

4. ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ



Науковий вісник НЛТУ України
Scientific Bulletin of UNFU

<http://nv.nltu.edu.ua>

<https://doi.org/10.15421/40290125>

Article received 29.01.2019 p.

Article accepted 28.02.2019 p.

УДК 621.311



ISSN 1994-7836 (print)
ISSN 2519-2477 (online)

@ ✉ Correspondence author

V. I. Roman

volodyaroman9@gmail.com

I. Г. Цмоць, В. І. Роман

Національний університет "Львівська політехніка", м. Львів, Україна

ВДОСКОНАЛЕННЯ МЕТОДУ ГРУПУВАННЯ ЕНЕРГЕТИЧНИХ ДАНИХ У СИСТЕМІ БАГАТОРІВНЕВОГО УПРАВЛІННЯ ЕНЕРГОЕФЕКТИВНІСТЮ ЕКОНОМІКИ РЕГІОНУ

Визначено основні причини повільного виконання операцій складного групування енергетичних даних у багаторівневих розподілених системах збереження даних, основні проблеми продуктивності методу MapReduce та методи його оптимізації. Розглянуто основні вимоги до системи багаторівневого управління енергоефективністю регіону, архітектуру та компоненти такої системи, запропоновано методи її вдосконалення. Розглянуто основні характеристики енергетичних даних, методи їх опрацювання, запропоновано оптимальну структуру збереження таких даних. Наведено основні механізми та методи швидкої вставки нових даних у систему та динамічного репартиціонування для забезпечення рівномірного розподілу даних у системі. Розроблено архітектуру системи збереження та опрацювання енергетичних даних. Розроблено метод партиціонування даних на основі методу MapReduce та структуру розподіленої системи збереження даних, яка забезпечує збереження даних з однаковими значеннями полів у одній партиції даних, тобто на одному комп'ютері. Цей метод партиціонування збирає в одну партицію записи з однаковими значеннями по полях, за якими будуть здійснюватися операції групування та агрегації. Такий підхід дає змогу зменшити час виконання та об'єми даних, які будуть транспортуватися по мережі під час виконання запитів групування та агрегації даних із використанням методу MapReduce. Також розроблена структура системи допомагає оптимізувати операції фільтрування даних.

Ключові слова: сховище даних; розподілені системи; агрегація даних; електроенергетика; MapReduce; BigData.

Вступ. На сьогодні аналіз і опрацювання Великих Даних (Big Data) у сфері електроенергетики є актуальною задачею, оскільки аналітика повних історичних масивів даних може дати точніші результати та глибше представлення природи даних, а також надати вказівки з оптимізації споживання енергоносіїв. До задач, що виконують у цій галузі належать задачі аналітики даних, візуалізації даних, машинного навчання та інтелектуального оброблення, опрацювання даних за допомогою OLAP підходу. Мова запитів SQL є одним із найпопулярніших та найпоширеніших методів аналітики даних, зокрема важливими функціями мови є групування та агрегація даних (за допомогою конструкції мови "GROUP BY"). Задачі групування та агрегація даних також відіграють центральну роль у підготовці даних до візуалізації у вигляді графіків та діаграм, а також і в задачах опрацювання даних за OLAP технологією.

Однією з основних характеристик BigData є власне об'єми даних, які неможливо або економічно не доцільно зберігати та опрацьовувати на одному чи кількох комп'ютерах. Тому основним підходом є використання високорозподілених кластерів, за якими рівномірно розподіляється масив даних. Зазвичай одиницею даних

виступає один запис, який зберігається системою на довільну комп'ютерах кластера. Під час задач групування та агрегації необхідно зібрати кілька записів із різних комп'ютерів кластера на одному комп'ютері для їх агрегації. Ця задача виконується MapReduce методом. Недоліком MapReduce методу є те, що під час агрегації великих даних виникає потреба активного переміщення даних по мережі для того, щоб зібрати необхідні групи даних на одному комп'ютері. Наслідком є значні часові затримки під час виконання агрегаційних запитів, які можуть бути недопустимі для систем реального часу або істотно уповільнювати роботу користувачів системи (HBase Rowkey Design, 2018; MapReduce, 2018).

Тому вдосконалення методу групування енергетичних даних у розподілених системах збереження є актуальною проблемою.

Аналіз останніх досліджень та публікацій. Архітектура багаторівневої системи управління енергоефективністю базується на основі розподілених систем із використанням хмарних сервісів і технологій Big Data. Одним із основних завдань цих систем є групування енергетичних даних для подальшого застосування (Tsmots, et al., 2018, 2017a, 2017b, 2013). Під групуван-

Інформація про авторів:

Цмоць Іван Григорович, д-р техн. наук, професор, завідувач кафедри автоматизованих систем управління.

Email: ivan.tsmots@gmail.com

Роман Володимир Ігорович, аспірант, кафедра автоматизованих систем управління. Email: volodyaroman9@gmail.com

Цитування за ДСТУ: Цмоць І. Г., Роман В. І. Вдосконалення методу групування енергетичних даних у системі багаторівневого управління енергоефективністю економіки регіону. Науковий вісник НЛТУ України. 2019, т. 29, № 1. С. 116–120.

Citation APA: Tsmots, I. H., & Roman, V. I. (2019). Improving Method of Energy Data Grouping in Multilayered System of Regional Energy Efficiency Management. *Scientific Bulletin of UNFU*, 29(1), 116–120. <https://doi.org/10.15421/40290125>

ням у статистиці розуміють поділ одиниць статистичної сукупності на групи, однорідні в якому-небудь істотному відношенні. Метод групування є основою для інших методів статистичного аналізу, що використовують для вивчення різних характеристик та особливостей даних. Групування поділяють на просте та складне залежно від кількості ознак (полів), що групуються. Складне групування відбувається за кількома ознаками. Складне групування буває комбінаційним та багатовимірним. Багатовимірне групування відбувається за комплексом ознак. Одними з найчастіше виконуваних задач є задачі багатовимірного групування, де ознаки містять дискретні (цілі) значення (Inmon, 2002; OLAP, 2018; MongoDB sharding, 2018).

У розподілених системах збереження даних вибірку даних поділяють на партиції (partitions, shards, segments). Партиція містить частину даних вибірки та розміщується на одному комп'ютері кластера (MongoDB sharding, 2018; Table partitioning strategies, 2018; Cassandra data model, 2018). Отже, дані розподіляються по всьому кластеру. У задачі партиціонування входить рівномірний розподіл даних за кластером та вставка нових даних у систему. Задача вставки пов'язана зі задачею пошуку рядка даних. У більшості систем вставка, тобто вибір партиції для рядка даних, здійснюється на основі головного ключа цього рядка, перетвореного через хеш-функцію (MongoDB sharding, 2018; Cassandra data model, 2018; HBase Rowkey Design, 2018). Хеш-функція забезпечує рівномірний розподіл даних за партиціями під час вставки, а також для швидкого пошуку рядка даних за його ключем. Недоліком цього підходу є цілком випадковий розподіл даних за кластером без урахування значень в інших ознаках даних. Задача репартиціонування полягає в поділі партиції на кілька партицій, коли вона містить більше даних за певний встановлений ліміт.

Для виконання операцій групування в системах збереження даних, де використовують партиціонування за допомогою головного ключа, перетвореного хеш-функцією, використовують MapReduce метод. Метод здійснює дві фази: map – опрацювання даних окремо на кожному комп'ютері, reduce – пересилання рядків з однаковими ключами, тобто ознаками, за якими здійснюється групування на одному комп'ютері. Оскільки партиціонування не враховує ознаки даних, за якими відбувається групування, то дані з однієї групи опиняються на різних комп'ютерах кластера. Відповідно недоліком підходу є затримка в часі обчислення і надмірне навантаження мережі кластера (MongoDB aggregation, 2018; Elasticsearch indexing, 2018).

Системи, що підтримують групування даних, використовують різні оптимізації. Одним зі способів оптимізації є використання індексів ознак, що групуються, наприклад як у системах MondoDB та Elasticsearch (Elasticsearch indexing, 2018; Elasticsearch aggregation, 2018; Druid Architecture, 2018). Але цей метод пришвидшує операції на одному комп'ютері, тобто map фазу, але не впливає на reduce фазу. OLAP система Druid використовує інший підхід партиціонування, що базується на основі поля часу та дає змогу здійснювати групування даних у часі та за іншими ознаками. Дані групуються тільки в межах одного комп'ютера, тому запити виконуються у реальному часі, тобто до секунд. Але недоліком системи є неможливість працювати не з часовими даними або групувати без ознаки часу (Druid Segments, 2018);

Data locality in Hadoop, 2018). Під час вирішення задач за допомогою MapReduce методу використовують Data Locality принцип. Він полягає у тому, що максимально можлива кількість операцій над даними має виконуватися у map фазі на тому комп'ютері, на якому знаходяться самі дані. Це допомагає максимально пришвидшити час виконання обчислень та максимально уникнути передачі даних по мережі для reduce фази (Data locality in Hadoop, 2018). In memory принцип полягає у триманні даних в оперативній пам'яті для пришвидшення доступу в разі звернення до них. Максимального пришвидшення групування за допомогою MapReduce методу можна досягнути з використанням Data Locality та in memory принципів.

Піж час партиціонування даних з урахуванням значень ознак можна забезпечити максимальну концентрацію груп даних за певними ознаками в одній партиції, тобто на одному комп'ютері, що дасть змогу за допомогою Data Locality принципу пришвидшити виконання MapReduce методу та мінімізувати переміщення даних у мережі кластера.

Під час вирішення задачі партиціонування з врахуванням ознак даних можна використати різноманітні структури даних для пришвидшення певних операцій, наприклад для визначення наявності певної групи ознаки в партиції можна використати Фільтр Блума (Bloom Filter), для приблизного, але швидкого визначення розміру партиції методу HyperLogLog, що може знадобитися у задачах вставки та репартиціонування (Data locality in Hadoop, 2018; Bloom filter, 2018).

Мета та завдання дослідження. Мета роботи – розроблення методу та структури розподіленої системи збереження даних, оптимізованої під швидке виконання задачі групування даних. Для досягнення мети необхідно виконати такі завдання:

- розробити архітектуру розподіленої системи збереження даних;
- розробити метод партиціонування даних з врахуванням значень ознак, за якими очікується групування;
- розробити метод швидкої вставки даних;
- розробити метод репартиціонування даних.

Основні результати дослідження. *Архітектура багаторівневої системи управління енергоефективністю економіки регіону.* Структура багаторівневої системи управління енергоефективністю (БСУЕ) економіки регіону є орієнтована на використання телекомунікаційних і Web технологій, баз даних, засобів збору, аналітичного та інтелектуального опрацювання енергетичних даних, візуалізації результатів опрацювання та прийняття управлінських рішень. Регіональна система моніторингу споживання енергоносіїв складається з двох підсистем: інформаційної та аналітичної. Структуру розробленої системи моніторингу споживання енергоносіїв зображено на рисунку, де TCP/IP – стек протоколу обміну, ETL (Extract, Transform, and Load) – виймання, перетворення та завантаження, OLAP (online analytical processing) – інтерактивна аналітичне оброблення, BI (Business Intelligence) – бізнесова аналітика, Data Mining – добування даних, N – кількість вузлів сенсорної мережі.

Технічною основою інформаційної підсистеми є безпроводні сенсорні мережі. Загалом бездротова сенсорна мережа – це розподілена система, яка складається з малогабаритних інтелектуальних (не завжди) сенсорних пристроїв, які пов'язані зі сховищем даних у хмарному сервері.

Сховище даних у хмарному сервісі має відкриті інтерфейси для доступу та публікації інтегрованих даних, управління якими здійснюється хмарним сервісом. Дос-

туп до цих даних може здійснювати будь-який користувач мережі Інтернет за допомогою публічних, документованих інтерфейсів доступу, які надає хмарний сервіс.



Рисунок. Структура регіональної системи моніторингу споживання енергоносіїв

Аналітична підсистема складається із сховища енергетичних даних, засобів ETL, OLAP, BI, Data Minig, прогнозування, візуалізації і підтримки прийняття рішень. Сховище енергетичних даних будується на основі Big Data технологій, які забезпечують можливість ефективно зберігати, опрацьовувати та надавати доступ до великих об'ємів даних у розподіленому режимі. Окрім цього, ці технології забезпечують ефективне опрацювання неструктурованих даних. Для реалізації і надання доступу до OLAP технології у сховищі використовують OLAP сервери. Користувачі можуть отримати доступ до цих засобів як напряму через інтерфейси, такі як через програмні додатки (десктопні, веб чи мобільні).

У БСУЕ економіки регіону використовують такі методи збирання енергетичних даних:

- *ручний збір* – внесення даних у систему користувачами;
- *інтерактивний* – внесення даних користувачем у інтерактивному режимі з використанням даних із статистичних збірників і сайтів (державних служб статистики України та Євростату, обліку енергоресурсів);
- *автоматичний* – автоматичний збір даних із використанням інтелектуальних сенсорів і технології інтернету речей;
- *напівавтоматичний* – поєднання автоматичного і ручного методів введення енергетичних даних.

З використанням інформаційно-аналітичної системи в БСУЕ економіки регіону опрацьовуються семантично неузгоджені дані, які надходять від систем управління енергоефективністю транспорту, промисловості та житлово-комунального господарства. Для такого опрацювання використовується інтеграція таких інформаційних технологій:

$$IT_{IAC} = \{IT_{DW}, IT_{WEB}, IT_{ITSM}, IT_{OLAP}, IT_{EDMS}, IT_{DM}, IT_{KDD}\},$$

де: IT_{IAC} – інформаційні технології інформаційно-аналітичної системи; IT_{DW} – технологія інформаційних сховищ (Data Warehouse); IT_{WEB} – WEB-технології; IT_{ITSM} – управління IT-послугами (IT Service Management); IT_{OLAP} – технологія оперативної аналітичного оброблення (OLAP – On-Line Analytical Processing); IT_{EDMS} – технологія автоматизації ділових процесів (EDMS – Enterprise

Document Management System); IT_{DM} – технологія інтелектуального аналізу даних (DM – Data Mining); IT_{KDD} – технологія, яка формує з даних нові нетривіальні знання у формі моделей, залежностей та законів (KDD – Knowledge Discovery in Databases).

Розроблення структури енергетичних даних і методу їх групування. Важливим етапом розробки засобів збереження даних у БСУЕ економіки регіону є визначення структури енергетичних даних. Для збереження енергетичних даних пропонуємо наступну структуру даних, яку наведено в таблиці.

Для збереження великих масивів даних використовують такі розподілені засоби збереження даних:

- реляційні бази даних. Зберігають структуровані дані, малоефективні для зберігання великих об'ємів даних на великих кластерах;
- NoSQL бази даних на основі key/value пар. Добре працюють з неструктурованими даними, можуть зберігати великі об'єми даних на кластерах, ефективні для операцій вставки і пошуку за ключем, але малоефективні для складніших операцій, зокрема фільтрування та групування;
- розподілені файлові системи. Добре працюють з будь-якими типами даних, операції над даними зазвичай мають значну затримку виконання;
- розподілені бази даних, що використовують індексування даних. Добре справляються зі задачами опрацювання даних, можуть мати затримки в разі частих оновлень існуючих даних або не дають змоги оновлювати існуючі дані. Розроблений метод групування даних бере за основу цей клас систем.

Для візуального сприйняття енергетичних даних вони можуть відображатися у такому вигляді: таблиць; діаграм; графіків. Передувати візуалізації енергетичних даних можуть операції фільтрування, сортування та групування за певними атрибутами, які здійснюється в інтерактивному режимі через графічний інтерфейс або за допомогою спеціальних мов запитів, наприклад SQL. Під час візуалізації може використовуватися загальна статистика в певному зрізі або залежність двох або трьох атрибутів у часі.

Таблиця. Структура енергетичних даних

Атрибут	Опис
Адміністративна одиниця	атрибут представляє географічний розподіл даних, наприклад: місто, район, мікрорайон
Споживаний вид енергоресурсів	вид енергоресурсу: газ, електроенергія, бензин, тверде паливо, дизельне паливо, вугілля та інше
Величина вимірюваного параметру	кількість споживаного енергоресурсу в одиницях вимірювання
Одиниця вимірювання	куб, ват, літр, тонна та інше
Ідентифікатор початальника даних	унікальний ідентифікатор мікроконтролера чи лічильника
Джерело даних	ідентифікатор джерела даних: інтелектуальні сенсори, сайти, статистичні збірники, ручне внесення та інше;
Споживання ресурсу за видом діяльності	вид діяльності: транспорт, житлово-комунальне господарство, промисловість та інше
Час	часова характеристика

Для вивчення різних характеристик та особливостей енергетичних даних із допомогою статистичного аналізу необхідно виконати їх групування. Тому актуальною задачею є пришвидшення процесу групування енергетичних даних у розподіленій системі збереження, яке досягається формуванням партицій з максимальною концентрацією однакових груп за певними ознаками. Для досягнення цього розроблений метод розподілу нових даних за партиціями з подальшим їх точковим пошуком. Метод розробляється з урахуванням того, що дані будуть незмінні в часі та буде здійснюватися постійна вставка нових даних. Окрім цього, передбачає, що під час створення таблиці даних задаються назви полів (ознак), за якими буде відбуватися групування та партиціонування. Такі поля будемо називати ключовими полями. Для підтримки методу обрано master-slave архітектуру системи. Master компонент містить метадані про партиції та поля. Для кожного ключового поля створюється асоціативний масив (наприклад hash-map), що складається з набору key/value (ключ/значення) пар, де key містить значення поля, а value містить масив індексів партицій, у яких присутні записи з таким значенням у цьому ключовому полі.

Для внесення нових даних використовують метод вставки (вибору партиції), який працює так:

- запит на вставку нового запису здійснюється через мастер;
- на основі значень ключових полів з асоціативних масивів мастера (метаданих) отримуються індекси партиції, на яких є значення цих полів;
- з використанням отриманих індексів вибирається партиція, на яку буде відправлений запис;
- вибір партиції здійснюється з найбільшою кількістю значення ключових полів;
- мастер відправляє запис на обрану партицію та за потреби модифікує асоціативні масиви.

Пошук та фільтрація даних виконуються так:

- запит на пошук даних за певними ключовими полями приходить на мастер;
- з використанням асоціативних карт вибираються партиції, на яких знаходяться шукані значення;
- запит посилається на машини, що містять ці партиції;
- результати даних можуть відсилатися до клієнта напряму або через мастера, якщо це необхідно.

Для репартиціонування даних кожна партиція містить метадані. Для кожного ключового поля створюється асоціативний список. Ключем списку є значен-

ня поля, а значенням є лічильник, що показує скільки разів це значення трапляється в цьому полі в межах партиції. Цю структуру використовують для поділу великої партиції на дві менші та приблизно однакові за розміром партиції за методом, зазначеним нижче. У разі потреби одну з партицій можна перенести на інший комп'ютер кластера для балансування навантаження на кластер. Метод репартиціонування запускається, коли кількість рядків у партиції стає більшою, ніж певне задане значення ліміту розміру партиції.

Метод репартиціонування можна описати такими кроками:

1. Вибирається поле з найменшою кількістю унікальних значень.
2. Розділяємо його за значеннями на дві частини так, щоб вони мали максимально однакову кількість записів. Наприклад, поданий такий асоціативний масив партиції: {A→10, B→20, C→5, D→6}. Оптимальним способом поділу на дві партиції буде: {B→20} та {A→10, C→5, D→6}. Введемо дві змінні, що відповідають кількості записів у нових партиціях: $S_1 = 20$, $S_2 = 21$.
3. Якщо $S_1 - S_2 < T$, де T – це максимальне значення різниці, задане адміністратором системи, то йдемо до кроку 4, інакше якщо є нерозглянуті поля, то вибираємо наступне найменше за кількістю унікальних значень поле і йдемо на крок 2. Якщо всі поля розглянуто, то вибираємо одну з стратегій, описану нижче.
4. Фізично розділяємо партицію на дві, будуємо для них метадані, надсилаємо дані для оновлення метаданих мастера.

У випадку якщо всі ключові поля були протестовані на оптимальність у зазначеному вище методі, але оптимальне поле поділу не знайдено, то можна застосувати одну з таких стратегій:

1. Вибрати оптимальне поле для поділу, таке, де $S_1 - S_2 \rightarrow \text{MIN}$. Цей підхід дає змогу зберегти цільову групу в партиції, крайньою ситуацією є, коли у партиції залишиться тільки одне значення групи для поля. У такому разі подальше використання цієї стратегії неможливе, необхідно використати стратегію 3. Така ситуація буде виникати часто, якщо в систему інтенсивно вставляються дані з цим значенням у полі. Однак використання стратегії не суперечить наявності кількох шардів з цією групою для забезпечення дистрибуції даних та обчислень. Тому це рекомендована стратегія. У випадку неможливості її використання будемо використовувати стратегію 3.
2. Повторити метод з поділом на три або більше партицій. Використання цієї стратегії має сенс, коли в полях є більше трьох унікальних значень, наприклад {A→31, B→30, C→32}, поділ на три партиції у цьому випадку буде давати кращий розподіл між партиціями. Рекомендовано використовувати стратегію в окремих випадках, наприклад зазначених у цьому прикладі, тобто для шардів з трьома рівномірно розподіленими групами.
3. Поділ однієї з груп на дві варто використовувати коли неможливо застосувати стратегію 1. У разі розподілу групи за двома шардами потрібно мінімізувати розділ груп в інших ключових полях.

Описані методи вставки даних та репартиціонування забезпечують збір груп на одній чи кількох машинах кластера. Під час виконання операцій групування методом MapReduce можна вдосконалити та пришвидшити за допомогою:

1. Використання Data Locality підходу. Тобто за рахунок збереження всіх даних групи на одній чи кількох ма-

шинах більша частина операцій групування виконається на map фазі, що пришвидшить час виконання запиту та зменшить об'єми даних, що передаються по мережі.

2. Мастер машина на основі метаданих може передавати на робочі машини кластера разом зі запитом додаткову інформацію про те, у яких партиціях знаходиться група. MapReduce метод може використати цю інформацію для оптимізації reduce фази, тобто передавати дані тільки між тими парами машин, які містять у себе дані, що лежать в одній групі. Також reduce процес може створюватися саме на одній з цих машин для запобігання передачі його даних у мережі до випадково вибраної машини з reduce процесом.

Також пошук даних та фільтрування можна комбінувати разом із запитом на групування даних. Первинно відбувається фільтрація даних на мастері, тобто за допомогою асоціативних масивів вибираються тільки ті партиції, які містять дані зі значеннями, зазначеними в запиті. Запит відсилається мастером не на весь кластер, а тільки на машини, що містять ці партиції.

Висновки:

1. Розроблено метод та архітектуру розподіленої системи збереження енергетичних даних, оптимізовану під швидке виконання задачі групування даних методом MapReduce з використанням принципу Data Locality. Мінімізовано об'єми даних, що необхідно передавати по мережі на reduce фазі. Обрано master-slave архітектуру. Забезпечено підхід партиціонування даних з урахуванням значень ознак, за якими очікується групування або фільтрація.
2. Розроблено метод швидкої вставки та пошуку даних у розподіленій системі з врахуванням партиціонування за різними ознаками. Пошук даних можна здійснювати як і за ключовими так і за по ключовими ознаками даних.
3. Розроблено метод репартиціонування даних під час переповнення розмірів партиції. Враховано розмір нових партицій, перебудову метаданих та розбивку партиції на основі ознак, за якими очікується групування.

Перелік використаних джерел

- Bloom filter. (2018). Retrieved from: https://en.wikipedia.org/wiki/Bloom_filter
- Cassandra data model. (2018). Retrieved from: <https://shermadigital.com/blog/designing-a-cassandra-data-model/>

- Data locality in Hadoop. (2018). Retrieved from: <https://data-flair.training/blogs/data-locality-in-hadoop-mapreduce/>
- Druid Architecture. (2018). Retrieved from: <https://dev.to/rusrushal13/learning-about-the-druid-architecture-184c>
- Druid Segments. (2018). Retrieved from: <http://druid.io/docs/latest/design/segments.html>
- Elasticsearch aggregation. (2018). Retrieved from: <https://www.elastic.co/guide/en/elasticsearch/reference/6.4/search-aggregations.html>
- Elasticsearch indexing. (2018). Retrieved from: <https://www.elastic.co/guide/en/elasticsearch/guide/current/index-doc.html>
- HBase Rowkey Design. (2018). Retrieved from: <http://hbase.apache.org/0.94/book/rowkey.design.html>
- HyperLogLog. (2018). Retrieved from: <https://en.wikipedia.org/wiki/HyperLogLog>
- Inmon, W. H. (2002). *Building the Data Warehouse*. (4th ed.). Wiley Computer Publishing.
- MapReduce. (2018). Retrieved from: <https://en.wikipedia.org/wiki/MapReduce>
- MongoDB aggregation. (2018). Retrieved from: <https://docs.mongodb.com/manual/aggregation/>
- MongoDB sharding. (2018). Retrieved from: <https://docs.mongodb.com/manual/sharding/>
- OLAP. (2018). Retrieved from: https://en.wikipedia.org/wiki/Online_analytical_processing
- Pivniak, H. H., Vypanasenko, S. I., Khovanska, O. I., Khatskevych, Yu. V., & Dreshpak, N. S. (2013). *Systemy enerhomenedzhmentu ta yikh matematychnye zabezpechennia*. Dnipropetrovsk: NHU. [In Ukrainian].
- Table partitioning strategies. (2018). Retrieved from: <https://docs.oracle.com/database/121/VLDBG/GUID-6CE884AF-84A4-4E6A-A3EF-DCCEBCAB2DB2.htm#VLDBG00202>
- Tesliuk, T. V., Tsmots, I. H., Opotiak, Yu. V., & Tesliuk, V. M. (2017a). Arkhitektura bahatorivnevoi systemy upravlinnia enerhoefektyvnosti rehionu. *Bulletin of the National University "Lviv Polytechnic". Series: Computer Science and Information Technology*, 864, 201–209. [In Ukrainian].
- Teslyuk, T., Tsmots, I. G., Teslyuk, V., Medykovskyy, M., & Opotyak, Y. (2017b). Architecture and Models for System-Level Computer-Aided Design of the Management System of Energy Efficiency of Technological Processes at the Enterprise. *Automation 2017. International Conference Automation 2017. Advances in Intelligent Systems and Computing*, 689, 538–557.
- Tsmots, I. H., Tsymbal, Yu. V., Roman, V. I., & Sydorenko, R. V. (2018). Zasoby zberezhenia, opratsiuvannia ta zakhystu danykh bahatorivnevoi systemy upravlinnia enerhoefektyvnosti rehionu. *Scientific Bulletin of UNFU*, 28(2), 137–143. <https://doi.org/10.15421/40280226>

I. H. Tsmots, V. I. Roman

Lviv Polytechnic National University, Lviv, Ukraine

IMPROVING METHOD OF ENERGY DATA GROUPING IN MULTILAYERED SYSTEM OF REGIONAL ENERGY EFFICIENCY MANAGEMENT

The authors have defined the main reasons of performance issues with grouping operations based on MapReduce algorithm in multilayered energy management systems. Defined main requirements for such systems, designed architecture of data warehouse component, methods of fast data insertion, data repartitioning, even distribution of data in such systems. Authors defined energy data schema, energy data specifics and structure, also common operations performed with energy data. Authors have developed optimization method for MapReduce algorithm in grouping operations on distributed data storage system, architecture of energy data management system and partitioning strategy. Developed method uses data partitioning based on fields that are expected to be grouped, data with same fields values are gathered on same cluster nodes. This provides data locality principle and reduces data transfer by networks during grouping execution. Also such partitioning allows to fast data insert and search by its primary key, and improves filtering operation performance in some cases. Designed storage architecture uses master-slave architecture. Master node has metadata information about partitions, specifically sets of fields values which are contained in partition. Based on this information new data records are inserted into partition which is most suitable and contains maximum of other records with identical values in fields. Alongside partitioning algorithm uses information about primary key and ranges of primary keys that are managed by specific partition, this information is also stored on master node. This allows fast data search by primary key. Authors have defined different strategies of repartitioning to prevent cases when single partition becomes too big and hardly manageable by single computer. In this case repartitioning strategy breaks single partition to few parts. Defined strategies are used to break partition and save data locality optimization. Defined partitioning strategy for grouping operation using MapReduce algorithm provides that the maximum number of records from single group are concentrated in single or few partitions only. This means that we can do most of grouping locally on map phase and reduce number of computation and network traffic on shuffle and reduce phases.

Keywords: data warehouse; distributed systems; data aggregation; electricity; MapReduce; BigData.