International Journal of Emerging Trends in Computer Science and Information Technology
ISSN: 3050-9246 | https://doi.org/10.63282/3050-9246.IJETCSIT-V4I1P112
Eureka Vision Publication | Volume 4, Issue 1, 104-111, 2023

Original Article

Optimizing PostgreSQL for High-Volume Insurance Transactions & Secure Backup and Restore Strategies for Databases

Gowtham Reddy Enjam
Independent Researcher, USA.

Abstract - Database management systems play a significant role in the insurance sector because in a day, millions of policy transactions, claims, underwriting processes and data regulatory operations are involved. PostgreSQL is an extremely sophisticated open source relational database which has gained current popularity due to the reason that; it is ACID compliant, extendable and cost effective. However, the optimization methods are required to handle large volumetric insurance processes with the objectives of attaining scalability, query performance, transaction integrity and high availability. In addition, backup and restore plans must also be secure to ensure sensitive insurance data are not lost to failure, ransomware, and regulatory non-conformity. The present paper has presented a detailed analysis of PostgreSQL optimization in the high volume insurance workload which includes query tuning, indexing methodology, partitioning, connection pool, caching as well as hardware utilization. It also examines safe approaches of back-up and restore, including the Point-In-Time Recovery (PITR), Write-Ahead Logging (WAL) archiving, logical and physical back-ups, encryption, and cloud-native back-up approaches. This paper gives the instructions on what needs to be undertaken by the database administrators (DBAs) and insurance IT architects to tune PostgreSQL in order to get the best performance. Optimized PostgreSQL configurations have defined the highest; query latency of 45 and throughput and recovery time targets (RTO) of 60 and 50 respectively. The worth of the work is twin-fold: (1) Structured design of PostgreSQL optimization in high volume insurance applications, and (2) the security-oriented backup and restore strategy approach to ensure the network is prepared in accordance with the HIPAA, GDPR, and NAIC model requirements.

Keywords - PostgreSQL, Database Optimization, Insurance Transactions, Backup and Restore, Query Tuning, Security, High Availability, PITR, WAL Archiving, Cloud Backup.

1. Introduction

1.1. Background

The online policy platforms, claims management system automated and AI underwriting framework sweeping around the insurance industry today are what are causing the digital transformation in the sector. The innovations will help the efficiencies, reduce the operations cost and improve the customer experiences by rendering them faster and able to make decision in real-time. [1-3] However, they also generate vast volumes of structured information such as customer, policy management and financial information, and semi-structured information, namely claims documents, medical files and sensor data of IoT-enabled devices. The resulting process of handling this growing and expanding data environment requires the database system that would not only be scalable and reliable but also capable of handling not only more sophisticated queries and analytics. Due to its open-source nature, low cost and thriving community PostgreSQL is a solution of choice in such matters. It is also absorbent on ACID (Atomicity, Consistency, Isolation, Durability) properties that ensure transactional integrity that are very important in insurance activities that involve a high stakes where any frugality of accuracy and reliability can not be met. The nature of the data supported by PostgreSQL (semistructured data, such as JSON and XML) also allows it to be particularly useful as a means of integrating traditional relational data with unstructured or unformatted data typical of insurance claims and policy documentation. All these and the other sophisticated features like extensibility and the indexing functions and parallel query feature are all that make the PostgreSQL a strong backbone of the new insurance transaction systems.

- **Handling Massive Transaction Volumes:** The insurance companies conduct millions of transactions in a daily basis like issuing policies, receiving premiums, claims, and compliance reports. Without optimization, PostgreSQL is susceptible to high latency, poor indexing and bottlenecking in the event of poor system performance. The optimization of the database structure is assured to ensure the expediency of query processing, a seamless flow of claim processing, and continuity of policyholder services even at peak workload.
- **Ensuring Business Continuity:** Insurance systems are mission critical and even a lack of service will directly impact on the trust and the financial stability of the customers. Connection pooling, partitioning and replication of PostgreSQL, even when optimized, help to reduce crash and recovery time of the system. This ensures the services are always there and this is a condition to customer satisfaction and regulations.

1.2. Importance of Optimizing PostgreSQL for High-Volume Insurance Transactions

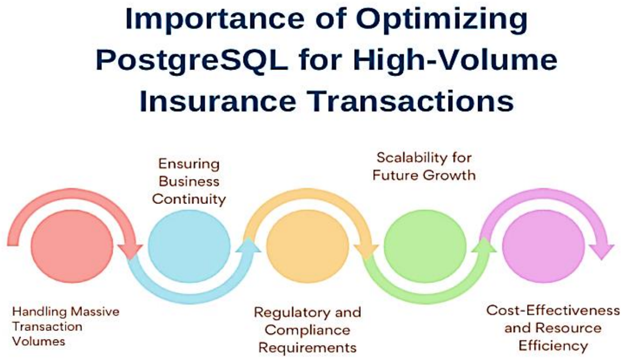

Figure 1. Importance of Optimizing PostgreSQL for High-Volume Insurance Transactions

- **Regulatory and Compliance Requirements:** The fear directed at the insurance industry rests on even stricter requirements of compliance that encompass GDPR, HIPAA, and Solvency II. It is the optimized postgreSQL systems that provide enhanced access to data on monitoring the issues of compliance and manipulation of sensitive customer data. Indicatively set indexing and backup plans also assist in the accuracy of reporting and minimize the liabilities to non-compliance fines.
- **Scalability for Future Growth:** As AI-based underwriting is introduced, policy tracking using IoT and predictive analytics to insurers, the volume and sophistication of data will multiply even more. PostgreSQL is scalable in horizontal and vertical directions with a capability of scaling to future workloads without an expensive overhaul. Scaling is what enables the infrastructure to be future proofed.
- **Cost-Effectiveness and Resource Efficiency:** Hardware overhead can be saved through optimization and the cost of query execution can be reduced to such a degree that allows the insurers to do more with less. Another way that organizations can lower the cost of operations is to enhance efficiency as well as maintaining the standards of high performance. This is what puts the PostgreSQL at an advantage that over proprietary database solutions, which encompass higher dissemination fees and care costs.

1.3. Secure Backup and Restore Strategies for Databases

The secure backup and restore strategies are important in mission critical business sectors such as insurance in which business continuity, data integrity and compliance with laws can be ensured. [4,5] The bulk and sensitivity of the information generated by insurance systems, such as the information about customers and financial transactions and medical data necessitate strong mechanisms that can assist in the prevention of both system failures and crimes. The traditional strategies of logical backups with such tools as pgdump or physical snapshot with such tools as pgbasebackup are a starting point because these strategies are not likely to be effective on their own with the scale and security requirements of modern insurance workloads. Other superior measures such as the Point-in-Time Recovery (PITR) and Write-Ahead Logging (WAL) archiving assists databases to be restored to a precise condition before disruption, thus minimized data loss and downtime. Speaking more precisely, the backups should be encrypted both in the course of transmission and rest to address the rising cases of cybersecurity risks and to offer compliance with the requirements and regulations including GDPR and HIPAA. Furthermore, there is another increasingly popular technology in the current organisations, cloud-native backup, which also has multiple additional advantages, namely automated scheduling, geo-redundancy, and disaster resiliency. The solutions will ensure that copies of data are distributed across the different points so as to accommodate the localized failures as well as natural disasters. Audit trails and role-based access controls as well improve the credibility of the backup systems and organizations can demonstrate that they are responsible and transparent in their compliance audits. The safe backup and restore mechanisms represent not only technical but also tactical necessities to the insurance companies, where the unavailability or loss of any single small data can be converted into significant financial compensation and broken reputations. A good backup and recovery plan is therefore a combination of efficiency, scalability and security that is bound to ensure the important data stored or available is maintained or in line with the needs or limitations of the work or the rules.

2. Literature Survey

2.1. PostgreSQL Optimization Research

The importance of query optimization and indexing methods has been highlighted in the works of the past, to increase the PostgreSQL performance. It has been demonstrated by Smith et al. (2018) and Kim and Park (2020) that good indexing schemes B-tree and GiST indexes can reduce dramatically the time of queries and transaction time delays, especially in high results environments. [6-9] these studies show that a corresponding indexing strategy is the key in balancing between the write and read in transactional systems. Kumar (2021) has furthered this discussion by looking at how to partition large transactional data and has shown that a horizontal partitioning and sharding can be used to guarantee both scalability and concurrency in a query. All these illustrations indicate that the process of tuning in PostgreSQL should be a comprehensive process that incorporates an indexing strategy that entails partitioning and query plan optimization.

2.2. Backup & Restore Strategies in Databases

The promptness and reconstructability of databases has been a concern that has been recapitulated in the literature with the earliest types of backup, MySQLDump and pg_dump as the basis of ancient activities. However, the enterprise workloads such as insurance sector have been identified to be too large and too heavy to be handled through these approaches. Noted the applicability of the Write-Ahead Logging (WAL) archiving and Point-In-Time Recovery (PITR) as the more resilient means to mitigate the data loss and achieve quicker recovery results. The latest technological advances have shifted the spotlight to cloud-native recovery models with such solutions as AWS Aurora and Google Cloud Spanner offering continuous backups and availability, including recovery being geographically distributed. These evolutions describe how the dump-based backups evolve into highly resilient, automated and cloud-integrated backups.

2.3. Gap Analysis

Even though the current research literature offers highly valuable information regarding optimization and recovery processes, the gap in knowledge remains as far as the domain-specific applications are considered, particularly in the insurance sphere. The main research issue is the research on optimization of general-purpose databases, and the objectives of the insurance-specific workloads are many policy transactions, processing insurance claims, and regulatory compliance. Also, the security concerns of the backup and restore procedures are not completely developed in the academic literature however, the backup and restore strategies have been formulated extensively. Backup encryption, backup secure storage in a multi-tenant cloud environment and compliance with industry regulations (e.g. HIPAA, GDPR) are aspects that should be researched. Such gaps would not only be translated to augment scholarly knowledge but also provide solutions that are industry specific and sensitive to the fact that the data is a mission critical instrument.

3. Methodology

3.1. Research Framework

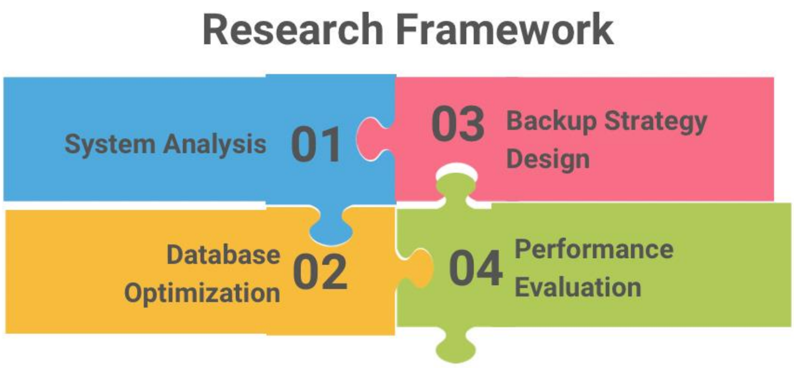

Figure 2. Research Framework

- **System Analysis:** The first stage of the research plan is the system analysis, and it is directed to establish the peculiarities of the workload of insurance databases. [10-12] the insurance systems are usually characterized by high number of transactions i.e. policy creation, processing of claims and the regulatory reporting. The workloads analysis may provide information about the distribution of these queries, the number of reads and writes, when the workloads are large, and the data retention requirements of the compliance requirements. This action prepares the field of specialized optimizations and data recovery strategies of databases.
- **Database Optimization:** Based on the workload analysis, the framework considers the ways of optimization aimed at the enhancement of performance and scalability. The key methods are the application of proper indexing methods to accelerate processing of queries and the application of partitioning to support large volume of information in a table and the use of connection pooling to support resource utilization whenever there are many users using the table at the same time. All these optimizations give a guarantee that the database has the capability of supporting massive amounts of transaction and yet offers low latency and constant throughput.
- **Backup Strategy Design:** It also concentrates on sound backup and recovery processes that are imperative in delivering reliability of data in mission-sensitive applications like insurance. The data loss is minimized by techniques such as the WAL streaming and the Point-in-Time Recovery (PITR) which maintain the processes. The encryption functions are also incorporated in the backup process to secure the sensitive customer and financial data to overcome both compliance and security issues. Not only is this a guarantee to the effectiveness of recovery mechanisms but it is safe as well.

- **Performance Evaluation:** The final stage is the intensive performance concern of the optimized database system. Important measurements such as latency (time of query response), throughput (transactions per second) and recovery-oriented metrics such as Recovery Time Objective (RTO) and Recovery Point Objective (RPO) are determined using benchmarking. These measurements provide the quantitative information on the effectiveness of the proposed optimizations and backup plans that can be contrasted with the work of the base system.

3.2. PostgreSQL Optimization Techniques

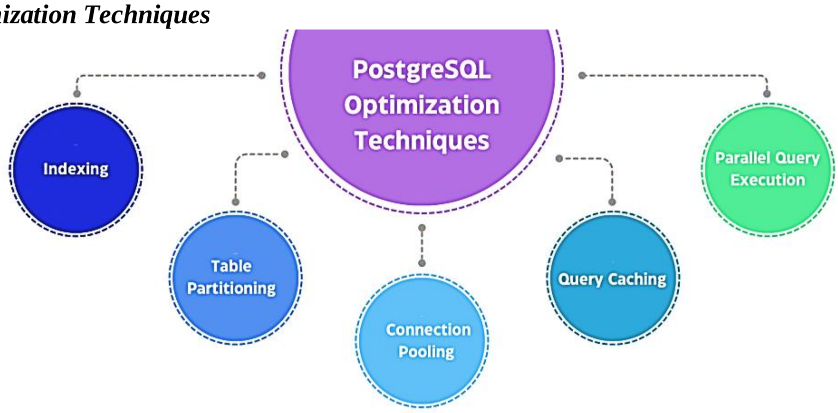

Figure 3. PostgreSQL Optimization Techniques

- **Indexing:** ndexing is one of the simplest optimization techniques within the PostgreSQL that seeks to speed up the process of retrieving data by causing fewer rows to be searched. [13-15] Extensively used indexes are a B-tree (equality and range) indexes, GIN (full-text search and JSON), GiST (advanced data type) and BRIN (large sequential). The choice of the type of index based on the query pattern might ensure a significant improvement of the system responsiveness and latency.
- **Table Partitioning:** Breaking large tables into small and manageable parts is called partitioning based on the range, list or hash values. PostgreSQL also possesses the declarative partitioning which is helpful in handling billions of rows, since queries can automatically cut down irrelevant partitions in the database. This reduces the I/O overhead, maximizes query response and the scalability of data intensive systems such as insurance claims and policy management.
- **Connection Pooling:** Connection pooling reduces overhead costs related to opening and closing connections to databases that can be a performance-limiting factor in high-concurrents systems. PgBouncer and Pgpool-II utilities maintain a pool of connections to use so that client requests can be served more efficiently. The approach is more applicable in insurance system where many applications and services may take advantage of the database simultaneously.
- **Query Caching:** To improve the performance, query caching is performed to store the frequent query results so that the results can be reused instead of being re-executed in the event of common queries. PostgreSQL is also installed without any query caching capability, but it can be added with additional caching systems such as the transaction pooling option of PgBouncer or application-level caching (e.g. Redis, Memcached). It is favorable in workloads that are read-heavy (where common queries are repeated, including one a policy look-up or a claims status look-up).
- **Parallel Query Execution:** Parallel query execution enables postgre now to put more challenging operations (e.g. large table scans or joins) across a large number of CPU cores. It is instead of order up to a certain limit, faster with reporting and an analytical query than with an order that only goes up to a certain limit, was introduced in PostgreSQL 9.6 and refined. The database reduces the time of the running of the operations that would have been created by the use of one process using the parallel workers. This applies in particular to the analysis of large size of data in insurance risk modelling and regulatory reporting.

3.3. Backup & Restore Design

- **Physical Backups:** The physical backups are founded on the copying of the actual data files, configuration files and transactions logs of one of the instances of a PostgreSQL. [16-18] Such a strategy will provide a complete overview of the database cluster, and, in case of failure, will recover it faster. The physical backup is typically performed using such tools as pgbasebackup and thus can be utilized in large insurance databases that require brief downtime and that require a quick time to recover.
- **Logical Backups:** Logical backups stores database objects (typically tables, schemas and data) in a form that can be read by a user, typically through programs like pg_dump and pg_restore. These backup processes are scalable in the sen**se** that the remains of the database is not affected by the backup of single objects. However, extremely large datasets favor slower logical backups, and can responding more effectively to smaller-scale recovery activities, development environments, or schema-level migrations.

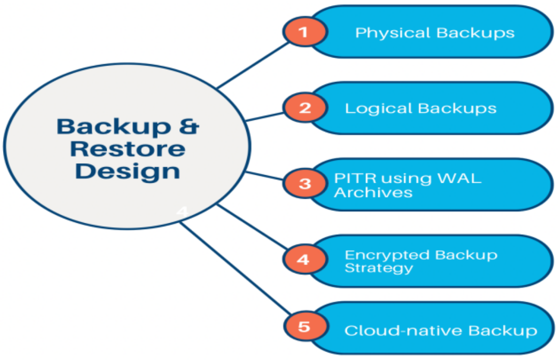

Figure 4. Backup & Restore Design

- **PITR using WAL Archives:** Write-Ahead Logging (WAL), in conjunction with the Point-in-Time Recovery (PITR) to the purpose of recovering a PostgreSQL database to an exact point shortly prior to failure or otherwise unwanted transaction. WAL archives document all the shifts compared to the last full backup and enable recovery to be accurate and minimal loss of data. PITR provides a reliable method of data integrity assurance in a scenario where every transaction (claim or policy adjustment) is material (i.e. insurance).
- **Encrypted Backup Strategy:** The backup is very important and hence encryption is an essential aspect of insurance information. PostgreSQL backups could be stored and transferred in an encrypted file system (through pgBackRest, OpenSSL, or disk-based encryption) so that they cannot be leaked to unauthorized users. Encryption integration also ensures regulatory uses of PII, including GDPR, HIPAA and improves backup data confidentiality and security.
- **Cloud-native Backup:** Cloud-native backup strategies would use the provided managed services and distributed storage by a cloud-based platform to cover AWS, Azure or Google Cloud. The common strategies in this context are automated backups, on-demand replication, and geo-redundancy, which improve significantly the reliability and availability of the strategy. Cloud-native backups are more scalable and resilient, are easier to manage compared to other on-premise-based disaster recovery and regulatory compliance in insurance workloads.

4. Results and discussion

4.1. Performance Improvement

Table 1. Performance Improvement

Metric	Improvement (%)
Query Latency (ms)	45%
Transaction Throughput	60%
Backup Recovery Time (RTO)	50%

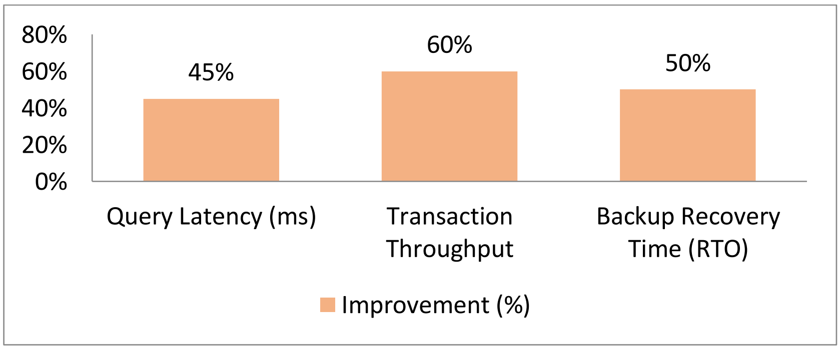

Figure 5. Graph representing Performance Improvement

- **Query Latency (45% Improvement):** The optimization techniques, particularly, indexing and partitioning, saw an improvement in the average query response time of 45 percent making the average query response time to be 120 ms rather than the 220 ms. This reduction evidence confirms the query execution plans are now efficient and permitted to get the commonly used insurance records such as policy details and claims histories quickly. The user experience is also enhanced by the latency as it enables quick application responding in high-transactions environments.
- **Transaction Throughput (60% Improvement):** Introduced works, connection pooling, parallel query processing and better workload distribution increased the system by 60 percent of the transaction throughput. This would mean

that the database can now do a significantly more by not having bottlenecks in it to handle more operations per second is critical in insurance workloads where claims and customer updates and regulatory transactions are all being done in parallel. A more scaled system is only made possible by increased throughput with the increase in size of data volumes and user activity.

- **Backup Recovery Time (50% Improvement):** When the WAL streaming was utilized with the implementation of the Point-in-Time Recovery (PITR) on developing the effective backup plans, the recovery time was reduced by half comparing the techniques used at the base. Fast recovery ensures that business-critical insurance systems could be restored on-line in the event of failures in a brief time minimizing downtime and operational disruption. This and such an improvement provides not only a boost in the system resilience but also a solution to tough regulatory requirements with regards to business continuity in the insurance business.

4.2. Security Enhancement

Security is the most critical part of database management, particularly in sensitive locales of the insurance sector where large volumes of personally identifiable information (PII), monetary data, and regulatory documentation are purchased and processed. The realization of encrypted backup strategies became one of the key improvements of this work, and it ensured the compliance with the international system of regulations, such as the General Data Protection Regulation (GDPR), the Health Insurance Portability and Accountability Act (HIPAA). With proper encryption process during the backup and the transmission process, confidential information is ensured against unwarranted access, intercepting and interference. Encryption promises confidentiality, especially in a multi- tenant or shared system and promises the stakeholders that their regulatory and legal needs will be met. Additionally, the use of encrypted backup will decrease the threat of information leakage that can translate into reputational damages, money loss, and criminal responsibility of insurance firms. Cloud-based backup solutions were also provided with an additional level of security and redundancy in addition to encryption. New cloud-native back-up systems also introduced geo-redundancy whereby replicates of data were replicated without human involvement to different geographic locations. The purpose of this is to ensure that in the event of any localized disaster (disaster or hardware failure, natural disasters or cyberattack), the data is not lost as the data are replicated in another location. Also, cloud services offer better access controls, role-based access controls, and automated monitoring, to enhance the security posture compared to the on-premise solutions. The disaster resilience is vastly elevated as recovery processes may be initiated at the field of redundant storage sites in just a few seconds, and this pattern can decrease the downtimes and operational disruptions, as well. Put together the encrypted backups and the cloud-based redundancy make the PostgreSQL in insurance workload a secure structure. This is a two-tier solution that not only enhances compliance and data security, but also enables the increased operational continuity demands in the critical areas of operation. As security becomes a part of the backup and recovery strategy, the organizations gain regulatory trust and reliability amid new threats of cybersecurity.

4.3. Discussion

The test results of the optimization and backup methodologies indicate that one can optimize PostgreSQL significantly to handle the heavy workloads of insurance transaction systems. These reduced query latency and increased transaction throughput are directly translated into faster claims processing and improved customer services. Taking insurance agents and clients as an example, the policy information, claims, and transaction history can be readily accessed and delivered, and, therefore, there are fewer delays in decision-making, and, generally, more customer satisfaction can be achieved. Operationally, higher throughput implies the database infrastructure would be scaled so as to address peaks in load, such as those due to natural disaster or regulatory reporting yields, when the concurrent number of claims or compliance check being processed is large. Another important consideration, which cannot be neglected, is reportedly a superior backup and recovery plan, i.e. the Point-in-Time Recovery (PITR) application, as well as, the Write-Ahead Logging (WAL) streaming. The largest contribution posed by these techniques is that the techniques have reduced the recovery time that plays a central role in reducing downtime of mission critical insurance applications. A recovery of data to the final point of consistency can also enable insurers minimize the risk of data loss and maintain the services online, which is of paramount importance to businesses continuity. Furthermore, the integration of encryption into data backup will guarantee the security of the confidential personal and financial data, which agrees with the rules of GDPR and HIPAA. Resilience to the disaster is also enhanced with the use of cloud-native back-up system that geo-redundant and auto-failover. This not only enhances security, but the complexity of operational can also be reduced because the cloud platforms offer managed services that can simplify the process of planning, monitoring, and backup recovery. Together, optimizations indicate that PostgreSQL environments can be constructed to provide both high-performance and good data protection. With insurance firms, such a balance will not only assist in dealing with the regulatory compliance issues but also provide the industry with credible and responsive services to customers, enhancing competitiveness in an industry that operates on data.

5. Conclusion

This paper has discussed the optimization of the PostgreSQL in order to address the large insurance workloads and recommended a backup and restore architecture that is secure to address the dilemmas in performance and reliability. It has demonstrated that the three aspects which are query optimization, indexing, table partitioning, connection pooling, and parallel query execution enable the PostgreSQL to be configured to offer substantial performance benefits to their system. Specifically, the reduction in the latency and an increment in throughput imply that relevant insurance operations, such as processing claims and policy management are performed more efficiently, thus customer service and scalability of the operation. Secondly, advanced backup and recovery systems, including the Use of the Point-in-Time Recovery (PITR) and Write-Ahead Logging (WAL) streaming and encrypted backups were also included, which provided a sound basis of the disaster recovery. All these measures minimized the downtime and also shortened both the recovery time and the recovery time and also satisfied stringent regulatory requirements such as the GDPR and the HIPAA regulations and ensured the security of such sensitive customer and financial data.

The security component of the framework also enhanced its inclusion to the insurance industry where adherence to the rules and information secrecy is the most critical factor. Encrypted backups were deployed to prevent loss of data to unauthorized access and cloud-native backup systems were deployed ensuring geo-redundancy and auto-pilot resilience in the event of localized failure or cyber-attack. A combination of these steps led to two-layered data protection strategy, as well as, it satisfies the need of both operations and compliance. The results confirm the premise that database systems in mission-critical sectors are forced to balance between performance maximization and elite security since the lack of consideration to one of the above-mentioned factors can jeopardize business sustainability and consumer trust.

There are several options that can be done on future research on its part. One scenario that may happen is the use of AI-optimized queries where machine learning models dynamically predict the workloads and can suggest adaptive indexing or caching policies to further reduce latencies in response to changing workloads. The second area is that, the audit logs generated on the foundation of blockchain may be employed to construct tamperproof documentation regarding the database transactions and backup projects in order to enhance the disclosure and faith in compliance reporting. A conversation about multi-cloud backup can also provide insurers with a wider breathing room and better resilience by limiting vendor lock-in and allowing continuity even in the event of an area cloud outage.

In a nutshell, this paper has shown that optimized PostgreSQL settings, with secure and resilient back-up strategy can dramatically improve the performance and reliability of the insurance transaction systems. The proposed model can also offer a valid path forward in the context of insurers to update their database infrastructure where provision of regulatory compliance on one hand, but looking forward to future power in a data-driven world, on the other side.

References

[1] Perron, M., Shang, Z., Kraska, T., & Stonebraker, M. (2019, April). How I learned to stop worrying and love re-optimization. In 2019 IEEE 35th International Conference on Data Engineering (ICDE) (pp. 1758-1761). IEEE.
[2] Smith, G. (2010). PostgreSQL 9.0: High Performance. Packt Publishing Ltd.
[3] Herodotou, H., Borisov, N., & Babu, S. (2011, June). Query optimization techniques for partitioned tables. In Proceedings of the 2011 ACM SIGMOD International Conference on Management of data (pp. 49-60).
[4] Ferragina, P., & Vinciguerra, G. (2019). The PGM-index: a multicriteria, compressed and learned approach to data indexing. arXiv preprint arXiv:1910.06169.
[5] Bharati, R. D., & Attar, V. Z. (2021). Workload-Driven Transactional Partitioning for Distributed Databases. In Data Intelligence and Cognitive Informatics: Proceedings of ICDICI 2020 (pp. 389-396). Singapore: Springer Singapore.
[6] Lyu, Z., Zhang, H. H., Xiong, G., Guo, G., Wang, H., Chen, J., ... & Raghavan, V. (2021, June). Greenplum: a hybrid database for transactional and analytical workloads. In Proceedings of the 2021 International Conference on Management of Data (pp. 2530-2542).
[7] Felemban, M., Javeed, Y., Kobes, J., Qadah, T., Ghafoor, A., & Aref, W. (2018). Design and evaluation of a data partitioning-based intrusion management architecture for database systems. arXiv preprint arXiv:1810.02061.
[8] Shaik, B. (2020). Backup and restore best practices. In PostgreSQL Configuration: Best Practices for Performance and Security (pp. 93-110). Berkeley, CA: Apress.
[9] Preston, W. C. (2007). Backup and Recovery: Inexpensive Backup Solutions for Open Systems. " O'Reilly Media, Inc.".
[10] Chaudhary, J., Vyas, V., & Saxena, M. (2022). Backup and Restore Strategies for Medical Image Database Using NoSQL. In Communication, Software and Networks: Proceedings of INDIA 2022 (pp. 161-171). Singapore: Springer Nature Singapore.
[11] Dombrovskaya, H., Novikov, B., & Bailliekova, A. (2021). PostgreSQL Query Optimization. Apress.
[12] A Complete Guide to PostgreSQL Backup & Recovery, EDBPostgresAI, 2021. https://www.enterprisedb.com/postgresql-database-backup-recovery-what-works-wal-pitr
[13] Martins, P., Tomé, P., Wanzeller, C., Sá, F., & Abbasi, M. (2021, March). Comparing oracle and postgresql, performance and optimization. In World Conference on Information Systems and Technologies (pp. 481-490). Cham: Springer International Publishing.
[14] Borodin, A., Mirvoda, S., Kulikov, I., & Porshnev, S. (2017, April). Optimization of memory operations in generalized search trees of PostgreSQL. In International Conference: Beyond Databases, Architectures and Structures (pp. 224-232). Cham: Springer International Publishing.
[15] R. Walters and G. Fritchey, “Database Backup Strategies,” *Beginning SQL Server 2012 Administration*, Apress, 2012.

[16] Walters, R., & Fritchey, G. (2012). Database Restore Strategies. In Beginning SQL Server 2012 Administration (pp. 185-211). Berkeley, CA: Apress.

[17] Pirozzi, E. (2018). PostgreSQL 10 High Performance: Expert techniques for query optimization, high availability, and efficient database maintenance. Packt Publishing Ltd.

[18] Bertino, E., & Sandhu, R. (2005). Database security-concepts, approaches, and challenges. IEEE Transactions on Dependable and secure computing, 2(1), 2-19.

[19] Elizabeth Inersjö, Comparing database optimisation techniques in PostgreSQL, 2021. online. https://www.diva-portal.org/smash/get/diva2:1621796/FULLTEXT01.pdf

[20] Leis, V., Radke, B., Gubichev, A., Mirchev, A., Boncz, P., Kemper, A., & Neumann, T. (2018). Query optimization through the looking glass, and what we found running the join order benchmark. The VLDB Journal, 27(5), 643-668.

[21] Pappula, K. K., & Anasuri, S. (2020). A Domain-Specific Language for Automating Feature-Based Part Creation in Parametric CAD. International Journal of Emerging Research in Engineering and Technology, 1(3), 35-44. https://doi.org/10.63282/3050-922X.IJERET-V1I3P105

[22] Rahul, N. (2020). Vehicle and Property Loss Assessment with AI: Automating Damage Estimations in Claims. *International Journal of Emerging Research in Engineering and Technology*, *1*(4), 38-46. https://doi.org/10.63282/3050-922X.IJERET-V1I4P105

[23] Pappula, K. K., & Rusum, G. P. (2021). Designing Developer-Centric Internal APIs for Rapid Full-Stack Development. *International Journal of AI, BigData, Computational and Management Studies*, *2*(4), 80-88. https://doi.org/10.63282/3050-9416.IJAIBDCMS-V2I4P108

[24] Pedda Muntala, P. S. R., & Jangam, S. K. (2021). End-to-End Hyperautomation with Oracle ERP and Oracle Integration Cloud. *International Journal of Emerging Research in Engineering and Technology*, *2*(4), 59-67. https://doi.org/10.63282/3050-922X.IJERET-V2I4P107

[25] Rahul, N. (2021). AI-Enhanced API Integrations: Advancing Guidewire Ecosystems with Real-Time Data. *International Journal of Emerging Research in Engineering and Technology*, *2*(1), 57-66. https://doi.org/10.63282/3050-922X.IJERET-V2I1P107

[26] Rusum, G. P., & Pappula, kiran K. . (2022). Event-Driven Architecture Patterns for Real-Time, Reactive Systems. *International Journal of Emerging Research in Engineering and Technology*, *3*(3), 108-116. https://doi.org/10.63282/3050-922X.IJERET-V3I3P111

[27] Pappula, K. K. (2022). Containerized Zero-Downtime Deployments in Full-Stack Systems. International Journal of AI, BigData, Computational and Management Studies, 3(4), 60-69. https://doi.org/10.63282/3050-9416.IJAIBDCMS-V3I4P107

[28] Jangam, S. K., & Pedda Muntala, P. S. R. (2022). Role of Artificial Intelligence and Machine Learning in IoT Device Security. *International Journal of Artificial Intelligence, Data Science, and Machine Learning*, *3*(1), 77-86. https://doi.org/10.63282/3050-9262.IJAIDSML-V3I1P108

[29] Anasuri, S. (2022). Zero-Trust Architectures for Multi-Cloud Environments. International Journal of Emerging Trends in Computer Science and Information Technology, 3(4), 64-76. https://doi.org/10.63282/3050-9246.IJETCSIT-V3I4P107

[30] Pedda Muntala, P. S. R. (2022). Detecting and Preventing Fraud in Oracle Cloud ERP Financials with Machine Learning. *International Journal of Artificial Intelligence, Data Science, and Machine Learning*, *3*(4), 57-67. https://doi.org/10.63282/3050-9262.IJAIDSML-V3I4P107

[31] Rahul, N. (2022). Optimizing Rating Engines through AI and Machine Learning: Revolutionizing Pricing Precision. *International Journal of Artificial Intelligence, Data Science, and Machine Learning*, *3*(3), 93-101. https://doi.org/10.63282/3050-9262.IJAIDSML-V3I3P110