

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ОДЕСЬКА НАЦІОНАЛЬНА АКАДЕМІЯ ХАРЧОВИХ ТЕХНОЛОГІЙ  
ІНСТИТУТ КОМП'ЮТЕРНИХ СИСТЕМ І ТЕХНОЛОГІЙ  
«ІНДУСТРІЯ 4.0» ІМ. П.Н. ПЛАТОНОВА

**«ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ І  
АВТОМАТИЗАЦІЯ – 2021»**

***МАТЕРІАЛИ  
XIV МІЖНАРОДНОЇ НАУКОВО-ПРАКТИЧНОЇ КОНФЕРЕНЦІЇ***



21 - 22 ЖОВТНЯ 2021 р.

м.ОДЕСА

**MINISTRY OF EDUCATION AND SCIENCE OF UKRAINE  
ODESSA NATIONAL ACADEMY OF FOOD TECHNOLOGIES  
INSTITUTE OF COMPUTER SYSTEMS AND TECHNOLOGIES  
"INDUSTRY 4.0" NAMED AFTER P.N. ПЛАТОНОВА**

**«INFORMATION TECHNOLOGIES AND  
AUTOMATION– 2021»**

***PROCEEDINGS  
OF THE XIV INTERNATIONAL SCIENTIFIC AND PRACTICAL  
CONFERENCE***



**OCTOBER 21 - 22, 2021**

**ODESSA**

УДК 004.01/08

Інформаційні технології і автоматизація – 2021 / Матеріали XIV міжнародної науково-практичної конференції. Одеса, 21-22 жовтня 2021 р. - Одеса, Видавництво ОНАХТ, 2021 р. – 350 с.

Збірник включає матеріали доповідей учасників конференції, які об'єднані за тематичними напрямками конференції.

Збірник буде корисним як для фахівців і працівників фірм, зайнятих в області ІТ, так і для викладачів, магістрів і студентів вищих навчальних закладів, які навчаються за напрямками і спеціальностями програмного забезпечення обчислювальної техніки і автоматизованих систем, прикладної математики та обробки інформації, буде корисним професіоналам з комп'ютерного моделювання та розробки комп'ютерних ігор.

Результати досліджень у збірнику представляють собою своєрідний зріз сучасного стану справ в перерахованих галузях знань, який може допомогти як фахівцям, так і студентам університетів скласти загальну картину розвитку інформаційних технологій та пов'язаних з ними питань.

Наукові праці згруповані за напрямками роботи конференції та наведені в алфавітному порядку прізвищ авторів.

Матеріали (тези доповідей) друкуються в авторській редакції. Відповідальність за якість та зміст публікацій несе автор.

Рекомендовано для публікації Вченою Радою навчально-наукового інституту комп'ютерних систем і технологій «Індустрія 4.0» ім. П.М. Платонова від 23.09.2021 р., протокол № 2.

Матеріали подано українською, російською та англійською мовами.  
Редактор збірника Котлик С.В.

ТЕХНОЛОГИИ В СФЕРЕ ОБРАЗОВАНИЯ, Университет “Туран” (Казахстан)	
<b>Цінделіані Д.М., Ящук А.А., Повстяна Ю.С.</b> ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ GPS-МОНІТОРИНГУ, Луцький національний технічний університет (Україна)	167
<b>Яровий І.І., Ділова А.Є.</b> ТЕХНІЧНА ТВОРЧІСТЬ ЯК СКЛАДОВА ПРОЕКТНОГО СПОСОБУ НАВЧАННЯ, Механіко – технологічний фаховий коледж ОНТУ (Україна)	169
<b>Розділ 5. Проектування інформаційних систем та програмних комплексів</b>	172
<b>Ким В.Ю., Ким Е.Р.</b> ПРИМЕНЕНИЕ ИНФОРМАЦИОННЫХ СИСТЕМ В СФЕРЕ БУХГАЛТЕРСКОГО УЧЕТА И АУДИТА, Университет “Туран” (Казахстан)	172
<b>Liutenko I.V., Bieliaiev O.I.</b> DESIGN OF THE SOFTWARE FOR RETAIL INFORMATION SYSTEMS EFFECTIVENESS ASSESSMENT, NTU “KhPI” (Ukraine)	174
<b>Najdovski V., Manevska V.</b> FRAMEWORK FOR THE DEVELOPMENT OF AGRICULTURAL INFORMATION SYSTEMS, Faculty of Biotechnical Sciences, Faculty of Information and Communication Technologies, (Republic of North Macedonia)	177
<b>Антонова А.Р.<sup>1</sup>, Ільяшук Г.К.<sup>2</sup></b> АЛГОРИТМ ДОСЛІДЖЕННЯ ОПТИМІЗАЦІЇ СИСТЕМИ ОБРОБКИ ОНЛАЙН ЗАМОВЛЕНЬ, <sup>1</sup> Одеський національний технологічний університет, <sup>2</sup> Одеський державний екологічний університет (Україна)	179
<b>Афанасьєв Б.В., Зіноватна С.Л.</b> МОБІЛЬНИЙ ЗАСТОСУНОК ДЛЯ ФОРМУВАННЯ МНОЖИНИ МОТИВАЦІЙНИХ ЦИТАТ, Державний університет «Одеська політехніка» (Україна)	181
<b>Бабінчук О.О., Повстяна Ю.С.</b> ПРОЦЕС РОЗРОБКИ ІНФОРМАЦІЙНОЇ СИСТЕМИ ДЛЯ ОРГАНІЗАЦІЇ РОБОТИ АВТОТРАНСПОРТНОГО ПІДПРИЄМСТВА, Луцький національний технічний університет (Україна)	184
<b>Білик О.В.</b> ІНФОРМАЦІЙНА СИСТЕМА АВТОМАТИЗАЦІЇ ПОТОЧНОЇ РОБОТИ ТА ГОЛОСУВАННЯ «ВЧЕНА РАДА», Київський національний університет імені Тараса Шевченка (Україна)	185
<b>Богун Р.А., Селіванова А.В.</b> МЕТОДИ ТА ЗАСОБИ ПРОСУВАННЯ INSTAGRAM-АКАУНТУ, Одеська національна академія харчових технологій (Україна)	188
<b>Бондарчук В.К., Ліщинська Л.Б.</b> МЕТОДИ І ЗАСОБИ РОЗПОДІЛЕННЯ ДАНИХ МІЖ ХМАРНИМИ СХОВИЩАМИ, Вінницький національний технічний університет (Україна)	191
<b>Горборуков В.В.</b> СЕРВІС ОЦІНКИ ЕФЕКТИВНОСТІ РЕАБІЛІТАЦІЇ НА ОСНОВІ МКФ-ПРОФІЛЕЙ ПАЦІЄНТІВ, Національний університет «Києво-Могилянська академія» (Україна)	194
<b>Горбунов О.А., Щербина П.А.</b> АЛГОРИТМИ ПОБУДОВИ СКЕЛЕТОНУ ДЛЯ СИСТЕМИ РЕАБІЛІТАЦІЇ, Київський Національний Університет імені Тараса Шевченка (Україна)	196
<b>Гулевич О.О.</b> ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДІЯЛЬНОСТІ АВТОСАЛОНІВ, Університет державної фіскальної служби України (Україна)	197
<b>Дегтярьов Д.Ю., Ліщинська Л.Б.</b> ЗАСТОСУВАННЯ ПРОГРАМНИХ ЗАСОБІВ ДЛЯ ФОРМУВАННЯ ІНДИВІДУАЛЬНОЇ ПРОГРАМИ ТРЕНУВАНЬ, Вінницький національний технічний університет (Україна)	200
<b>Іванова Л.В., Котлик Д.О.</b> АВТОМАТИЗОВАНА ІНФОРМАЦІЙНА СИСТЕМА ОБЛІКУ ПІДВИЩЕННЯ КВАЛІФІКАЦІЇ ВИКЛАДАЧІВ, Відокремлений структурний підрозділ «Одеський технічний фаховий коледж ОНАХТ» (Україна)	202
<b>Каверинський В. В.</b> РОЗРОБЛЕННЯ ДІАЛОГОВОЇ ДОВІДКОВОЇ СИСТЕМИ	204

УДК 004.588

## МЕТОДИ І ЗАСОБИ РОЗПОДІЛЕННЯ ДАНИХ МІЖ ХМАРНИМИ СХОВИЩАМИ

**Бондарчук В.К., Ліщинська Л.Б.** (viacheslav.bondarchuk.k@gmail.com, llb@vntu.edu.ua)  
*Вінницький національний технічний університет (Україна)*

*Досліджено та проаналізовано основні методи та засоби розподілення даних між хмарними сховищами. Доведено необхідність розподілу великих об'ємних даних.*

### Вступ

На сьогодні, дані є основою цифрового світу. Дані підтримують усе, від відеороликів, до мільярдів фінансових операцій, які відбуваються щодня. В основі всього цього лежить розподілене зберігання.

На практиці існує багато видів розподілених сховищ даних. Зазвичай вони поставляються як послуги, якими керують хмарні провайдери, або продукти, які розгортаються самостійно. Також є можливість створити власне сховище.

Щоб по-справжньому зрозуміти, спочатку потрібно усвідомити масштаби і сутність даних сьогодні. Розглянемо деякі конкретні цифри [1]:

- Steam мав пік у 18,5 мільйонів одночасних користувачів, розгорнув сервери з 2,7 петабайтами твердотільного накопичувача та поставив 15 екзабайт користувачам у 2018 році;
- Nasdaq у 2020 році за один день зібрав максимум 113 мільярдів записів, збільшившись у середньому з 30 мільярдів лише двома роками раніше.

Кожен фрагмент даних ретельно зберігається та обробляється.

Одномашинні сховища даних не можуть задовільнити такі вимоги. Тому, замість цього використовуються розподілені сховища даних, які пропонують ключові переваги у продуктивності, масштабованості і надійності.

Продуктивність є критичним показником. Існує значна кількість досліджень [1], які кількісно визначають вплив затримок на бізнес. Повільний час реагування не просто засмучує людей - це коштує трафіку, продажів і, зрештою, доходу.

У випадку одномобільних сховищ даних часто досить перейти на більш швидку машину. Якщо цього недостатньо, то потрібно застосовувати інші форми масштабування.

### Результати дослідження

Дані — це інформація зібрана і трансформована для деяких цілей, зазвичай аналізу. Це може бути будь-який символ, текст, цифри, малюнки, звук або відео. Дані поза контекстом зазвичай не зрозумілі людині або комп'ютеру[1].

Усередині сховища комп'ютера, дані є набором чисел, поданих у вигляді байтів, які складаються з бітів, які мають значення один або нуль. Дані обробляються процесором, що використовують логічні операції для отримання нових даних з вихідних.

Системи для розподілу даних повинні відповідати таким вимогам:

- доступність;
- відмовостійкість.

Хоча доступність і відмовостійкість можуть виглядати подібними, але насправді вони різні.

Доступна, але не допускає помилок: система, яка виходить з ладу щохвилини, але відновлюється протягом мілісекунд. Користувачі можуть отримати доступ до послуги, але довготривалі завдання ніколи не мають достатньо часу на завершення.

Відмовостійка, але недоступна: система, де половина вузлів постійно перезавантажуються, а інші стабільні. Якщо ємність стабільних вузлів недостатня, то деякі запити доведеться відхилити.

Для розробника додатків ключовим моментом є те, що розподілені сховища даних можуть масштабувати продуктивність і надійність далеко за межі окремих машин. Але вони

мають застереження щодо того, як вони працюють, що може обмежити їх потенціал.

Існують такі види розподілу даних:

- за допомогою масштабування;
- за допомогою шардингу;
- за допомогою реплікації;

Існує багато способів розподілу даних, кожен з яких має свої компроміси. Два основні підходи - вертикальне і горизонтальне розділення.

Вертикальний розподіл - означає розділення даних за відповідними полями, вони можуть бути властивостями якогось загального об'єкта. Точний спосіб вертикального розподілу даних між машинами у кінцевому підсумку залежить від властивостей сховища даних і шаблонів використання комп'ютерів (також відомих як машини або вузли). Вертикальне масштабування означає зміну процесора машини, оперативної пам'яті, ємності пам'яті чи іншого обладнання[1].

Горизонтальне масштабування - це те, чому розподілені сховища даних можуть випереджати сховища даних для однієї машини. Поширюючи роботу на сотні комп'ютерів, сукупна система має більш високу продуктивність і надійність. Хоча розподілені сховища даних покладаються переважно на горизонтальне масштабування, вертикальне масштабування використовується разом для оптимізації загальної продуктивності і вартості. Масштабування існує у широкому спектрі від ручного до повністю керованого[1].

Горизонтальний розподіл (також відомий як шардінг) - це коли розподіляються дані на підмножини, усі з однією схемою. Наприклад, горизонтальне розділення таблиці реляційних баз даних, згрупувавши рядки у фрагменти, які будуть зберігатися на окремих машинах. Стратегії розподілу поділяються на дві категорії, алгоритмічну і динамічну, але існують гібриди.

Схема горизонтального і вертикального шардування наведена на рисунку 1.

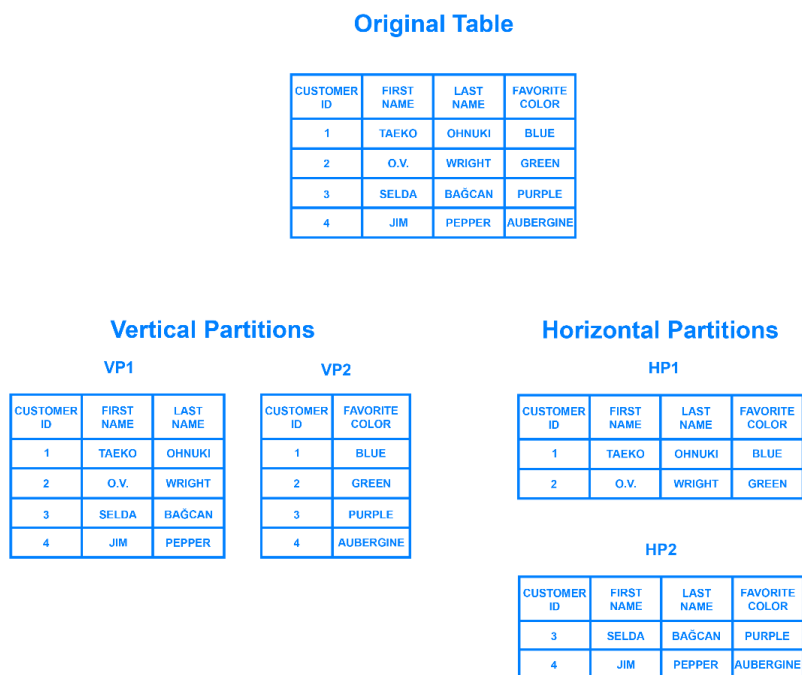


Рисунок 1 - Схема горизонтального та вертикального шардування

Алгоритмічний розподіл визначає, якому фрагменту розподілити дані на основі функції ключа даних. Наприклад, зберігаючи URL-адреси зіставлення даних "ключ-значення" до HTML, ми можемо розподіляти дані за допомогою розподілу ключів-значень відповідно до першої літери URL-адреси. Наприклад, усі URL -адреси, що починаються на "А", надходять на першому комп'ютері, "В" на другому, тощо. Існує незліченна кількість



стратегій з різними компромісами[1].

На практиці розподіл розділів є досить складним і може створити багато проблем, про які потрібно знати:

– частинки можуть мати нерівномірний розмір даних. Це поширене явище в алгоритмічному шардінгу, де важко отримати правильну функцію. Ми пом'якшуємо це, адаптуючи стратегію розподілу даних;

– осколки можуть мати гарячі точки, де певні дані запитуються за величиною частіше, ніж інші. Перерозподіл даних для обробки додавання або видалення вузлів із системи утруднений при збереженні високої доступності;

– індекси також потрібно розділити. Індекси можуть індексувати фрагмент, на якому він зберігається (локальний індекс), або він може індексувати весь набір даних і бути розділений (глобальний індекс). Кожен з них має компроміси;

– транзакції між розділами можуть працювати, або вони можуть бути відключені, повільні або непослідовні. Це особливо важко під час створення власного розподіленого сховища даних з одномобільних сховищ даних.

Динамічне шардування явно вибирає розташування даних і зберігає це місце в таблиці пошуку. Щоб отримати доступ до даних, потрібно звернутись до служби за допомогою таблиці пошуку або перевірити локальний кеш. Таблиці пошуку можуть бути досить великими, і, отже, вони можуть мати таблиці пошуку, що вказують на таблиці додаткового пошуку, наприклад, B+-деревх[1]. Структурна схема шардигу наведена на рисунку 2.

## Sharding structure

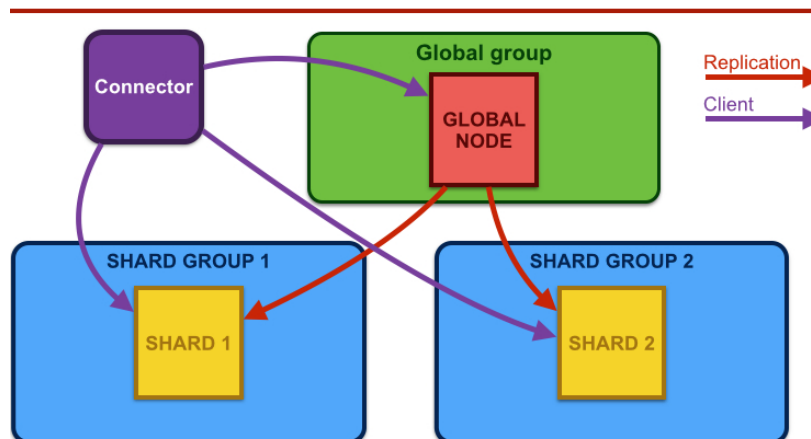


Рисунок 2 – Структура шардингу даних

Розбиття даних - це лише частина. Маршрутизація запитів може відбуватися на різних рівнях стека програмного забезпечення. А саме:

– розбиття на стороні клієнта - це коли клієнт тримає логіку прийняття рішення, до якого вузла запитувати. Перевагою є концептуальна простота, а недоліком - кожен клієнт повинен реалізувати логіку маршрутизації запитів;

– розбиття на основі проксі-це коли клієнт надсилає всі запити проксі. Потім цей проксі-сервер визначає, до якого вузла серверної частини слід звертатись. Це може допомогти зменшити кількість одночасних з'єднань на ваших серверних серверах та відокремити логіку програми від логіки маршрутизації;

– розбиття на основі сервера - це коли клієнт підключається до будь-якого серверного вузла, і вузол або обробляє, або перенаправляє, або пересилає запит;

Остання концепція - це реплікація. Реплікація означає зберігання кількох копій одних і тих же даних[1]:

– надмірність даних - коли апаратне забезпечення неминуче виходить з ладу, дані не втрачаються, оскільки є інша копія;

– доступність даних - клієнти можуть отримати доступ до даних з будь-якої копії. Це підвищує стійкість до перебоїв у роботі центру обробки даних та мережевих розділів;

– зменшення затримки мережі - клієнти можуть отримати доступ до найближчої до них репліки, зменшивши затримку мережі.

Реалізація реплікації вимагає неперевершених протоколів консенсусу та вичерпного аналізу сценаріїв збоїв. Реплікуючи дані близько один до одного, мінімізується затримка мережі при оновленні даних між машинами. Однак, копіюючи дані далі один від одного, створюється захист від збоїв центру обробки даних, розділів мережі та потенційно зменшується мережева затримка читання[1].

Отже, за результатами дослідження, проаналізовано основні методи і засоби розподілення даних між хмарними сховищами; доведено необхідність розподілу великих даних.

#### СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Introduction to Distributed Data Storage. URL: <https://towardsdatascience.com/introduction-to-distributed-data-storage-2ee03e02a11d>

УДК 004.681

#### **СЕРВІС ОЦІНКИ ЕФЕКТИВНОСТІ РЕАБІЛІТАЦІЇ НА ОСНОВІ МКФ-ПРОФІЛЕЙ ПАЦІЄНТІВ**

**Горбурков В. В.** ([slavon07@gmail.com](mailto:slavon07@gmail.com))

*Національний університет «Києво-Могилянська академія» (Україна)*

*Актуальною є проблема здійснення оцінки ефективності реабілітації пацієнтів з використанням міжнародної класифікації функціонування (МКФ). Створений програмний сервіс оцінки ефективності реабілітації дозволяє визначати якість проведеної реабілітації пацієнтів та проводити комплексну аналітику процесу їх відновлення.*

Впровадження нового підходу оцінки стану пацієнтів на основі їх МКФ-профілей дозволяє зробити прозорим процес реабілітації, що в свою чергу сприяє підвищенню якості медичної допомоги [1, 2]. Однак крім безпосереднього оцінювання по даним шкалам, необхідно визначати безпосередню спроможність пацієнта до відновлення та якість проведеної реабілітації.

Таким чином після формування МКФ-профілю стану пацієнта до та після реабілітації необхідно застосувати певні методики оцінювання, що включають в себе послідовну оцінку реабілітаційного потенціалу та оцінку ефективності реабілітаційних заходів [3]. Спосіб оцінки ефективності реабілітації пацієнтів визначається шляхом вимірювання показників порушень функцій організму, активності і участі пацієнта за допомогою стандартизованих оціночних шкал категорії «Міжнародної класифікації функціонування, обмеження життєдіяльності та здоров'я» (рис. 1).

Створений програмний сервіс оцінки реабілітаційного потенціалу пацієнта та реабілітаційного інтегрального показника після курсу реабілітації реалізує відповідну методику та дозволяє визначити ефективність проведеної реабілітації. Встановлення інтегрального реабілітаційного показника у пацієнтів в пізньому відновлювальному періоді відображає ступінь реалізації початкового реабілітаційного потенціалу. Таким чином низькі значення свідчать про те, що реабілітаційний потенціал не реалізований або реалізований частково, високі ж значення вихідної реабілітаційної оцінки свідчатимуть про те, що до



***XIV МІЖНАРОДНА НАУКОВО-ПРАКТИЧНА КОНФЕРЕНЦІЯ***

**«ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ І  
АВТОМАТИЗАЦІЯ – 2021»**

**21 - 22 ЖОВТНЯ 2021 р.  
м.Одеса**

***XIV INTERNATIONAL SCIENTIFIC AND PRACTICAL CONFERENCE***

**«INFORMATION TECHNOLOGIES AND  
AUTOMATION– 2021»**

**OCTOBER 21 - 22, 2021  
Odessa**

Збірник включає доповіді учасників конференції. Тези доповідей публікуються у вигляді, в якому вони були подані авторами.

Відповідальність за зміст і форму подачі матеріалу несуть автори статей.

The collection includes reports of conference participants. Abstracts are published in the form in which they were submitted by the authors.

The authors of the articles are responsible for the content and form of submission of the material.

**Редакційна колегія:** Котлик С.В., Корнієнко Ю.К., Плотніков В.М.

**Комп'ютерний набір і верстка:** Соколова О.П.

**Відповідальний за випуск:** Котлик С.В.