PostgreSQL Data Warehouse Implementation and Performance Optimization For Energy Companies*

* Published: https://ieeexplore.ieee.org/abstract/document/10553614
Note: (Sub-titles are not captured in Xplore and should not be used)

Dakota Joiner
Computer Science
Okanagan College
Kelowna, Canada
0000-0002-3094-0015

Mathias Clement
Computer Science
Okanagan College
Kelowna, Canada
0000-0001-8206-307X

Keegan Pereira
Computer Science
Okanagan College
Kelowna, Canada
0000-0002-2893-3406

Shek (Tom) Chan
Mathematics and Statistics
Langara College
Vancouver, Canada
0000-0001-6143-7175

Youry Khmelevsky
Computer Science
Okanagan College
Kelowna, Canada
0000-0002-6837-3490

Albert Wong
Mathematics and Statistics
Langara College
Vancouver, Canada
0000-0002-0669-4352

Joe Mahony
Research and Development
Harris SmartWorks
Ottawa, Canada
JMahony@harriscomputer.com

Michael Ferri
Research and Development
Harris SmartWorks
Ottawa, Canada
mferri@harriscomputer.com

Abstract—With smart grids replacing traditional energy grids in recent years, energy companies are facing a new challenge to efficiently manage and analyze the collected data. The use of sensors in smart grids has led to a surge in the data being collected. Due to its design, the classic relational Online Transaction Processing (OLTP) relational database management system (RDBMS) starts to become inefficient for queries against the RDBMS beyond a terabyte size. For data extraction efficiency on large volumes of data, OLAP along with data warehousing (DW) has become a popular solution. A Business Intelligence (BI) tools are often used on the top of DWs [1]. The adoption of big-data-driven technologies is still lagging among energy companies.

In this paper, based on our previous research projects and research results [2]–[5], we show that, through an optimization process, a gain in performance that is up to 2800 times faster compared with that of the original OLTP DBMS using the same hardware and the same operating system is possible.

The applied research project was funded by an NSERC grant and was conducted in 2020-2023 at Okanagan and Langara Colleges.

Index Terms—Smart Meters, OLTP, Data Warehouse, Performance, PostgreSQL DBMS, Design, Tuning.

I. Introduction

Historically, Harris SmartWorks, an NSERC project industrial client, has employed an OLTP database to serve clients' needs. While it performs well for daily transactions, its performance is affected by the increasing complexity of customer data extraction and reporting requests. A data warehouse is a cost-effective and efficient solution to this problem. By storing the data used in the extraction and reporting in a different database, it may be possible to enhance the performance of the OLTP and the efficiency of the reporting process.

The principles and properties of data warehousing lend themselves well to more efficient data retrieval. Data pre-aggregation to reduce the number of joins, bitmap indexing, partitioning, and parallel processing are targeted techniques to increase performance. Although data warehouses excel in these areas, some concessions must be made to achieve the desired results. These concessions can be strategically chosen only to keep what is necessary for data analysis.

Overall, the transformation of the database structure can lead to significant improvements in speed and space used while purposefully leaving out unnecessary information for data analysis. Note that the data warehouse is not the sole solution. It is usually used in parallel with the OLTP to offset its weaknesses.

Harris SmartWorks, in collaboration with Okanagan and Langara Colleges through several student capstones and funded applied research projects, tested the use of data warehousing from August 2021 to August 2022 to determine if it is a viable solution [2]–[5]. Many previous students' applied research and capstone projects since 2007 contributed to the great success of this project [6]–[22]. The results of these efforts led to the work documented in this paper.

The contributions of this applied research paper are in the following areas: (1) the investigation of several different DW schemas, (2) the rigorous testing for the suggested solutions (including execution path valuations for different queries),

We are thankful for the support of our research to NSERC and Harris SmartWorks, as well as to students and faculty of Okanagan and Langara Colleges.

II. R

The move from the initial Star schema to Star 1 was driven by the need to simplify and streamline the design of the different tables. In particular, two changes were made that would likely result in performance enhancements. First, a unique value (location_key) was removed from the star.location table, as it was unnecessary, and its inclusion resulted in table bloat in that it provided no new data that could not be derived through the Extract, Transform and Load (ETL) process. Next, the read_facts table was refactored to hold a fact for each read from a channel on a meter rather than all read from every channel of a meter in one record at any read time stamp. This allowed for a dynamic setup where read_facts now holds the read_val and Unit of Measurement (UoM) of every channel for each meter while removing reference to channel_id. Additionally, more entries can be added for extra channels rather than more channel columns or another table to hold all the added channels. Although removing an identification column is not a usual practice, the unit of measurement is unique for a given channel_id and commodity type. This means a primary key can be made without the use of channel_id. These changes make Star 1 simpler and streamlined compared to its predecessor.

The storage space taken by the Star 1 schema is tracked as it is a concern in the implementation phase. Table III shows a breakdown of the space requirement for the schema, its tables,

TABLE VII

Query Times (in ms

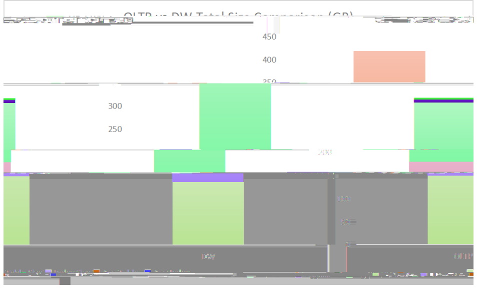

Fig. 6. Storage comparison of OLTP and DW based on Tables I, III and VIII

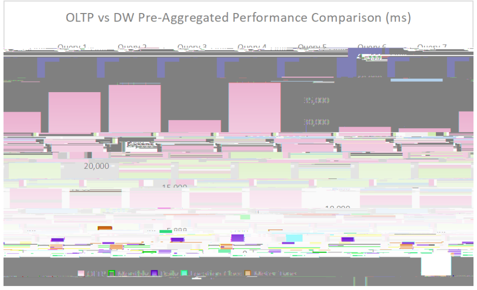

Fig. 7. OLTP vs PDW Performance

are in Figure 9 and Table X. Query times within the DW are lower than those within the OLTP. The pre-aggregation queries use the RI location_class compared with the OLTP using CO location_class.

TABLE X
QUERY TIME (IN MS) FOR OLTP WITH LOCATION_CLASS CO VERSUS PRE-AGGREGATION)

Query	Q1 (ms)	Q2 (ms)	Q3 (ms)	Q4 (ms)	Q6 (ms)	Q7 (ms)
OLTP	854	911	901	852	944	1,673
DW Agg	13 Loc Class	13 Loc Class	13 Loc Class	267 Monthly	3,779 Daily	593 Monthly
OLTP/DW Ratio	65.69	70.07	69.30	3.19	0.24	2.82

VII. PERFORMANCE IMPROVEMENT FOR THE DW

To further improve the performance of the DW, the following techniques were considered in the research:

- Partitioning
- Parallel Querying
- Bitmap indexing

A. Partitioning

Partitioning has been made viable for performance improvements in version 14 of PostgreSQL, which was used for the research. It has several advantages and some drawbacks, as discussed below.

Using range partitioning, a large table with millions of records can be split into multiple tables, each corresponding to a different range of data. Figure 10 shows an example range partition on time and the ability to sub-partition on more fine-grained ranges.

There are some simple rules on how partitioning works. One is that they work with inheritance, meaning child tables must have the same columns as their parents. Also, a partitioned table cannot be reclassified as a standard table and vice versa. Data must be transferred to a new table. Data cannot be moved between partitions by users unless it is also changed to fit into its new partition [32].

The pre-aggregated read facts table has a three-layer structure that helps with search performance and organization. Each partition has a range of values designated to it, determining what data it will hold.

Accessing the data is simple: query the top table, and the program can automatically tell where to look and search through the lower tables. This also allows for quickly dropping old data. You can select the specific partition, call for it to be dropped, and eliminate the whole data section without

Fig. 8. OLTP vs PDW Performance May 12th.

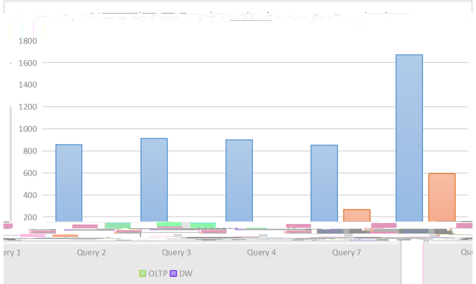

Fig. 9. Comparison of the OLTP queries with location class CO vs the fastest Pre-Aggregated queries. Time data from Table X.

Fig. 10. Diagram of potential partitioning structure

scanning the rest of the table to find all the relevant values.

As shown in Table XI, partitioning allows for significant workers per query and has already been used by both the DW and OLTP.

According to the Postgress 2022 technical documentation [32

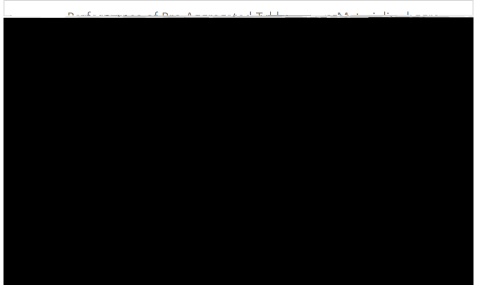

Fig. 13.   Time data from Table X.

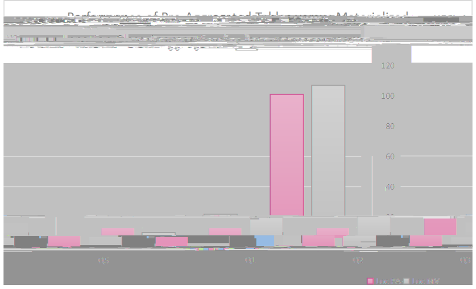

Fig. 14.   Time data from Table X.

College have also contributed to the research and development

[30] H. Märtens, E. Rahm, and T. Stöhr, “Dynamic query scheduling in parallel data warehouses,” Chichester, UK, pp. 1169–1190, 2003. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/cpe.786

[31] F. A. Khan, A. Ahmad, M. Imran, M. Alharbi, Mujeeb-ur-rehman, and B. Jan, “Efficient data access and performance improvement model for virtual data warehouse ,” pp. 232–240, 2017. [Online]. Available: https://go.exlibris.link/jjrcywDr

[32] T. P. G. D. Group, “Documentation PostgreSQL 10.20,” pp. 445–456, 2022.