

ШАХОВСЬКА НАТАЛІЯ

Національний університет «Львівська політехніка»

<https://orcid.org/0000-0002-6875-8534>e-mail: Nataliya.b.shakhovska@lpnu.ua**ЗАГОРОДНИЙ ІВАН**

Національний університет «Львівська політехніка»

<https://orcid.org/0009-0000-0104-6252>e-mail: ivan.i.zahorodnyi@lpnu.ua**КОГУЧ ОКСАНА**

Національний університет «Львівська політехніка»

<https://orcid.org/0009-0004-6861-8082>e-mail: oksana.h.kohuch@lpnu.ua**ТРИСКА РОМАН**

Національний університет «Львівська політехніка»

<https://orcid.org/0009-0001-6403-1469>e-mail: roman.m.triska@lpnu.ua

ІНТЕРФЕЙСИ ВІРТУАЛЬНИХ СХОВИЩ ДАНИХ В УМОВАХ ШВИДКОЗМІНЮВАНОВОГО ІНФОРМАЦІЙНОГО СЕРЕДОВИЩА

В статті проведено огляд сучасних віртуальних сховищ даних, їх структури, основні переваги та виклики. Виокремлено основні технології, які використовуються в цій сфері, а також надано уявлення про перспективи розвитку та застосування віртуальних сховищ даних у майбутньому.

Ключові слова: віртуальне сховище даних, різноманітні дані, хмарна система зберігання даних.

SHAKHOVSKA NATALIYA, ZAGORODNIY IVAN, KOHUCH OKSANA, TRISKA ROMAN
Lviv Polytechnic National University

INTERFACES OF VIRTUAL DATA STORAGE IN THE CONDITIONS OF A FAST-CHANGING INFORMATION ENVIRONMENT

The article analyzes data storage architectures, explores the main approaches to cloud technologies, and highlights the features of virtual data storage architecture formation. In the process of analysis, positive and negative features of individual data storage architectures were highlighted. Cloud information technologies are considered from the point of view of their use for data storage and distribution. When considering cloud data storage, attention is focused on their architectural features from the point of view of efficiency and availability for users. Storage efficiency is an important characteristic of cloud storage infrastructure, especially given its emphasis on overall cost savings. Performance has many aspects, but the main task of a cloud storage system is to move data between the user and the remote cloud service provider. The system architecture was designed based on the proposed and modeled methods of organizing access to the cloud storage. According to the designed architecture, the data flows in the system were analyzed and investigated. The developed algorithm for the collection and integration of disparate data based on virtual data warehouses, in contrast to the Inmon and Kimball models, makes it possible to process NoSQL temporal (InfluxDB), graph (Neo4j) and document-oriented (MongoDB) data with a significantly lower value of capacity complexity and the possibility execution of recursive queries. The developed interface should be able to work with relational, temporal, graph and document-oriented data sources. To do this, it must provide the ability to perform SQL-like queries against relational databases, as well as specialized queries for each type of non-relational data. It allows to increase the efficiency of processing both relational and non-relational data, which makes it a powerful tool for analyzing and processing information from various sources. The development of virtual data warehouses may be associated with the introduction of new technologies, such as artificial intelligence, blockchain, quantum computing, etc. In addition, further improvements in metadata management and data security are expected.

Keywords: virtual data storage, various types of data, cloud data storage system.

Постановка проблеми

В останні роки обсяги даних швидко зростають, що ставить перед сучасними технологіями виклик створення ефективних методів зберігання та обробки цих даних. Одним з таких методів є віртуальні сховища даних (ВСД), які набули значного популярності через свою гнучкість та масштабованість.

Віртуальні сховища даних зазвичай складаються з трьох основних компонентів: віртуальний рівень доступу, фізичний рівень зберігання та управління метаданими. Віртуальний рівень забезпечує абстракцію від фізичних джерел даних та надає єдиний інтерфейс доступу до різноманітних джерел даних. Фізичний рівень відповідає за конкретну реалізацію зберігання даних, яка може включати в себе різноманітні технології, такі як хмарні сервіси, бази даних, файлові системи тощо. Управління метаданими відповідає за організацію та керування метаданими, що використовуються для ідентифікації та опису даних у сховищі.

Основними перевагами використання віртуальних сховищ даних є гнучкість, масштабованість та ефективність. Вони дозволяють легко масштабувати сховища даних в залежності від потреб користувачів, а також забезпечують швидкий доступ до даних через єдиний інтерфейс. Однак існують виклики, пов'язані з безпекою даних, управлінням версіями, забезпеченням конфіденційності тощо.

У сучасних віртуальних сховищах даних використовуються різноманітні технології, такі як віртуальні машини, контейнери, технології розподіленого зберігання даних тощо. Ці технології дозволяють підтримувати високу доступність, надійність та масштабованість віртуальних сховищ даних.

Аналіз останніх джерел

У сучасних віртуальних сховищах даних можна виділити кілька ключових характеристик: [1]

- Гнучкість та масштабованість: вони дозволяють адаптуватися до змінних потреб користувачів та обсягів даних.
- Єдина точка доступу: через віртуальний інтерфейс користувачам стає легше працювати з різноманітними джерелами даних.
- Управління метаданими: відповідає за ефективне організацію та пошук необхідної інформації.
- Безпека та конфіденційність: важливі аспекти у забезпеченні цілісності та захисту даних.

Для реалізації цих характеристик використовуються різноманітні технології, такі як хмарні сервіси, віртуальні машини, контейнери, технології розподіленого зберігання даних тощо.

Хмарна система зберігання даних, або зберігання даних як послуга – це абстрактне поняття, яке відповідає системі зберігання даних, яку можна адмініструвати за вимогою через спеціальний інтерфейс. Цей інтерфейс абстрагує місцезнаходження системи, так що локальна вона чи віддалена, або гібридна – не має значення. Хмарні інфраструктури зберігання даних утворюють нові архітектури, які підтримують різні рівні обслуговування поверх потенційно великої групи користувачів і географічно розподілених накопичувачів [1].

Ефективність зберігання даних – важлива характеристика хмарної інфраструктури зберігання, особливо враховуючи її акцент на загальну економію. Щоб зробити систему зберігання ефективнішою, потрібно зберігати більше даних. Загальним рішенням є скорочення обсягу вихідних даних, щоб вони займали менше фізичного простору. Два способи досягнення цієї мети: стиснення – упаковка даних шляхом їх кодування з використанням різних представлень – і дедуплікація – виключення всіх дублікатів даних. Хоча обидва методи корисні, стиснення передбачає обробку (перекодування даних в інфраструктуру і з неї), а дедуплікація – обчислення сигнатур для пошуку дублікатів.

Одна з найбільш особливостей хмарного зберігання даних - здатність забезпечити економію. Це економія на придбанні накопичувачів, їх енергопостачанні, ремонті, а також на управлінні зберіганням. Якщо розглядати хмарне зберігання з цієї точки зору (включаючи SLA і підвищену ефективності зберігання), воно може виявитися вигідним при певних моделях використання. API доступу до сховища є найважливішим компонентом послуг. Багато додатків вимагають доступу до сховища послуг з використанням API, який оптимізований для цієї конкретної системи зберігання даних, або на власному обладнанні або хмарним. Так хмарна система зберігання Amazon S3 API надає розробникам SDK для .Net і Java, а також бібліотеки для додаткових платформ і мов. Ці інтерфейси зазвичай використовують протоколи веб-сервісів передачі репрезентативного стану (REST) та/або простий протокол доступу до об'єктів (SOAP) [2].

Виділимо деякі характеристики сучасної архітектури віртуального зберігання даних. Ці характеристики не належать винятково до певного рівня, а можуть відповідати декільком (табл. 1).

Таблиця 1

Характеристики архітектури віртуального зберігання даних, побудовано за даними [3]

Характеристика	Опис
Керованість	Здатність керувати системою при наявності мінімальних ресурсів
Метод доступу	Протокол, через який надаються послуги Хмарного зберігання даних
Продуктивність	Вимірюється пропускну здатністю і часом затримки
Багатокористувацькість	Підтримка багатьох користувачів (арендаторів)
Масштабованість	Можливість поступового нарощування для задоволення нових вимог, або обробки підвищеного навантаження
Готовність даних	Вимірюється часом безвідмовної роботи системи
Управління	Можливість управляти системою – зокрема, вибираючи вартість, продуктивність, або інші характеристики
Ефективності зберігання	Міра ефективності використання накопичувачів
Вартість	Міра вартості зберігання даних (часто в доларах за гігабайт)

Однією з відмінностей між хмарною та традиційною системами зберігання є засоби доступу до них (рис. 1). Більшість постачальників пропонує різні методи доступу, однак загальноприйнятими є API Web-сервіс. Більшість з них реалізовані на принципах REST, основою якого є об'єктно-орієнтована схема, що розроблена поверх протоколу HTTP (з використанням TCP в якості транспорту).

Метою роботи є розроблення інтерфейсу віртуального сховища даних для опрацювання різнотипних джерел даних в умовах швидкозмінюваного інформаційного середовища. Розроблений інтерфейс має бути здатний працювати з реляційними, часовими, графовими та документ-орієнтованими джерелами даних. Для цього він повинен забезпечувати можливість виконання SQL-подібних запитів до реляційних баз даних, а також спеціалізованих запитів для кожного типу нереляційних даних.

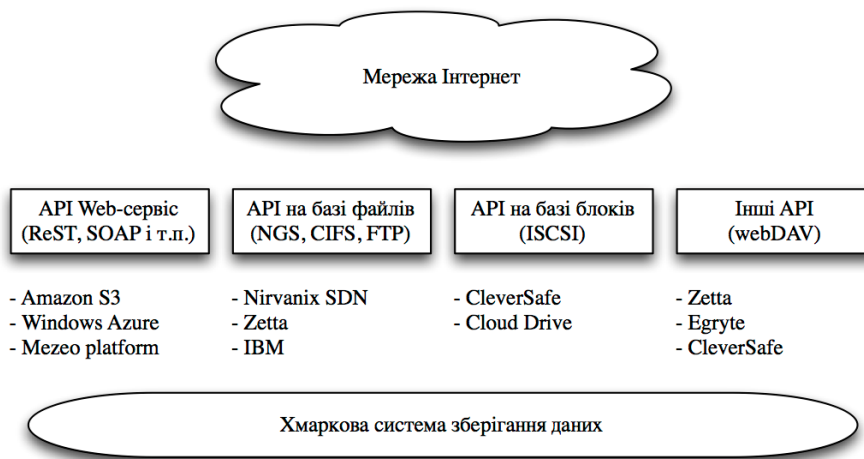


Рис. 1. Засоби доступу до хмарних систем зберігання даних, побудованих за даними [4]

Виклад основного матеріалу

Інтерфейс віртуального сховища даних включає:

Можливість вибору типу джерела даних (реляційні, часові, графові, документ-орієнтовані).

- Редактор запитів, який підтримуватиме SQL для реляційних баз даних та спеціалізовані мови запитів для нереляційних джерел.
- Можливість виконання рекурсивних запитів для обробки складних структур даних.
- Завантаження даних на сховище відбувається згідно з рис. 2. Отримавши IP адрес найближчого супутника, клієнта авторизується на сервісі і тільки після того, може отримати доступ до даних. Коли клієнту необхідно завантажити документ, він за допомогою програмного додатку надсилає документ на супутник з використання протоколу TCP. Супутник в свою чергу кешує дані та надсилає дані на сховище отримання напряму, або через найближче сховище з використанням протоколу UDT.

Доступ до даних демонструє рис. 3. У випадку необхідності отримання документу, клієнт за допомогою програмного додатку надсилає запит на отримання документу до супутнику з використання протоколу TCP. Супутник в свою чергу отримує документ зі сховища напряму або через інше сховище з використанням протоколу UDT. Документ тимчасово кешується на супутнику та надсилається клієнту (TCP).

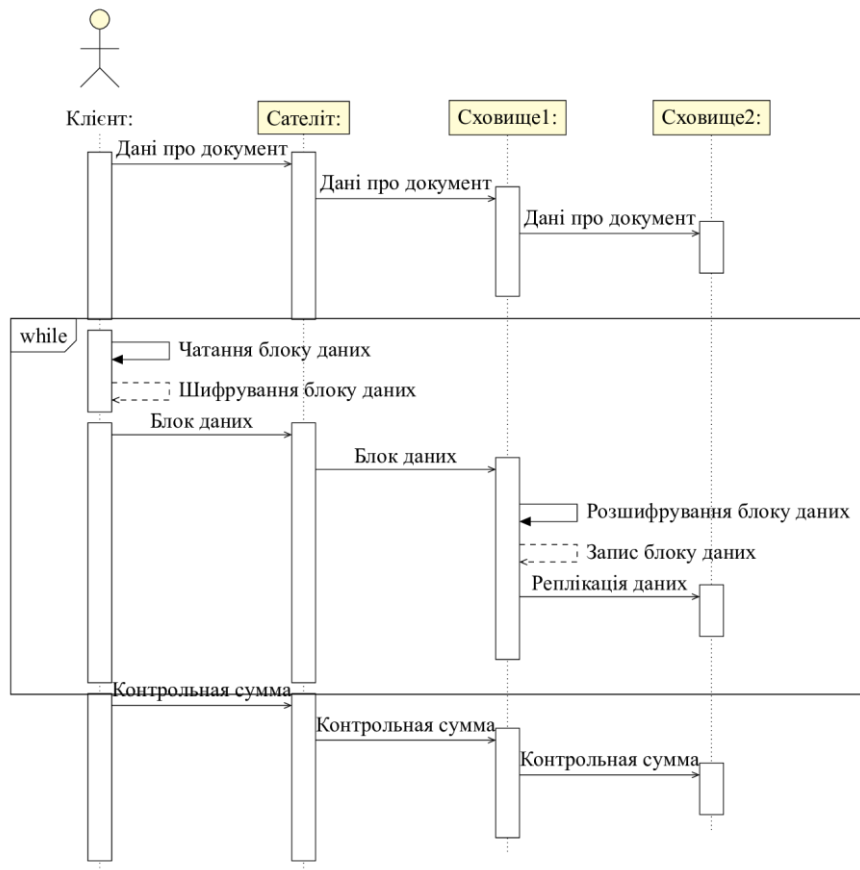


Рис. 2. Діаграма послідовності завантаження даних

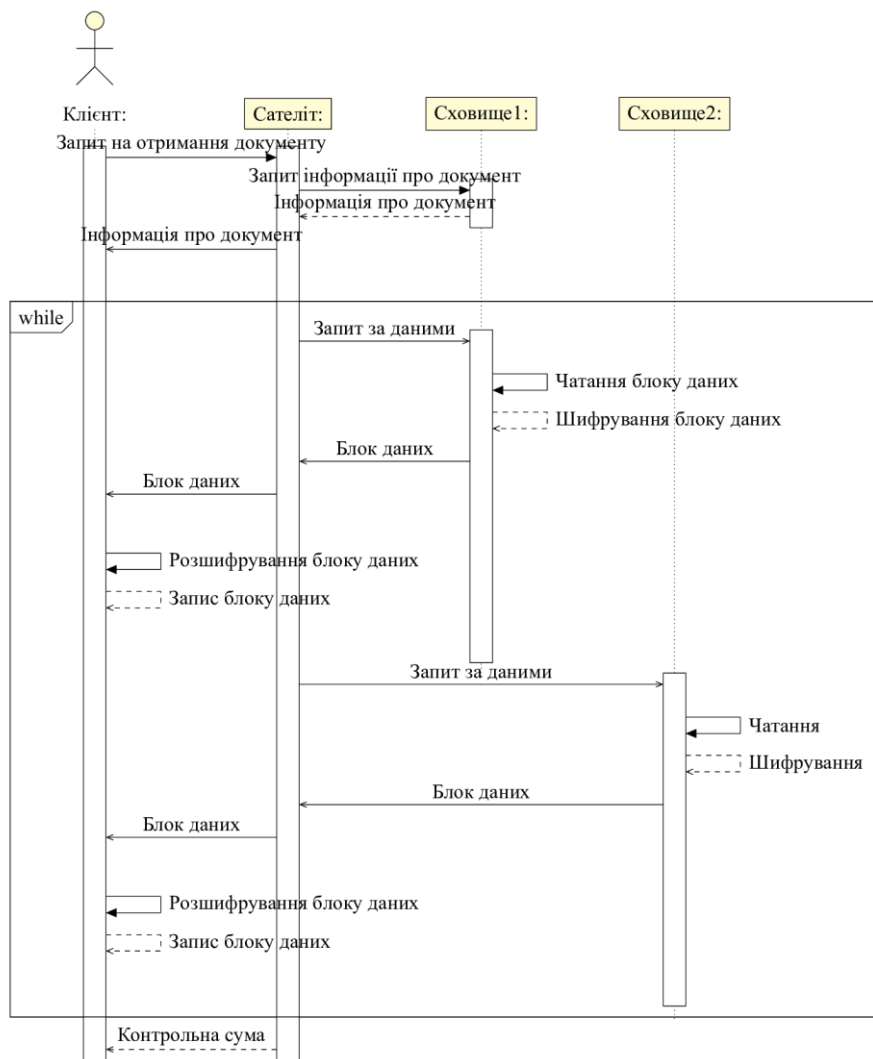


Рис. 3. Діаграма послідовності доступу до даних

Оскільки клієнтські дані зберігають у хмарному сховищі даних у вигляді файлової структури, передача даних в системі зводиться до передачі файлів. Передачу окремого файлу можна представити, як передачу окремих пакетів даних (рис. 4).

Під час передачі будь-якої кількості даних система формує пакети передачі даних та добавляє їх у чергу передачі даних. Після чого система виконує ряд функцій:

1. Перевірка на існування з'єднання з пунктом призначення даних.
2. Якщо з'єднання відсутнє:
 - створення з'єднання з пунктом призначення;
 - аутентифікація на пункті призначення;
3. Почергове шифрування па передачу пакетів даних (з черги) по даному каналу зв'язку.
4. У випадку заповнення вихідного кешу, система переходить до очікування інших подій

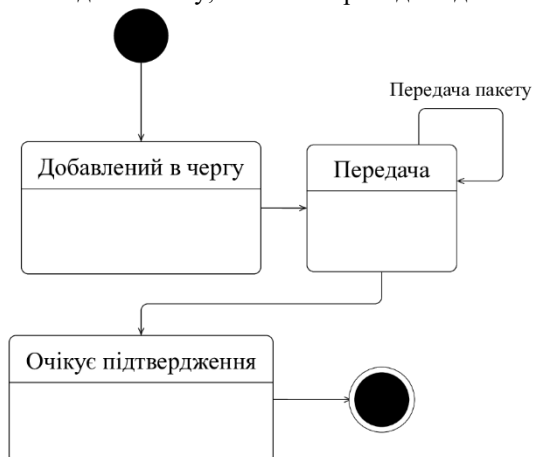


Рис. 4. Діаграма станів файлу для передачі

Логічну модель системи передачі даних в хмарковому сховищі даних можна представити у вигляді діаграми класів (рис. 5). З діаграми класів очевидно, що концептуальна модель системи передачі даних в Сховищі даних та Сателіті аналогічні. Даний підхід спрощує побудову систем такого характеру. Також з даної структури очевидно, що «З'єднання» є абстрактним класом та може використовуватися протоколами як TCP, або UDT, для транспортування даних. Такий підхід дає суттєві переваги у випадках існування єдиного обмеженого каналу зв'язку.

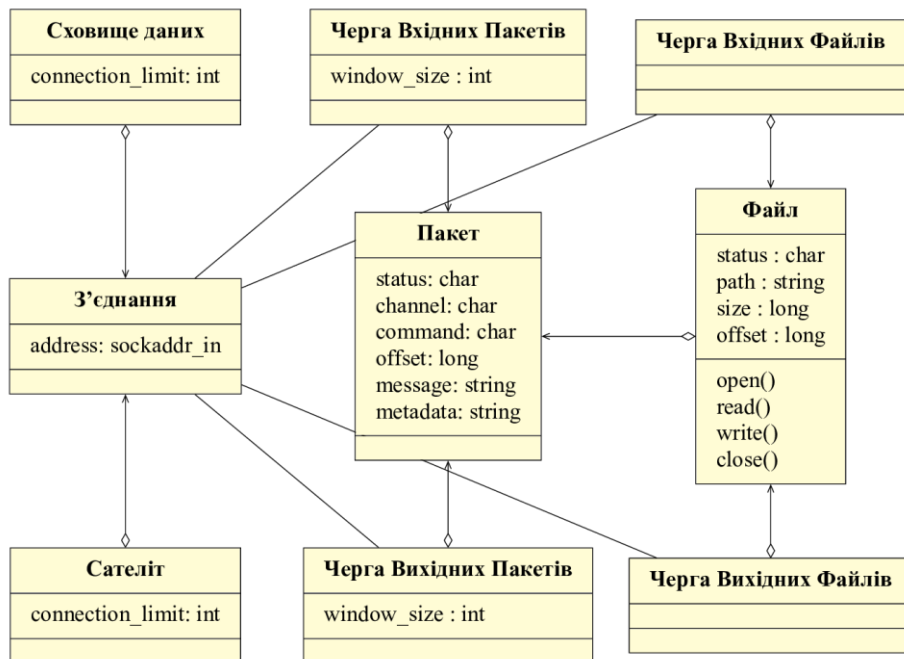


Рис. 5. Діаграма класів системи

Розроблений алгоритм колекціонування та інтеграції розрізнених даних на основі віртуальних сховищ даних, на відміну від моделей Інмона та Кімбола, дає змогу опрацювати NoSQL часові (InfluxDB), графові (Neo4j) та документ-орієнтовані (MongoDB) дані із значно меншим значенням емпіричної складності та можливістю виконання рекурсивних запитів.

Висновки

У статті здійснено аналіз архітектур зберігання даних, досліджено основні підходи до хмаркових технологій, виділено особливості формування архітектури віртуальних сховищ даних. У процесі аналізу виділено позитивні і негативні риси окремих архітектур зберігання даних. Розглянуто хмаркові інформаційні технології з точки зору їх використання для зберігання та поширення даних. При розгляді хмаркових сховищ даних зосереджена увага на їх архітектурних особливостях з точки зору ефективності та доступності для користувачів. Ефективність зберігання даних – важлива характеристика хмаркової інфраструктури зберігання, особливо враховуючи її акцент на загальну економію. Продуктивність має багато аспектів, але головне завдання хмаркової системи зберігання даних – переміщення даних між користувачем і віддаленим постачальником хмаркових послуг. Спроектано архітектуру системи на основі запропонованих і змодельованих методів організації доступу до хмаркового сховища. Відповідно до спроектованої архітектури проаналізовано та досліджено потоки даних в системі.

Розвиток віртуальних сховищ даних може бути пов'язаний з впровадженням нових технологій, таких як штучний інтелект, блокчейн, квантові обчислення тощо [5]. Крім того, очікується подальше удосконалення методів управління метаданими та забезпечення безпеки даних.

References

1. Huo Y., Wang H., Hu L., Yang H. A cloud storage architecture model for data-intensive applications. Computer and Management (CAMAN), 2011 International Conference on. IEEE. 2011. P. 1–4
2. Chun B. G., Dabek F., Haeberlen A. Efficient Replica Maintenance for Distributed Storage Systems. NSDI. 2006. № 6. С. 45–58.
3. Jones M. T. Anatomy of a cloud storage infrastructure. IBM developer works (November 30, 2010). 2010. <http://www.ibm.com/developerworks/cloud/library/cl-cloudstorage/cl-cloudstorage-pdf.pdf>.
4. Ghosh, P., Sadhu, D., & Sen, S. (2021). A real-time business analysis framework using virtual data warehouse. *Int. Arab J. Inf. Technol.*, 18(4), 585-595.
5. Asogwa, E., Amanze, B., Ngene, C., Belonwu, T., & Chukwuogo, O. (2022). Study on Theoretical Aspects of ontology-based and Virtual Data Integration in medical intelligence process and its Applications.