для розробників програм із застосуванням ЕОМ слід призначати регламентовану перерву для відпочинку тривалістю 15 хвилин через кожну годину роботи за ВДТ;

– для операторів із застосуванням ЕОМ слід призначати регламентовані перерви для відпочинку тривалістю 15 хвилин через кожні дві години;

– для операторів комп'ютерного набору слід призначати регламентовані перерви для відпочинку тривалістю 10 хвилин після кожної години роботи.

У всіх випадках, коли виробничі обставини не дозволяють застосувати регламентовані перерви, тривалість безперервної роботи з ВДТ не повинна перевищувати 4 години.

При 12-годинній робочій зміні регламентовані перерви повинні встановлюватися в перші 8 годин роботи аналогічно перервам при 8-годинній робочій зміні, а протягом останніх чотирьох годин роботи, незалежно від характеру трудової діяльності, через кожну годину тривалістю 15 хвилин. Необхідно зазначити, що перерви під час роботи не повинні бути строго визначені за часом, а необхідно передбачати певний творчий індивідуальний підхід.

Для запобігання втоми під час деяких перерв доцільно виконувати спеціальні вправи, які наведені у ДСанПІН 3.3.2.007-98. Також ці санітарні правила визначають Відстань від екрана до ока фахівців, які працюють за комп'ютером.

Окрім ДНАОП та ДСанНіП, які регламентують вимоги безпеки та санітарно-гігієнічні вимоги до обладнання робочих місць користувачів ВДТ, є ще інші нормативні акти загального призначення, які необхідно враховувати під час організації роботи користувачів ВДТ. Важливим нормативним актом є [45].

Гігієнічна класифікація праці необхідна для оцінки конкретних умов та характеру праці на робочих місцях. На основі такої оцінки приймаються рішення, спрямовані на запобігання або максимальне обмеження впливу несприятливих виробничих факторів.

4.2 Вплив електромагнітного імпульсу ядерного вибуху на елементи виробництва та заходи захисту

У воєнний час при застосуванні ядерної зброї проти України на електронно-обчислювальне обладнання в першу чергу буде впливати електромагнітний імпульс **(**EMI) ядерного вибуху у вигляді короткого імпульсу, який вражає головним чином електричну та електронну апаратуру. EMI виникають в основному в результаті взаємодії гамма-випромінювання з атомами навколишнього середовища. На утворення EMI йде невелика кількість ядерної енергії, але він здатен викликати високі імпульси струмів та напруг в кабелях повітряних і підземних ліній зв'язку, сигналізації, управління, електропередачі, в антенах радіостанцій. Вплив EMI може привести до згортання чутливих електронних та електричних елементів, зв'язаних з великими антенами чи відкритими дротами, а також до порушень в обчислювальних пристроях. Вплив EMI необхідно враховувати для всіх електричних та електронних систем. Для найбільш важливих приладів треба використовувати засоби захисту і підвищувати їх стійкість до EMI [46].

Особливістю ЕМІ, як вражаючого фактору є його здатність розповсюджуватись на десятки і сотні кілометрів в оточуючому середовищі. Тому ЕМІ може вплинути своєю дією на об'єкти, там де вибухова хвиля, світлове випромінювання, проникаюча радіація втрачають своє значення, як вражаючі фактори. При наземних та низьких повітряних вибухах в лініях зв'язку та електрозабезпечення виникають напруги, які можуть викликати пробій ізоляції провідників та кабелів відносно землі, пробій ізоляції елементів приладів підключених до повітряних і підземних ліній. Степінь враження залежить від наведеного імпульсу напруги чи струму і також електричної міцності обладнання.

Найбільш піддані впливу ЕМІ системи зв'язку, сигналізації, управління. Використані в цих системах кабелі та апаратура мають обмежену електричну міцність не більше 10кВ імпульсної напруги, тоді як наведені імпульси напруги від ЕМІ можуть перевищувати ці значення. Найбільш піддана впливу ЕМІ апаратура виконана на напівпровідниках та інтегральних схемах, працюючих на малих струмах і напругах, і значить відчутних до впливу зовнішніх електричних і магнітних кіл, в тому числі і елементи програмного засобу для управління процесом міграції віртуальних машин в обчислювальній хмарі. ЕМІ пробиває ізоляцію, спалює елементи електричних схем радіоапаратури, викликає коротке замикання в радіопристроях, іонізацію діелектриків, змінює або повністю стирає магнітний запис. Встановлено, що при дії ЕМІ на апаратуру найбільша напруга наводиться на вході. В транзисторах відбувається така залежність: чим більший коефіцієнт підсилення транзистора, тим менша його електрична міцність.

ЕМІ пошкоджує також резистори, викликає іскріння в їх міжконтактних з'єднаннях і деяких областях провідної поверхні. Найбільшу небезпеку ЕМІ представляє для апаратури, яка встановлена в особливо міцних спорудах, які витримують великі тиски ударної хвилі. В цих спорудах апаратура не виходить з ладу від механічних пошкоджень, але ЕМІ може вивести з ладу всю незахищену апаратуру системи зв'язку, сигналізації і керування. Найбільших значень досягають напруги, які наводяться між кабелем і землею. Напруженість електромагнітного поля всередині споруди в деяких випадках недостатня для

того, щоб вивести з ладу апаратуру, але такі поля в змозі викликати короткочасний збій роботи радіотехнічних пристроїв.

Розглянемо можливі шляхи рішення задачі захисту від ЕМІ. Ідеальним захистом від ЕМІ виявилося б повне укриття приміщення, в якому розміщена радіоелектронна апаратура, металевим екраном [46]. Водночас зрозуміло, що практично забезпечити такий захист у ряді випадків неможливо, тому що для роботи апаратури часто потрібно забезпечити її електричний зв'язок із зовнішніми пристроями. Тому використовуються менш надійні засоби захисту, такі, як струмопровідні сітки, або плівкові покриття для вікон, щільникові металеві конструкції для повітрезабірників і вентиляційних отворів і контактні пружинні прокладки, розміщувані по периметру дверей і люків.

Більш складною технічною проблемою рахується захист від проникнення ЕМІ в апаратуру через різноманітні кабельні входи. Радикальним рішенням даної проблеми міг би стати перехід від електричних мереж зв'язку до практично не схильних до впливу ЕМІ волоконно-оптичних. Проте заміна напівпровідникових приладів у всьому спектрі виконуваних ними функцій електронно-оптичними пристроями можлива тільки у віддаленому майбутньому. Тому в даний час в якості засобів захисту кабельних входів найбільші широко використовуються фільтри, у тому числі волоконні, а також іскрові розрядники, металлоокисні варистори і високошвидкісні зенеровські діоди.

Всі ці засоби мають як переваги, так і недоліки. Так, ємнісно-індуктивні фільтри достатньо ефективні для захисту від ЕМІ малої інтенсивності, волоконні фільтри захищають у відносно вузькому діапазоні надвисоких частот. Іскрові розрядники мають значну інерційність й в основному придатні для захисту від перевантажень, що виникають під впливом напруг і струмів, що наводяться в обшивці літака, кожусі апаратури й оплетенні кабеля.

Металоокисні варистори є напівпровідниковими приладами, що різко підвищують свою провідність при високій напрузі. Проте, при застосуванні цих приладів у якості засобів захисту від ЕМІ варто враховувати їх недостатньо високу

швидкодію і погіршення характеристик при кількаразовому впливі навантажень. Ці недоліки відсутні у високошвидкісних зенеровських діодах, дія яких заснована на різкій лавиноподібній зміні опору від високого значення практично до нуля, при перевищенні прикладеної до них напруги граничного розміру. Крім того на відміну від варисторів характеристики зенеровських діодів після багатократних впливів високих напруг і переключень режимів не погіршуються.

Найбільш раціональним підходом до проектування засобів захисту від ЕМІ кабельних входів є створення таких роз'ємів у конструкції яких передбачені спеціальні заходи, що забезпечують формування елементів фільтрів і установку вмонтованих зенеровських діодів. Використання роз'ємів подібної конструкції дозволить вирішити проблему обмеження малогабаритних характеристик пристрою захисту. Складність рішення задачі захисту від ЕМІ і висока вартість розроблених для цього засобів і методів змушують піти по шляху їхнього вибіркового застосування в особливо важливих системах зброї і військової техніки [46]. Такий же шлях обраний і для захисту систем, що мають велику протяжність, керування і зв'язку. Проте основним методом вирішення проблеми спеціалісти вважають створення так званих розподілених мереж зв'язку.

4.3 Висновки до четвертого розділу

В цьому розділі розглянуто важливі питання охорони праці та безпеки в надзвичайних ситуаціях, зокрема описані вимоги до режимів праці і відпочинку при роботі з ЕОМ, а також вплив ЕМІ ядерного вибуху на елементи виробництва та заходи захисту.

ВИСНОВКИ

Кваліфікаційна робота присвячена актуальній проблемі дослідження та розробки МПЗ, що дозволяє підвищити ефективність СППР на основі ВВД.

Виконані у роботі наукові дослідження представлені такими основними результатами:

– проаналізовано сучасний стан проблеми розробки МПЗ СППР на базі ВВД. У деяких роботах автори звертають увагу на необхідність розробки інструментів, що поєднують функціональну міць BI – систем із здатністю до обробки більших обсягів даних. Для досягнення цієї мети пропонується використовувати різні технології, що базуються на застосуванні вітрин даних. Водночас аналіз дозволив констатувати недостатність робіт із цієї проблеми, що підтверджує актуальність теми роботи;

– проведено аналіз методологій та технологій побудови СППР на основі ВВД. Згідно аналізу, під час розробки ВВД є доцільним використання методів побудови логічних вітрин даних. Найбільш перспективним для побудови BI -систем для аналізування великих даних варто застосувати технологію, котра підтримує механізм In-Memory;

– створено МПЗ СППР. При побудові логічної моделі СППР сформовано базові UML -діаграми. Для створення метамоделей застосовано MDA -підхід. Проведений порівняльний аналіз дозволив констатувати, що найкращими параметри має Power BI система для аналізу на основі ВВД. Виконані обчислення підтвердили ефективність використаного рішення СППР.

Отже у кваліфікаційній роботі розв’язано актуальну науково-практичну проблему дослідження та розробки МПЗ, що забезпечує підвищення ефективності СППР, побудованої на основі ВВД.

ПЕРЕЛІК ДЖЕРЕЛ

1. Davenport, Thomas H., Harris, Jeanne G. Competing on Analytics: The New Science of Winning. Harvard Business Review Press, 2007. ISBN: 978-1422103326.

2. Data Warehouse, Data Marts and Online Analytical Processing (OLAP) [Електронний ресурс] – Режим доступу: https://www.managementstudyguide.com/datawarehouse-data-marts-and-olap.htm (дата звернення: 04.04.2025).

3. What is a Data Mart? [Електронний ресурс] – Режим доступу: https://www.oracle.com/autonomous-database/what-is-data-mart/ (дата звернення: 04.4.2025).

4. Бідюк П.І., Тимощук О.Л., Коваленко А.Є., Коршевнюк Л.О. СИСТЕМИ І МЕТОДИ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ. Підручник. Київ: КПІ ім. Ігоря Сікорського, 2022. – 610 с.

5. Gartner, Inc. [Електронний ресурс] – Режим доступу: https://www.gartner.com (дата звернення: 06.0410.2025).

6. R. Kimball “The Data Warehouse Toolkit: the Complete Guide to Dimensional Modeling”, 2002.

7. Акіменко В.В. Прикладні задачі інтелектуального аналізу даних (DATA MINING). – К.: КНУ ім. Тараса Шевченко, 2018. – 152 с.

8. B. Sudrajat “Analysis and Design of Data Mart Decision Support Systems at PT Marlindo Tirta Nusantara”, Journal Publications & Informatics Engineering Research Volume 3, Number 2, April 2019.

9. Mkrtychev, S.; Enik, O. Automated Underwriting Control in a Regional Insurance Company, Proceedings of the International Conference on Trends of Technologies and Innovations in Economic and Social Studies on October 02, 2018.

10. R. A. Jaleel and T.M.J. Abbas “Design and Implementation of Efficient Decision Support System Using Data Mart Architecture”, Proc. of the 2 nd

International Conference on Electrical, Communication and Computer Engineering (ICECCE) 12-13 June 2020, Istanbul, Turkey

11. N. Houari and B. H. Far “An Intelligent Project lifecycle Data Mart-based Decision Support System”, Canadian Conference on Electrical and Computer Engineering 2004 (IEEE Cat. No.04CH37513).

12. Ahmad, S. Azhar, P. Lukauskis “Development of a decision support system using data warehousing to assist builders/developers in site selection”, Automation in Construction 13 (2004) 525 – 542.

13. A.A.A. Gad-Elrab Modern Business Intelligence: Big Data Analytics and Artificial Intelligence for Creating the Data-Driven Value [Електронний ресурс] – Режим доступу: https://www.intechopen.com/chapters/76332 (дата звернення: 10.04.2025).

14. Інформаційна технологія побудови розподілених сховищ даних гібридного типу [Текст] : автореф. дис. ... канд. техн. наук : 05.13.06 / Яцишин Андрій Юрійович ; Нац. техн. ун-т України "Київ. політехн. ін-т". - Київ, 2016. - 21 с. : рис.

15. Озера даних та сховища даних – хмарні джерела для аналітики в режимі реального часу. [Електронний ресурс] – Режим доступу: https://www.rbcgrp.com/ua/ozera-danih-ta-shovishha-danih-hmarni-dzherela-dlja-analitiki-v-rezhimi-realnogo-chasu/ (дата звернення: 12.04.2025).

16. Analytics on AWS [Електронний ресурс] – Режим доступу: https://aws.amazon.com/what-is/data-lake/?nc1=h_ls (дата звернення: 12.04.2025).

17. Bhargava S. “DWH-Performance Tuning Using Metadata Driven Approach”, Dissertation Submitted In partial fulfillment For the award of the Degree of Master of Technology, 2011.

18. Корпоративні сховища даних [Електронний ресурс] – Режим доступу: https://ua.ibagroupit.com/solutions-and-products/analytics/dwh/ (дата звернення: 14.04.2025).

19. Асєєв Г. Архітектура корпоративного сховища даних // Вісник Книжкової палати. – 2010. – № 10. – С. 20-25.

20. Logical Data Warehouse [Електронний ресурс] – Режим доступу: https://www.tibco.com/reference-center/what-is-a-logical-data-warehouse (дата звернення: 16.04.2025).

21. Teradata Database Design [Електронний ресурс] – Режим доступу: https://docs.teradata.com/r/w4DJnG9u9GdDlXzsTXyItA/Iipw7BdwMhNUKxZA BfKDBg (дата звернення16.04.2025).

22. Relational online analytical processing (ROLAP) [Електронний ресурс] – Режим доступу: https://searchoracle.techtarget.com/definition/relational-onlineanalytical-processing (дата звернення16.04.2025).

23. Юрчак В. В. Технології побудови BI-систем для аналізу великих даних // Інформаційні моделі, системи та технології: Праці XII наук.-техн. конф. Тернопіль, 2024. с. 113.

24. In-Memory Processing [Електронний ресурс] – Режим доступу: https://hazelcast.com/glossary/in-memory-processing/ (дата звернення: 17.04.2025).

25. Bara A. et al. A model for Business Intelligence Systems' Development, Informatica Economică vol. 13, no. 4/2009, pp. 99-107.

26. Model-View-Controller design pattern [Електронний ресурс] – Режим доступу: https://help.hcltechsw.com/commerce/9.1.0/developer/concepts/csdmvcdespat.html (дата звернення: 18.04.2025).

27. Kalelkar M. et al. Implementation of Model-View-Controller Architecture Pattern for Business Intelligence Architecture, International Journal of Computer Applications (0975 – 8887) Volume 102– No.12, September 2014.

28. Glass, Russell, Callahan, Sean. The Big Data-Driven Business: How to Use Big Data to Win Customers, Beat Competitors, and Boost Profits. Wiley, 2014. ISBN: 978-1118889800.

29. Петрик М.Р., Мудрик І.Я. Архітектура програмного забезпечення (на базі використання CASE-засобів IBM(sad)) навчальний посібник, Тернопіль: ТНТУ імені Івана Пулюя, 2017. 100 с.

30. Об'єктно-орієнтоване моделювання програмних систем: Навчальний посібник. – Львів: Видавничий центр ЛНУ імені Івана Франка, 2007. - 108 с

31. Object Management Group [Електронний ресурс] – Режим доступу: https://www.omg.org/ (дата звернення: 20.04.2025).

32. Stefanov V., List B., Korherr B. (2005) Extending UML 2 Activity Diagrams with Business Intelligence Objects. In: Tjoa A.M., Trujillo J. (eds) Data Warehousing and Knowledge Discovery. DaWaK 2005. Lecture Notes in Computer Science, vol 3589. Springer, Berlin, Heidelberg.

33. Юрчак В. В. Процес розробки метамоделі об'єктів BI-системи // VIII Міжнародна студентська науково-технічна конференція «Природничі та гуманітарні науки. Актуальні питання». Тернопіль, 2025. с. 226..

34. Інформаційно-аналітичні системи та технології: комп’ютерний практикум: для студ. спец. 051 «Економіка», освітньої програми “Економічна аналітика”, першого (бакалаврського) рівня вищої освіти / КПІ ім. Ігоря Сікорського; уклад.: О.В.Цеслів. А.С.Коломієць –Київ : КПІ ім. Ігоря Сікорського, 2024. – 141 с.

35. Provost, Foster, Fawcett, Tom. Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking. O'Reilly Media, 2013. ISBN: 978-1449361327..

36. MS Power BI [Електронний ресурс] – Режим доступу: https://powerbi.microsoft.com/en-us/desktop/ (дата звернення: 22.04.202521).

37. Платформа Tableu [Електронний ресурс] – Режим доступу: https://www.tableau.com/products/our-platform (дата звернення: 23.04.202521).

38. Сайт компании Qlik [Електронний ресурс] – Режим доступу: https://www.qlik.com/us/ (дата звернення: 23.04.2025).

39. Kharchenko, O., Raichev, I., Bodnarchuk, I., Zagorodna, N.: Optimization of software architecture selection for the system under design and reengineering. In: 2018 14th International Conference on Advanced Trends in Radioelecrtronics, Telecommunications and Computer Engineering (TCSET), pp. 1245–1248. IEEE (2018).

40. Інформаційні системи в економіці : навчальний посібник / Пономаренко В. С., Золотарьова І. О., Бутова Р. К. та ін. – Х. : Вид. ХНЕУ, 2011. – 176 с.

41. Zagorodna, N., Skorenkyy, Y., Kunanets, N., Baran, I., Stadnyk, M. Augmented Reality Enhanced Learning Tools Development for Cybersecurity Major. CEUR Workshop Proceedings., 2022, 3309, pp. 25–32. https://ceur-ws.org/Vol-3309/short1.pdf.

42. Vyacheslav Nykytyuk, Vasyl Dozorskyy, Nataliia Kunanets, Volodymyr Pasichnyk, Oleksandr Matsiuk, Ihor Bodnarchuk: Electrical Probe-Signal Processing and Criterion for the Determination of Time Parameters of the Teeth Filling Material Polymerization Process in Dentistry. 4th IDDM 2021: Valencia, Spain. P. 54-63.

43. Закон України «Про охорону праці». [Електронний ресурс] – Режим доступу: https://zakon.rada.gov.ua/laws/show/2694-12 - (дата звертання 06.05.2025).

44. Класифікатор професій ДК 003:2010/ [Електронний ресурс] – Режим доступу: https://zakon.rada.gov.ua/rada/show/va327609-10 - (дата звертання 06.05.2025).

45. Гігієнічна класифікація умов праці за показниками шкідливості та небезпечності факторів виробничого середовища, важкості та напруженості трудового процесу. – К.: МОЗ України, 1998. – 34 с.

46. Безпека в надзвичайних ситуаціях. Методичний посібник для здобувачів освітнього ступеня «магістр» всіх спеціальностей денної та заочної (дистанційної) форм навчання / укл.: Стручок В. С. Тернопіль: ФОП Паляниця В. А., 2022. 156 с.

ДОДАТКИ

ДОДАТОК А
Тези конференції

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
імені ІВАНА ПУЛЮЯ
НАУКОВЕ ТОВАРИСТВО ім. ШЕВЧЕНКА

М А Т Е Р І А Л И

XII НАУКОВО-ТЕХНІЧНОЇ КОНФЕРЕНЦІЇ

«ІНФОРМАЦІЙНІ МОДЕЛІ, СИСТЕМИ ТА ТЕХНОЛОГІЇ»

18–19 грудня 2024 року

ТЕРНОПІЛЬ
2024

С. Шиндеровський
ДОСЛІДЖЕННЯ МЕХАНІЗМІВ ЗАХИСТУ ДЛЯ СЕРВЕРНОЇ ЧАСТИНИ НА NODE.JS
S. Shinderovskiy
RESEARCH ON PROTECTION MECHANISMS FOR THE SERVER SIDE ON THE NODE.JS 109

В. Шендрик, Ю. Парфененко, І. Захарченко
РЕІНЖИНІРИНГ ІНФОРМАЦІЙНОЇ СИСТЕМИ ПІДТРИМКИ УПРАВЛІННЯ ЕНЕРГЕТИЧНИМИ МІКРОМЕРЕЖАМИ З ВІДНОВЛЮВАЛЬНИМИ ДЖЕРЕЛАМИ ЕНЕРГІЇ
V. Shendryk, Y. Parfenenko, I. Zakharchenko
RE-ENGINEERING OF THE INFORMATION SYSTEM TO SUPPORT THE MANAGEMENT OF ENERGY MICROGRIDS WITH RENEWABLE ENERGY SOURCES 110

Т. Щур
ЕФЕКТИВНЕ УПРАВЛІННЯ ІНВЕНТАРЕМ ЗА ДОПОМОГОЮ ДРОНІВ І ШТУЧНОГО ІНТЕЛЕКТУ
T. Shchur, H. Osukhivska, EFFECTIVE INVENTORY MANAGEMENT USING DRONES AND ARTIFICIAL INTELLIGENCE 111

О. С. Юречко
АНСАМБЛЕВІ МЕТОДИ МАШИННОГО НАВЧАННЯ
P. S. Yurechko
ENSEMBLE METHODS OF MACHINE LEARNING 112

В. В. Юрчак
ТЕХНОЛОГІЇ ПОБУДОВИ BI-СИСТЕМ ДЛЯ АНАЛІЗУ ВЕЛИКИХ ДАНИХ
V. V. Yurchak
TECHNOLOGIES OF BUILDING BI-SYSTEMS FOR BIG DATA ANALASYS 113

О. Яремa
ЕВРИСТИЧНІ МЕТОДИ ГЕНЕРАЦІЇ S-БЛОКІВ ТА ЇХ ПРОГРАМНА РЕАЛІЗАЦІЯ
O. Yarema
PRACTICAL ASPECTS OF RESEARCH ON THE RESISTANCE OF S-BLOCKS TO DIFFERENTIAL CRYPTANALYSIS 114

СЕКЦІЯ 3. КОМП'ЮТЕРНІ СИСТЕМИ ТА МЕРЕЖІ

В. С. Бондаренко
РОЗРОБКА ДОДАТКУ ВІДЕОТРАНСЛЯЦІЇ ПІД МОБІЛЬНІ ПРИСТРОЇ НА БАЗІ ОПЕРАЦІЙНОЇ СИСТЕМИ ANDROID
V. S. Bondarenko
DEVELOPMENT OF A VIDEO BROADCASTING APPLICATION FOR MOBILE DEVICES BASED ON THE ANDROID OPERATING SYSTEM 115

І. Бородій, Г. Осухівська
АРХІТЕКТУРА ПРОГРАМНОЇ СИСТЕМИ ПРОГНОЗУВАННЯ СТАНУ ЯКОСТІ АТМОСФЕРНОГО ПОВІТРЯ
I. Borodii, H. Osukhivska
ARCHITECTURE OF A SOFTWARE SYSTEM FOR FORECASTING THE STATE OF ATMOSPHERIC AIR QUALITY 116

В. Бражніков
АКТУАЛЬНІСТЬ ЗАСОБІВ ОПТИМІЗАЦІЇ CRM-СИСТЕМ ДЛЯ ЛОГІСТИКИ
V. Brazhnikov
RELEVANCE OF CRM SYSTEM OPTIMIZATION TOOLS FOR LOGISTICS 117

УДК 004.6

В. В. Юрчак
(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

ТЕХНОЛОГІЇ ПОБУДОВИ BI-СИСТЕМ ДЛЯ АНАЛІЗУ ВЕЛИКИХ ДАНИХ

UDC 004.6

V. V. Yurchak

TECHNOLOGIES OF BUILDING BI-SYSTEMS FOR BIG DATA ANALASYS

При побудові BI (Business Intelligence) - систем для аналізу великих даних для різних завдань, зазвичай використовується технологія ROLAP. Реляційна онлайн-аналітична обробка (Relational OLAP, ROLAP) - це форма OLAP (online analytical processing – аналітична обробка у реальному часі), яка виконує динамічний багатовимірний аналіз даних, що зберігаються в реляційній базі даних, а не в багатовимірній базі даних.

Принцип роботи такої моделі досить простий: при активізації аналітичної функції формується SQL – запит, який йде на той чи інший бекенд. При цьому BI -система може бути інтегрована з будь-якою СУБД, що підтримує мову SQL (наприклад, реляційні СУБД MySQL, PostgreSQL та інші).

Перевага даної технології у тому, що для роботи BI- системи немає обмежень за обсягом даних. Недолік – відносно невисока швидкість виконання запитів, що негативно впливає на продуктивність BI -системи.

Для вирішення цієї проблеми використовуються різні підходи, у тому числі застосування спеціалізованих колонкових СУБД, наприклад ClickHouse або Vertica. Однак колонкові СУБД використовують власний обрізаний діалект SQL, що суттєво знижує функціональні можливості такого рішення.

Більш перспективним є рішення, засноване на застосуванні СУБД, в яких реалізована технологія In-Memory, що є технологією виконання комп'ютерних обчислень повністю в пам'яті комп'ютера, наприклад, в оперативному запам'ятовучому пристрої (ОЗП).

Цей термін, зазвичай, має на увазі великомасштабні складні обчислення, які вимагають спеціального системного програмного забезпечення для виконання обчислень на комп'ютерах, що працюють разом у кластері. Як кластер комп'ютери об'єднують свою оперативну пам'ять, тому розрахунок, по суті, виконується між комп'ютерами і використовує загальний простір ОЗП всіх комп'ютерів разом. Обробка в пам'яті усуває всі повільні звернення до даних і покладається виключно на дані, що зберігаються в ОЗП [1]. Загальна продуктивність обробки не знижується через затримку, яка зазвичай спостерігається при доступі до жорстких дисків або твердотільних накопичувачів. Обробка в пам'яті часто виконується за допомогою технології відомої як мережі даних у пам'яті.

Такі СУБД мають перевагу у швидкості порівняно з традиційними дисковими СУБД. Багато існуючих реляційних СУБД підтримують механізм In Memory, наприклад, пізні версії MS SQL Server, Oracle Database та інші.

За результатами представленого аналізу вибираємо як технологію побудови BI -системи для аналізу великих даних технологію, засновану на застосуванні реляційної СУБД, що підтримує механізм In-Memory.

Література

1. In-Memory Processing [Електронний ресурс]. – Режим доступу: https://hazelcast.com/glossary/in-memory-processing/ (дата звертання: 04.12.2024).

Міністерство освіти і науки України
Тернопільський національний технічний університет
імені Івана Пулюя
Маріборський університет (Словенія)
Технічний університет в Кошице (Словаччина)
Каунаський технологічний університет (Литва)
Львівський національний університет
імені Івана Франка,
Гірничо-металургійна академія ім. Станіслава Сташиця (Польща)
Луцький національний технічний університет,
Чернівецький національний університет
імені Юрія Федьковича,
Вроцлавський економічний університет (Польща)
Університет технологій та економіки
імені Хелени Ходковської (Польща)
Донбаська державна машинобудівна академія

Студентське наукове товариство

VIII МІЖНАРОДНА

студентська науково - технічна конференція

"ПРИРОДНИЧІ ТА ГУМАНІТАРНІ НАУКИ.

АКТУАЛЬНІ ПИТАННЯ"

24-25 квітня 2025 р.

(збірник тез конференції)

Тернопіль 2025

Хемій С. **ПРОГРАМНА СИСТЕМА ДЛЯ МАТЕМАТИЧНОГО МОДЕЛЮВАННЯ СПЕКТРАЛЬНИХ ХАРАКТЕРИСТИК ЕЛЕКТРОННИХ СТАНІВ З РІЗНИМИ ЕФЕКТИВНИМИ ПОТЕНЦІАЛАМИ**	**219**
Холод Ю. **ПРОГРАМНИЙ МОДУЛЬ ДЛЯ ПЕРЕГЛЯДУ ЗОБРАЖЕНЬ**	**220**
Хромов В. **РОЗРОБКА RESTFUL API ДЛЯ ВЕБ-ДОДАТКІВ З ВИКОРИСТАННЯМ NODE.JS ТА EXPRESS**	**222**
Чендей Б. **РОЗРОБКА ВЕБ-ПЛАТФОРМИ ДЛЯ НАДАННЯ ПОСЛУГ РЕПЕТИТОРСТВА З ПРОГРАМУВАННЯ З ВИКОРИСТАННЯМ REACT.JS**	**223**
Shtokalo A. **VOICE-DRIVEN CODING ENVIRONMENTS: ENHANCING ACCESSIBILITY AND EFFICIENCY**	**224**
Юрчак В. **ПРОЦЕС РОЗРОБКИ МЕТАМОДЕЛІ ОБ'ЄКТІВ BI-СИСТЕМИ**	**226**
Недошитко Л., Яворський Б **ШТУЧНИЙ ІНТЕЛЕКТ У МЕДИЦИНІ: РЕВОЛЮЦІЯ В ОХОРОНІ ЗДОРОВ'Я**	**228**
Якуб'як Ю. **РЕАЛІЗАЦІЯ АЛГОРИТМУ ОПТИМІЗАЦІЇ ПРОЦЕСУ ЗБОРУ ДАНИХ ПОТЕНЦІЙНИХ КЛІЄНТІВ**	**229**
Горват М. **ЧИСЕЛЬНЕ РОЗВ'ЯЗАННЯ ДИФЕРЕНЦІАЛЬНИХ РІВНЯНЬ ПРИ ПРОВЕДЕННІ РОЗРАХУНКІВ НА ЖОРСТКІСТЬ**	**232**
Драбик І. **ПРО ЧИСЛЕНЕ РОЗВ'ЯЗАННЯ ЗВИЧАЙНИХ ДИФЕРЕНЦІАЛЬНИХ РІВНЯНЬ**	**234**
Мартинюк С. **ЛОГІЧНИЙ ПРИЙОМ ПОРІВНЯННЯ В МАТЕМАТИЧНОМУ АНАЛІЗІ**	**236**
Островський О. **РОЗВ'ЯЗОК ДИФЕРЕНЦІАЛЬНОГО РІВНЯННЯ ПРОХОДЖЕННЯ СТРУМУ В КАБЕЛІ**	**238**
Дубиняк Т., Цюпа С., Микулик П. **ЗАЛЕЖНІСТЬ ІНДУКТИВНОСТІ КОТУШКИ ВІД ВЕЛИЧИНИ ПОВІТРЯНОГО ЗАЗОРУ ПРИ КОНТРОЛІ ШОРСТКОСТІ**	**240**

УДК 004.056
Юрчак В. – ст. гр. СНям-61
Тернопільський національний технічний університет імені Івана Пулюя

ПРОЦЕС РОЗРОБКИ МЕТАМОДЕЛІ ОБ'ЄКТІВ BI-СИСТЕМИ

Науковий керівник: професор, доктор технічних наук
Литвиненко Я. В.

Yurchak V.
Ternopil Ivan Puluj National Technical University

PROCESS OF DEVELOPING A METAMODEL OF BI– SYSTEM OBJECTS

Supervisor: professor, Dr. Sci. Yaroslav Lytvynenko

Ключові слова: метамодель, BI-система, вітрина даних
Keywords: metamodel, BI system, data mart

Метамодель - це модель, котра описує іншу якусь модель або транзитивне відношення між двома моделями. Для побудови метамоделей варто застосувати підхід MDA (Model Driven Architecture) [1].

Виділяються три основні категорії об'єктів BI (Business Intelligence)- системи [2]:

Для розробки метамоделі об'єктів BI-системи використовуємо діаграми класів. Діаграми класів описують внутрішню структуру класів та відносини між класами. Сюди входять відносини класів, а також атрибути та поведінки, пов'язані з кожним класом. Діаграми класів чудово ілюструють успадкування та складові відносини.

На діаграмі класів UML представлені:

- класи в об'єктно-орієнтованій системі;
- атрибути та методи класів;
- зв'язок між класами.

Не представлені на діаграмі класів UML:

- деталі того, як класи взаємодіють один з одним;
- алгоритмічні деталі, тобто як конкретна поведінка класу реалізована.

Діаграми класів відмінно підходять для:

- виявлення пов'язаних даних та атрибутів;
- отримання швидкого уявлення про важливі об'єкти у системі;
- з'ясування достатності чи недостатності класів;
- визначення, чи є відносини між об'єктами надто складними чи простими;
- виявлення залежностей між класами чи об'єктами.

Разом з тим діаграми класів не дозволяють вирішити такі питання, як виявлення алгоритмічної (не керованої даними) поведінки та розуміння загального потоку управління додатком.

Метамодель BI-системи на основі логічної вітрини даних – рис. 1.

Тут: СД – сховище даних; СОД - сховище операційних даних; ЛВД - логічна вітрина даних.