УДК 004.65

DOI: 10.31673/2412-9070.2022.031621

В. В. ЖЕБКА, доктор техн. наук, професор;
В. О. КОРЕЦЬКА, канд. пед. наук, доцент;
В. В. ТРОФИМЕНКО, студент;
К. О. ГОРДІЄНКО,
Державний університет телекомунікацій, Київ

ОЦІНЮВАННЯ МОЖЛИВОСТЕЙ БАЗИ ДАНИХ Google BigQuery ЯК АЛЬТЕРНАТИВИ MySQL

Розглянуто бази даних як центральну частину сучасної комп'ютерної системи. Ефективність роботи з інформацією забезпечується засобами системи керування базами даних. Це інтерфейс між кінцевим користувачем та програмою і, природно, самою базою даних, на якій виконуються завдання. Використання системи керування базами даних дає можливість створювати, оновлювати, шукати, видаляти і відновлювати дані в базах даних, а також визначати взаємозв'язки між її компонентами.

Аналіз останніх тенденцій розвитку IT-компаній свідчить про ефективність застосування хмарних технологій у процесі роботи з даними. Хмарні сервіси від Google пропонують революційні підходи в обробленні і зберіганні даних. Вони спростили доступ до даних, аналітики й обчислювальних потужностей, змінили уявлення щодо витрат, пов'язаних зі зберіганням даних.

На увагу заслуговує хмарне зберігання даних Google BigQuery, яке працює за serverless технологіями, забезпечуючи супершвидкість виконання SQL-запитів.

Проаналізовано функціональний інструментарій MySQL та Google BigQuery. MySQL є рішенням для малих і середніх застосунків, а Google BigQuery використовується для великих хмарних баз даних.

Наведено порівняння досліджуваних систем та зазначено можливий шлях імпорту даних із MySQL до Google BigQuery. Зроблено висновок про те, що можливості Google BigQuery можна розширити за допомогою низки сторонніх інструментів. Наприклад, інтегрувавши його з Google Таблиці, Microsoft Excel, QlikView, BIME Analytics та Microsoft Power BI.

Установлено, що перспективність застосування Google BigQuery полягає в розширенні можливостей сумісного використання даної бази даних з іншими програмними продуктами та оптимізація продуктивності запитів.

Ключові слова: база даних; система керування базами даних; MySQL; Google BigQuery; хмарні сервіси; великі дані.

Вступ

Обов'язковою частиною будь-якої сучасної комп'ютерної системи є база даних, саме в якій зберігається більшість даних. Такі бази даних є сховищами інформації, де вона зв'язана між собою, сортується та залишається незмінною структурно або напівструктурно, що робить її легкою для пошуку та доступною у використанні [1].

Ефективність роботи з інформацією забезпечується засобами системи керування базами даних (СКБД). СКБД — це інтерфейс між кінцевим користувачем та програмою, а отже, самою базою даних, на якій виконуються завдання. Використання системи керування базами даних дає можливість створювати, оновлювати, шукати, видаляти і відновлювати дані в базах даних, а також визначати взаємозв'язки між її компонентами (таблицями для реляційного типу). У різних галузях діяльності нагромаджено величезну кількість даних, що зумовлює посилення вимог стосовно їх оброблення та збереження, зокрема до продуктивності систем керування базами даними. Ця проблема особливо актуальна для даних, що потребують глибокого аналізу. З огляду на таку ситуацію з'являються нові підходи щодо побудови систем, які мають подолати недоліки наявних.

Останнім часом більшість IT-компаній намагаються перенести свою інфраструктуру від традиційних локальних (on-premise) рішень у хмару. Хмарні сервіси від Google пропонують революційні підходи до оброблення і збереження даних. Вони спростили доступ до даних, аналітики і обчислювальних потужностей, змінили уявлення про витрати, пов'язані зі зберіганням даних.

На увагу заслуговує хмарне зберігання даних Google BigQuery, яке працює за безсерверними (serverless) технологіями, що забезпечує супершвидкість виконання SQL-запитів.

Постановка проблеми. У процесі розроблення багатьох вебзастосунків або прикладних вирішень гостро постає питання вибору системи керування базами даних для виконання покладених на неї завдань. База даних має відповідати низці вимог, серед яких — надійність, розширюваність, продуктивність і здатність витримувати великі навантаження. У статті наведено дослідження, присвячене порівнянню можливостей Google BigQuery та MySQL у контексті мікросервісної архітектури та особливостей використання хмарної бази даних Google BigQuery за умов цифрової трансформації.

Мета і задачі дослідження. Метою дослідження є покращення процесу аналізу та вивчення даних за допомогою Google BigQuery як альтернативи MySQL. Для досягнення мети поставлено і реалізовано такі завдання:

• оцінити тенденції розвитку сучасних СКБД;

• дослідити функціональний інструментарій MySQL та Google BigQuery;

• порівняти переваги та недоліки досліджуваних баз даних;

• розглянути перспективи використання Google BigQuery.

Аналіз останніх досліджень і публікацій. Сьогодні у світі розроблено і застосовуються сотні різних СКБД. Центральним компонентом сучасної IT-інфраструктури є системи керування базами даних. Вони підтримують зберігання, використання та оброблення даних для різних застосунків.

Операційні системи керування базами даних, зокрема MySQL та багато інших, використовують для тимчасового зберігання даних для сучасних застосунків і забезпечують транзакції й інші типи взаємодій для бізнес-вимог на рівні як окремих сайтів, так і на рівні корпорацій.

Розроблена в 1993 році шведською компанією MySQL AB СКБД є програмним забезпеченням для керування інформацією з відкритим вихідним кодом. Останніми роками СКБД MySQL випередила конкурентів і стала найпопулярнішою системою у світі. Вона вважається найшвидшим, стабільним і зручним рішенням у сфері СКБД. Крім того, пропонує велику гнучкість: MySQL може використовуватися на більш як 20 платформах, включно з Linux, Windows і Mac OS. MySQL є основою для динамічних сайтів — блогів, картинних галерей або магазинів.

Сучасні IT-технології дедалі частіше звертаються до хмарних сервісів, зокрема під час оброблення і зберігання великих обсягів інформації. Актуальним є застосування хмарного сховища даних Google BigQuery.

BigQuery — це хмарний сервіс Google, призначений для роботи з Big Data, створений в 2011 році. Це онлайн-сховище даних, що дає змогу надійно зберігати і швидко обробляти великі масиви інформації без потреби застосовувати окремий сервер.

Google BigQuery є PaaS-сервісом («платформа як послуга»), який підтримує більшість функцій системи керування базами даних. Він є складовим елементом Google Cloud Platform, в якому репрезентовано ще кілька десятків застосунків для аналізу, зберігання і обчислення даних (рис. 1).

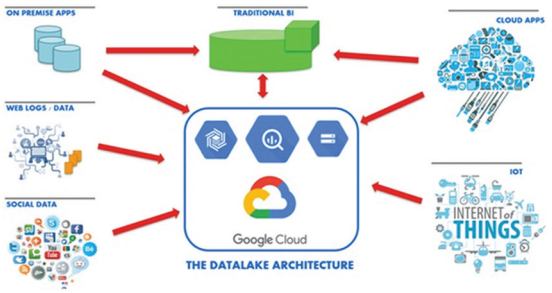

Рис. 1. Схематичне зображення головних сфер застосування Google Cloud

Основна частина

Бази даних організовують і керують ними на основі реляційної моделі мови SQL. Прикладні програми на SQL переважно є комбінаціями звичайних програм і операторів SQL. Програми взаємодіють із клієнтами, відображають дані і забезпечують високорівневий напрямок потоку даних. Така модель запропонована для збільшення продуктивності баз даних. Додатковою перевагою є незалежність даних методу оброблення запиту. Використання SQL дає можливість керувати базою даних у разі зміни логічних і фізичних схем. Паралельні системи баз даних мають пріоритет над традиційними, оскільки надають змогу оперувати з великими базами даних у режимі, що підтримує транзакції [2].

Значною перевагою MySQL перед схожими системами керування базами даних є більш висока швидкодія виконання запитів, яка досягається завдяки реалізації функціонала MySQL мовою C/C++, що вирізняється більшою швидкістю через свою низькорівневість і роботу безпосередньо з пам'яттю. Проте, не зважаючи на мову реалізації, бази даних, створені в середовищі MySQL, добре синхронізуються та взаємодіють із застосунками, розробленими мовою Java завдяки драйверу JDBC.

MySQL є рішенням для малих і середніх застосунків. Належить до складу серверів WAMP, LAMP і до портативних збірок серверів Denver, XAMPP. Зазвичай MySQL застосовують як сервер, до якого звертаються локальні або віддалені клієнти, проте в дистрибутив входить бібліотека внутрішнього сервера, даючи змогу вводити MySQL в автономні програми.

MySQL використовують у таких ситуаціях:

• у разі розподілених операцій, коли функціонала SQLite (інша популярна система) не вистачає;

• якщо потрібно забезпечити високий рівень безпеки, у чому MySQL досяг значного успіху;

• для роботи з інтернет-сторінками та вебзастосунками, оскільки MySQL є найбільш зручною СКБД для цієї сфери застосування;

• під час роботи зі специфічним проєктом, де функціонал MySQL дає оптимальний результат.

Технологія BigQuery кардинально змінила спосіб подання корпоративних даних. Будучи спочатку призначеною для роботи з гігантськими наборами даних, BigQuery стала однією з найкращих платформ для аналізу та вивчення даних (рис. 2). BigQuery — хмарне, безсерверне сховище корпоративних даних, яке допомагає розробляти, упроваджувати та масштабувати керування даними з інтелектуальних застосунків для цифрової трансформації [3].

Механізм запитів у Google BigQuery здатний виконувати запити SQL на терабайтах даних за лічені

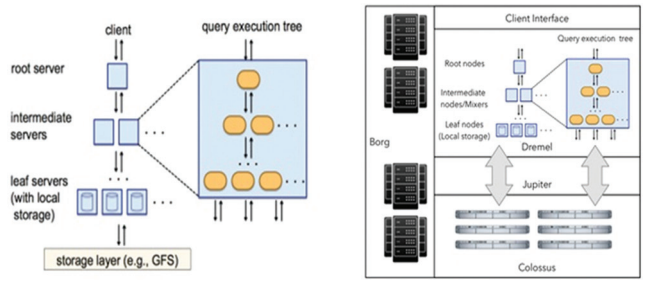

Рис. 2. Архітектура BigQuery

секунди, а на петабайтах — за лічені хвилини. Щоб дістати таку продуктивність, не доведеться організовувати та підтримувати будь-яку інфраструктуру і створювати або перебудовувати індекси.

BigQuery підтримує свій високоефективний формат колонкового зберігання, що робить методологію ETL особливо привабливою. Конвеєр оброблення даних, зазвичай реалізований на основі Apache Beam або Apache Spark, виймає потрібні вихідні дані (потокових даних або пакетних файлів), перетворює вилучені дані, готуючи їх до очищення або агрегування, а потім завантажує їх у BigQuery.

BigQuery може приймати як пакетні, так і потокові дані. Дані можна передавати до BigQuery безпосередньо через REST API. Часто користувачі, яким потрібно перетворити дані, наприклад здійснюючи обчислення з тимчасовим вікном, використовують конвеєри Apache Beam, які виконує сервіс Cloud Dataflow. І навіть під час передавання потокових даних до BigQuery можна запросити їх. Наявність загальної інфраструктури запитів для архівних (пакетних) та поточних (потокових) даних відкриває широкі можливості та спрощує багато процесів.

Дані BigQuery автоматично шифруються як під час зберігання, так і в процесі передавання. BigQuery піклується про безпеку розрахованих на багато користувачів запитів та ізоляції завдань. Є змога організувати спільний доступ до власних наборів даних за допомогою сервісу Google Cloud Identity and Access Management (IAM) та застосовувати до наборів даних (а також таблиць та подань у них) різні заходи безпеки, залежно від того, чи потрібна вам відкритість, можливість аудиту чи конфіденційність [4].

Ще одна перевага BigQuery в керуванні власним сховищем: можливість збільшення швидкості роботи без зусиль кінцевого користувача. Наприклад, удосконалення у форматах зберігання можуть автоматично застосовуватися до даних користувача. Аналогічно, поліпшення в інфраструктурі зберігання негайно відбиваються на роботі системи. Оскільки сховищем керує лише BigQuery, користувачам не потрібно турбуватися про резервне копіювання або реплікацію [5]. Усе, від оновлень та реплікації до резервного копіювання та відновлення, виконується системою керування сховищем автоматично. Однією з ключових переваг роботи зі структурованим сховищем на рівні абстракції таблиці (а не файла) та простого керування зберіганням цих таблиць є можливість для BigQuery підтримувати відповідні функції, наприклад, DML.

Розглянемо далі основні функції та можливості Google BigQuery.

• *Керування даними*. Сервіс дає змогу створювати і видаляти таблиці та функції користувача, а також імпортувати дані у форматах JSON, Avro, Parquet або CSV. Щоб використовувати дані в BigQuery, їх потрібно завантажити до сервісу Google Storage, а вже звідти провести імпорт даних через API. Також підтримується прямий імпорт та стримінг даних із Google Analytics.

• *Запити*. Запити Google BigQuery створюються через стандартний діалект SQL, а результат повертається в JSON-форматі. Стандартний розмір відповіді становить 128 Мб, але також він може бути і більшим (межа необмежена) у разі виставлення відповідних налаштувань.

• *Контроль доступу*. Користувачі сервісу можуть надавати стороннім особам публічний або обмежений доступ до своїх даних.

• *Машинне навчання*. Сервіс дає змогу створювати та запускати ML-моделі за допомогою SQL-запитів.

• *Інтеграція*. Сервіс можна використовувати як скрипт Google Apps Scripts або ж створений будь-якою іншою мовою, сумісною з REST API.

Порівнюючи переваги та недоліки досліджуваних баз даних MySQL і BigQuery можна зазначити таке:

• BigQuery — хмарний сервіс із високою швидкістю оброблення великих масивів даних;

• простота використання — у будь-якій іншій СКБД крім знання SQL доведеться довго розбиратися з тонкощами адміністрування та налаштуваннями бази;

• у BigQuery всю адміністративну частину на себе взяв Google. У цьому сервісі немає жодних налаштувань, індексів, двигунців таблиць, таймаутів чи зовнішніх ключів;

• доступність — вартість використання Google BigQuery залежить від обсягу завантажених у нього даних і становить 5$ за 1 Тб, що набагато дешевше за оренду сервера;

• у BigQuery реалізовано підтримання практично всіх функцій СКБД: віконні функції, зберігання даних як структур (нереляційні можливості), уявлення та табличні вирази (common table expression);

• BigQuery дає можливість використовувати такі важливі особливості, як властивості ACID (Atomicity, Consistency, Isolation, Durability —

атомарність, узгодженість, ізольованість, довговічність) транзакцій, а також автоматичну оптимізацію, завдяки чому користувачам не потрібно керувати файлами.

Серед недоліків сервісу BigQuery як СКБД можна виокремити неможливість підтримання рекурсивних запитів, створення збережених процедур та функцій, транзакцій. Донедавна в BigQuery були відсутні особливості, властиві сховищам даних, зокрема мова визначення даних (Data Definition Language, DDL; наприклад, оператор create) та мова маніпулювання даними (Data Manipulation Language, DML; наприклад, оператор INSERT).

Крім того, недоліком також є випадки значного зменшення швидкості роботи BigQuery.

Наведемо переваги MySQL (SQL) — реляційної бази даних SQL.

♦ Сумісність: MySQL доступний для всіх основних платформ, включно з Linux, Windows, Mac, BSD і Solaris. Він також має конектори з такими мовами, як Node. js, Ruby, C #, C ++, Java, Perl, Python і PHP, тобто він не обмежується мовою запитів SQL.

♦ Економічна ефективність: містить безкоштовні бази даних із відкритим вихідним кодом.

♦ Реплікаційна база даних: MySQL може бути репліковано на кілька вузлів, а отже, робоче навантаження може бути зменшено, а масштабованість і доступність застосунку можуть бути підвищенні.

♦ Подільність, хоча поділ неможливий у більшості баз даних SQL, це можливо зробити на серверах MySQL, що є вигідним.

Оцінюючи функціональні переваги MySQL, можна виокремити такі:

1. Використовуваний основний потік є багатопотоковим і підтримує кілька процесорів.

2. Існує кілька типів стовпців: 1, 2, 3, 4 та 8-байтове ціле число без знака / зі знаком, FLOAT, DOUBLE, CHAR, VARCHAR, TEXT, BLOB, DATE, TIME, DATETIME, TIMESTAMP, YEAR та Тип ENUM .

3. Реалізує бібліотеку функцій SQL через високооптимізовану бібліотеку класів і працює настільки швидко, наскільки це можливо, зазвичай після ініціалізації запиту не має бути виділення пам'яті. Жодних витоків пам'яті.

4. Повністю підтримує пропозиції SQL GROUP BY і ORDER BY та агреговані функції (COUNT(), COUNT(DISTINCT), AVG(), STD(), SUM(), MAX() та MIN()). Можна змішувати таблиці різних баз даних в одному запиті.

5. Підтримання LEFT 0UTER JOIN та ODBC ANSI SQL.

6. Усі стовпці мають значення за промовчанням. Можна використовувати INSERT для вставлення підмножини стовпця таблиці, а для тих стовпців, які не потрібно зазначати явно, встановлюються стандартні значення.

7. MySQL може працювати на різних платформах. Підтримання C, C++, Java, Perl, PHP, Python та TCL API.

Система MySQL має такі обмеження у своєму функціоналі, що не дає змоги використовувати її для роботи з програмами:

• недостатня надійність. У питаннях надійності певних процесів роботи з даними (наприклад, зв'язок, транзакції, аудит) MySQL поступається деяким іншим СКБД;

• низька швидкість розроблення. Як і багатьом іншим програмним продуктам із відкритим кодом, MySQL не вистачає деякої технічної досконалості, що часом позначається на ефективності процесів розроблення.

Оцінюючи функціональні недоліки MySQL можна зазначити такі:

1. Найбільшим недоліком MySQL є його система безпеки, яка переважно складна, а не стандартна. Крім того, вона змінюється лише тоді, коли mysqladmin викликається для перечитування дозволів користувачів.

2. Одним з інших серйозних недоліків MySQL є відсутність стандартного механізму RI (Referential Integrity); відсутність обмеження RI (обмеження фіксованого діапазону в заданому домені поля) може бути досягнуто за допомогою великої кількості типів даних.

3. MySQL не має мови збережених процедур, що є найбільшим обмеженням для програмістів, які звикли до баз даних корпоративного рівня.

4. MySQL не підтримує гаряче резервне копіювання.

5. Ціна MySQL залежить від платформи та методу встановлення. Linux MySQL є безкоштовним, якщо його встановлено користувачем або системним адміністратором, а не третьою особою, а третій план має сплачувати ліцензійний збір. Самостійне встановлення Unix або Linux безкоштовне, стороннє встановлення Unix або Linux коштує 200 доларів.

Перспективність використання Google BigQuery полягає в розширенні можливостей сумісного використання цієї бази даних з іншими програмними продуктами та оптимізація продуктивності запитів.

Час, що витрачається на виконання запиту, залежить від обсягу даних, які витягуються зі сховища, організації цих даних, кількості етапів, потреби в обробленні запиту, можливості розпаралелювання цих етапів, обсягу даних, оброблюваних кожним етапом, і обчислювальної дорожнечі кожного з етапів. Загалом простий запит, який читає три стовпці, буде виконуватися на 50% довше запиту, який читає лише два стовпці, оскільки

запит із трьома стовпцями має прочитати на 50% більше даних. Запит, що містить угруповання, зазвичай виконується повільніше, ніж запит без угруповання, адже операція угруповання додає додатковий етап оброблення запиту [6].

Існує можливість сумісної роботи досліджуваних систем. Оскільки BigQuery виступає як єдине місце збереження необроблених даних, то MySQL може виступати як шар кеша поверх нього і зберігати лише невеликі агреговані таблиці та надавати бажану відповідь за запитом. Проаналізуємо черговість дій під час імпорту з MySQL до BigQuery (рис. 3).

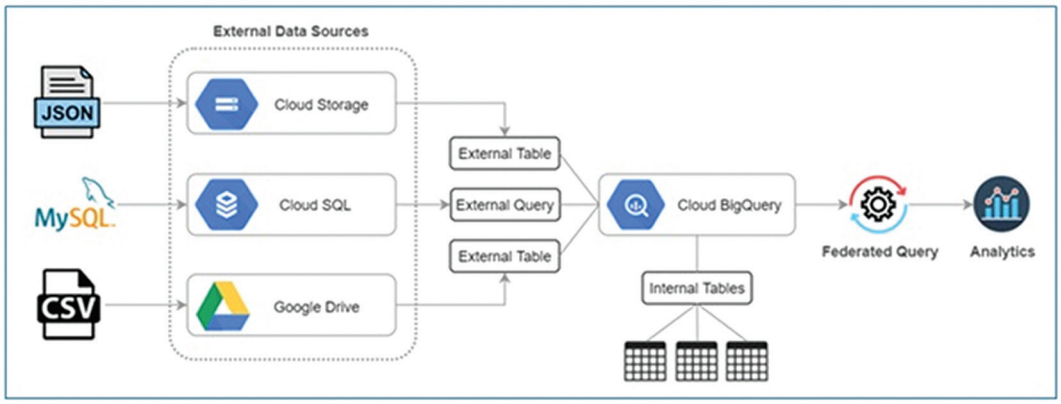

Рис. 3. Схематичне зображення процесу імпорту з MySQL до BigQuery

Крок 1. Експорт із MySQL.

Крок 2. Копіюємо набір даних у хмарне сховище за допомогою gsutil

gsutil rsync -m s3://xxxxx/tmp/my_table/ gs://xxxxx/tmp/my_table/

Крок 3. Завантажуємо дані до BigQuery за допомогою CLI.

Висновки

Отже, на основі аналізу функціонального інструментарію MySQL та Google BigQuery доходимо висновку про те, що MySQL є рішенням для малих і середніх застосунків, а Google BigQuery використовується для великих хмарних баз даних. Порівнюючи досліджувані системи та реалізацію ймовірного шляху імпорту даних із MySQL до Google BigQuery, можна зазначити, що здатність Google BigQuery можна розширити за допомогою низки сторонніх інструментів. Наприклад, інтегрувавши його з Google Таблиці, Microsoft Excel, QlikView, BIME Analytics та Microsoft Power BI.

Перспективність використання Google BigQuery полягає в розширенні можливостей сумісного використання даної бази даних з іншими програмними продуктами та оптимізації продуктивності запитів.

Різні впливові компанії, підприємці та ентузіасти Google позитивно використовують можливості Google BigQuery і високо оцінюють його функціональність та внесок у керування їхнім бізнесом. Наприклад, Twitter повідомив, що завдяки BigQuery вони змогли демократизувати свій аналіз даних та поділитися інформацією про компанію з широким колом внутрішніх груп. Група Alpega також спромоглася оптимізувати свої інновації за допомогою системи, використовуючи аналітику в реальному часі, яку їм раніше не вдавалося отримати.

Список використаної літератури

*1. **Васильєв О.** Програмування на С++ в прикладах і задачах: навч. посіб. Київ: Ліра-К, 2017. 258 с.*

*2. **Task** allocation in hybrid big data analytics for urban IoT applications / W. Ding, Z. Zhao, J. Wang, H. Li // ACM Transactions on Data Science. 2020. 1(3):1–22.*

*3. **MEFASD-BD:** multi-objective evolutionary fuzzy algorithm for subgroup discovery in big data environments-a mapreduce solution / F. Pulgar-Rubio, A. J. Rivera-Rivas, M. D. Pérez-Godoy [et al.] // Knowledge-Based Systems. 2017. 117:70–78.*

*4. **Patgiri R., Ahmed A.** Big data: The v's of the game changer paradigm // IEEE 18th international conference on high performance computing and communications; IEEE 14th international conference on smart city; IEEE 2nd international conference on data science and systems (HPCC/SmartCity/DSS). 2016. Piscataway: IEEE.*

*5. **Krocz K., Kizun O., Skublewska-Paszkowska M.** Perfomance analysis of relational databases MySQL, PostgreSQL, MariaDB and H2 // Journal of Computer Sciences Institute. (2020). 14. P. 1–7. URL: https://doi.org/10.35784/jcsi.1565.*

*6. **Пономаренко В. С., Мінухін С. В.** Методи та моделі розроблення комп'ютерних систем і мереж: монографія. Харків: Вид-во. ХНЕУ, 2016. 316 с.*

*7. **Universal** Method of Multidimensional Signal Formation for Any Multiplicity of Modulation / L. Berkman, L. Kriuchkova, V. Zhebka, S. Strelnikova // 5G Mobile Network Lecture Notes in Electrical Engineeringthis link is disabled. 2022, 831. C. 305–321.*

8. ***Protection*** *of telecommunication network from natural hazards of global warming / P. Anakhov, V. Zhebka, G. Grynkevych, A. Makarenko // Eastern-European // Journal of Enterprise Technologies. Kharkiv, 2020. 3(10 (105). P. 26–37.*

9. ***Удосконалення*** *інформаційної технології для підвищення функціональної стійкості мережі за допомогою теорії графів / В. О. Корецька, О. Ю. Ільїн, Є. О. Балашова [та ін.] // Телекомунікаційні та інформаційні технології. 2021. № 3 (72). С. 46–53.*

10. ***Лаврут О. О., Лаврут Т. В.*** *Модель та метод управління трафіком в мережах зв'язку критичного призначення. Prospects and priorities of research in science and technology: Collective monograph. Vol. 2. Riga, Latvia: Baltija Publishing, 2020. P. 36-60.*

V. V. Zhebka, V. O. Koretska, V. V. Trofymenko, K. O. Hordiienko

EVALUATION OF Google BigQuery DATABASE CAPABILITIES AS AN ALTERNATIVES TO MySQL

The article considers databases as a central part of a modern computer system. The efficiency of working with information is ensured by the means of database management system. This is the interface between the end user and the program, and of course, the database itself, on which the tasks are performed. Using a database management system allows you to create, update, search, delete and restore data in databases, as well as determine the relationships between its components.

Analysis of recent trends in IT companies indicates the effectiveness of cloud technology in working with data. Google's cloud services offer revolutionary approaches to data processing and storage. They have simplified access to data, analytics and computing power, and changed perceptions of storage costs.

Noteworthy is Google BigQuery cloud storage, which runs on serverless technology, which provides super speed of SQL queries.

The article presents an analysis of MySQL and Google BigQuery functional tools. MySQL is a solution for small and medium applications, and Google BigQuery is used for large cloud databases.

The comparison of the studied systems is given and the possible way of importing data from MySQL to Google BigQuery is indicated. It is concluded that the capabilities of Google BigQuery can be extended with a number of third-party tools. For example, integrating it with Google Spreadsheets, Microsoft Excel, QlikView, BIME Analytics and Microsoft Power BI.

It is established that the prospects of using Google BigQuery is to expand the ability to share this database with other software products and optimize query performance.

Keywords: database; database management system; MySQL; Google BigQuery; cloud services; big data.