



Ciencia Latina
Internacional

Ciencia Latina Revista Científica Multidisciplinar, Ciudad de México, México.
ISSN 2707-2207 / ISSN 2707-2215 (en línea), julio-agosto 2024,
Volumen 8, Número 4.

https://doi.org/10.37811/cl_rcm.v8i4

**ANÁLISIS DE RENDIMIENTO ENTRE
PROCEDIMIENTOS ALMACENADOS Y
CONSULTAS TRADICIONALES: CASO DE ESTUDIO
EN LOS MÓDULOS DE ARCHIVOS Y PRÉSTAMOS EN
EL SISTEMA DEL CECYTEG**

**PERFORMANCE ANALYSIS BETWEEN STORED
PROCEDURES AND TRADITIONAL QUERIES: CASE STUDY IN
THE FILE AND LOAN MODULES OF THE CECYTEG SYSTEM**

Luis Angel Riaño Calixto
Tecnológico Nacional de México

Uziel Trujillo Colón
Universidad Autónoma de Guerrero, México

DOI: https://doi.org/10.37811/cl_rcm.v8i4.13434

Análisis de Rendimiento entre Procedimientos Almacenados y Consultas Tradicionales: Caso de Estudio en los Módulos de Archivos y Préstamos en el Sistema del CECyTEG

Luis Angel Riaño Calixto¹MM22320017@acapulco.tecnm.mx<https://orcid.org/0000-0002-6711-063X>Tecnológico Nacional de México
IT de Acapulco, México.**Uziel Trujillo Colón**Uziel.trujillo@gmail.com<https://orcid.org/0009-0006-8238-6163>Universidad Autónoma de Guerrero
México

RESUMEN

Este artículo explora los diversos métodos de almacenamiento de datos utilizados en las bases de datos de los sistemas de gestión documental. El objetivo principal es analizar y proporcionar una visión clara y detallada sobre el rendimiento de los procedimientos almacenados y las consultas tradicionales en áreas como tiempo de ejecución, integridad de datos, escalabilidad, usabilidad, mantenibilidad y seguridad, facilitando la toma de decisiones sobre cuál método utilizar en cada caso. La metodología incluye una revisión de la literatura existente y estudios de caso en instituciones que han implementado estos métodos en sus sistemas, complementada por pruebas que detallan los resultados obtenidos. Los resultados destacan mejoras significativas en seguridad, mantenimiento y escalabilidad para los procedimientos almacenados, aunque también se identifican ventajas en las consultas tradicionales, como un mejor rendimiento en consultas menos complejas o exigentes. En conclusión, el artículo sugiere que, dependiendo del caso, tanto los procedimientos almacenados como las consultas tradicionales ofrecen oportunidades prometedoras para optimizar el rendimiento del sistema y la gestión de recursos.

Palabras clave: administración, base de datos, CECyTEG, educación, procedimientos almacenados

¹ Autor principal

Correspondencia: MM22320017@acapulco.tecnm.mx

Performance Analysis Between Stored Procedures and Traditional Queries: Case Study in the File and Loan Modules of the CECyTEG System

ABSTRACT

This article explores various data storage methods used in document management system databases. The primary objective is to analyze and provide a clear and detailed understanding of the performance of stored procedures and traditional queries in areas such as execution time, data integrity, scalability, usability, maintainability, and security, facilitating decision-making on which method to use in different scenarios. The methodology includes a review of existing literature and case studies of institutions that have implemented these methods in their systems, complemented by tests that detail the results obtained. The results highlight significant improvements in security, maintenance, and scalability for stored procedures, while also identifying strengths in traditional queries, such as better performance for less complex or demanding queries. In conclusion, the article suggests that, depending on the case, both stored procedures and traditional queries offer promising opportunities to optimize system performance and resource management.

Keywords: administration, CECyTEG, database, education, stored procedures

Artículo recibido 10 julio 2024

Aceptado para publicación: 15 agosto 2024



INTRODUCCIÓN

En la era digital actual, los sistemas de información constituyen un pilar fundamental para la administración, organización y almacenamiento de grandes volúmenes de datos generados por miles de usuarios. Existen diversas metodologías de almacenamiento de información en bases de datos, destacando entre ellas el uso de SQL (Structured Query Language) y NoSQL (Not Only SQL). En su estudio, Muhammad Zohaib Khan (2022) compara SQL y NoSQL, abordando ¿Qué son?, ¿Cómo manejan datos estructurados y no estructurados? y ¿Cuál es más rápido, eficiente y ofrece mejor rendimiento?.

Las bases de datos SQL presentan diversas ventajas, tales como una estructura clara y definida, consistencia e integridad de los datos, soporte maduro y establecido, capacidad avanzada de consulta y transacciones robustas. Estas características convierten a SQL en una opción ideal para aplicaciones que exigen precisión, estabilidad y una gestión rigurosa de la información (Walther, 2023).

En contraste, las bases de datos NoSQL destacan por sus ventajas en términos de escalabilidad horizontal, flexibilidad de esquema, rendimiento optimizado para cargas específicas, gestión eficiente de grandes volúmenes de datos y adaptabilidad a las tendencias actuales. Las características de NoSQL lo hacen especialmente adecuado para aplicaciones que gestionan grandes volúmenes de datos no estructurados y requieren una rápida adaptabilidad a los cambios en los requisitos del sistema (Walther, 2023).

En el ámbito de las bases de datos SQL, existen diversos métodos para el almacenamiento de datos, mediante consultas tradicionales, funciones, disparadores y procedimientos almacenados, cada uno de estos métodos tiene ventajas y desventajas (Septiadi, 2020). Una consulta tradicional (CT) es una instrucción SQL que se utiliza para recuperar, insertar, actualizar o eliminar información en una base de datos (Forta, 2019), mientras que un procedimiento almacenado (PA) es un conjunto de instrucciones SQL precompiladas que se agrupan y almacenan en una base de datos para su posterior ejecución (Navathe, 2017).

Empresas del sector tecnológico demandan la búsqueda constante de estrategias para optimizar sistemas y garantizar su seguridad, ofreciendo así una experiencia de usuario de primera categoría (Antònia Mas, 2005).



Por consiguiente, es esencial contar con una variedad de enfoques que permitan mejorar el tiempo de respuesta, incluyendo patrones de diseño que se ajusten a la arquitectura del sistema y métodos de consulta optimizados para proporcionar la información necesaria en el momento preciso.

El Colegio de Estudios Científicos y Tecnológicos del Estado de Guerrero (CECyTEG) es una institución educativa comprometida con la excelencia académica, actualmente compuesta por 10 planteles y 88 subplanteles. Con el crecimiento continuo del sistema de administración del CECyTEG y la gestión de volúmenes significativos de información procedentes de diversos procesos institucionales, es crucial asegurar respuestas eficientes y ágiles.

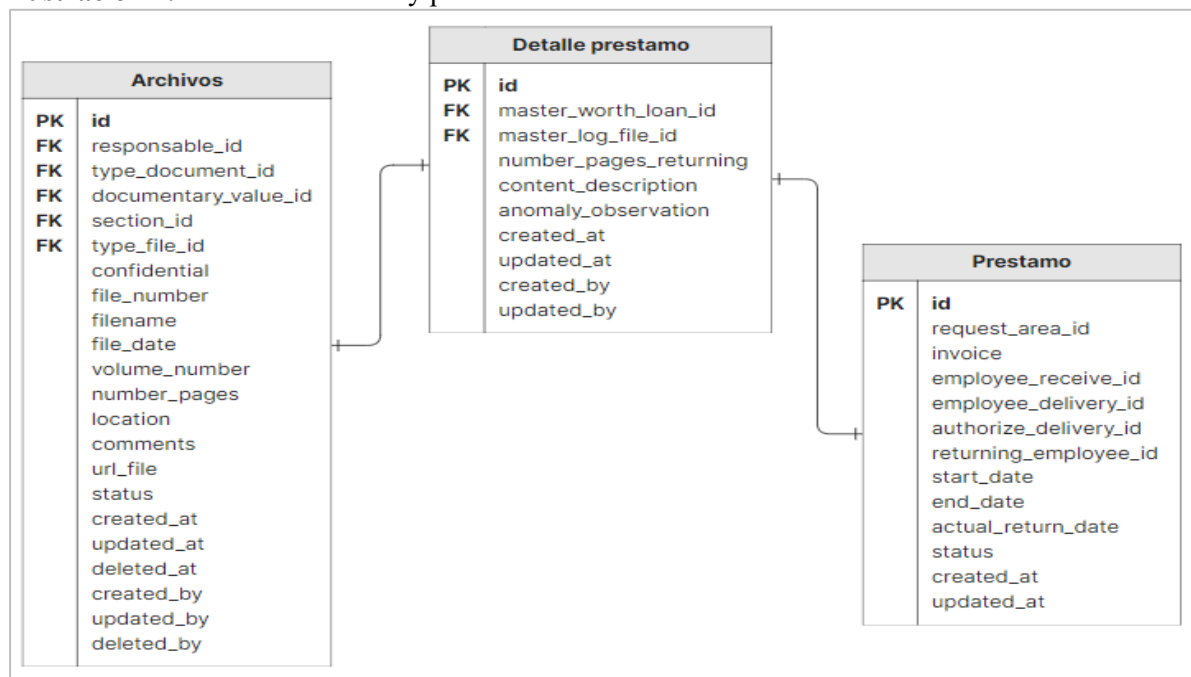
En este contexto, el uso de SQL se justifica debido a su capacidad para proporcionar una estructura clara y definida, así como su robustez en la gestión de transacciones y consultas avanzadas. Además, se han evaluado diversas estrategias para mejorar el sistema, considerando la alta demanda de información por parte de padres, alumnos y personal administrativo, así como la imperiosa protección de la integridad y seguridad de los datos.

El presente estudio realiza un comparativo evaluando el rendimiento y la funcionalidad de los procedimientos almacenados en contraste con las consultas tradicionales, abordando aspectos como tiempo de ejecución, integridad de datos, escalabilidad, usabilidad, mantenibilidad y seguridad. El análisis tiene como objetivo identificar las fortalezas y debilidades de los métodos de consulta CT y PA, facilitando así su integración efectiva en el sistema de administración del CECyTEG.

METODOLOGÍA

La investigación presenta un enfoque cuantitativo con el objetivo de detallar al máximo los resultados obtenidos. Las muestras se recopilaron mediante pruebas efectuadas en las tablas de archivos y préstamos de la base de datos (Ilustración 1).

Ilustración 1: Tablas de archivos y préstamo.



Fuente: Elaboración propia.

Proceso de evaluación (Ilustración 2):

Ambos métodos de consultas, procedimientos almacenados y consultas tradicionales fueron sometidos a evaluación, analizando las siguientes pruebas:

1. Consultas y Procedimientos Almacenados:

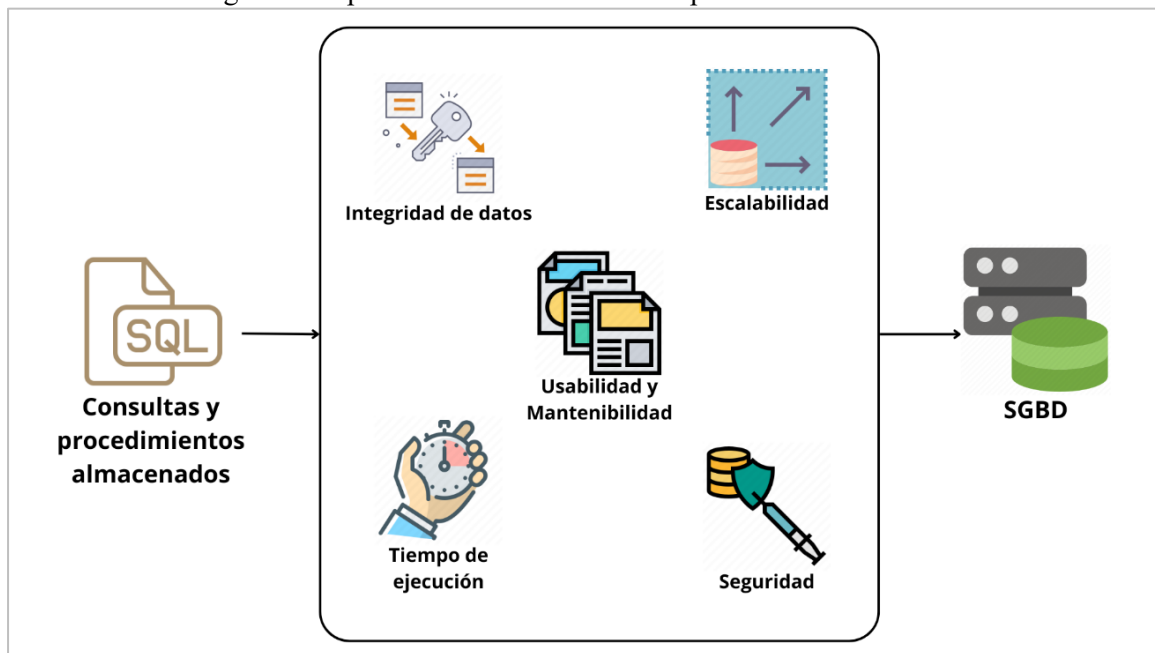
- **Tiempo de ejecución:** Se toma la medición del tiempo que tardan en ejecutarse y obtener los resultados los métodos de consulta del SGBD medido en segundos(s).
- **Integridad de datos:** Se verifica la capacidad de las consultas tradicionales y los procedimientos almacenados para mantener la integridad de los datos de entrada durante las operaciones de inserción y actualización.
- **Escalabilidad:** Se evalúa la capacidad de cada método de consulta para manejar un aumento progresivo en la carga de trabajo.
- **Usabilidad y Mantenibilidad:** Se evalúa la facilidad de uso y la capacidad de mantenimiento a largo plazo, considerando factores como la simplicidad del código, la facilidad de implementación y la adaptabilidad a cambios futuros.
- **Seguridad:** Se evalúa la seguridad de cada enfoque, examinando la capacidad de proteger la base de datos contra amenazas externas e internas.

2. Sistema de Gestión de Bases de Datos (SGBD):

- El SGBD se utilizará como plataforma de pruebas para ejecutar las consultas y los procedimientos almacenados. Se tomarán medidas de rendimiento y se analizarán los resultados para cada una de las categorías mencionadas.

Esta evaluación permite identificar el método más eficiente y seguro para gestionar las operaciones en *los módulos de archivos, préstamos y migración*.

Ilustración 2: Diagrama del proceso de evaluación de las pruebas.



Fuente: Elaboración propia.

A las pruebas de integridad de datos, escalabilidad, usabilidad, mantenibilidad y seguridad se implementará un sistema de puntuación para representar de manera más clara los puntos fuertes y débiles de ambos tipos de consultas, donde 1 indica la menor eficiencia y 10 indica la mayor eficiencia. Con el fin de realizar las pruebas, se añadieron más de 10,000 registros, lo cual nos permitirá evaluar y verificar los recursos utilizados por cada tipo de consulta según el tipo de prueba realizada. De esta manera, obtendremos los datos necesarios para tener una idea clara de cuál es la opción más eficiente. En la Tabla 1 se presenta las diferencias entre la sintaxis de una consulta tradicional y un procedimiento almacenado.

Tabla 1: Sintaxis de consulta tradicional y procedimiento almacenado.

Consulta Tradicional	Procedimiento Almacenado
SELECT parámetros FROM tabla WHERE condiciones;	CREATE PROCEDURE ConsultaDatos BEGIN SELECT parámetros FROM tabla WHERE condiciones; END;

Fuente: Elaboración propia

Es importante recalcar la manera en la que se ejecutan las diferentes consultas, lo cual también se toma en consideración en el estudio. La sintaxis de ambas consultas está representada en la Tabla 1. A continuación, en la Tabla 2 se representa un ejemplo de cómo se ejecutan los dos métodos:

Tabla 2: Ejemplo de ejecución de sintaxis en las consultas.

Consulta Tradicional	Procedimiento Almacenado
SELECT * FROM prestamos WHERE usuario_id = ?;	Call ConsultarPrestamo (?)
INSERT INTO archivo (nombre, url, área) VALUES (?, ?, ?);	Call AgregarArchivo(?, ?, ?)
UPDATE migracion SET fecha = '?' WHERE archivo_id = ?;	Call ModificarFechaMigracion (?, ?)

Fuente: Elaboración propia.

Tiempo de ejecución

En la prueba siguiente, se realizaron consultas de complejidad exponencial como se representa en la Tabla 3, con el propósito de determinar cuál consulta ofrece un mejor desempeño, todas las pruebas se ejecutaron una por una para poder medir correctamente el rendimiento.

Tabla 3: Pruebas para la medición del tiempo de ejecución.

Consultas	Inserciones
1 registro x 50 veces	1 registro x 50 veces
200 registros x 20 veces	500 registros x 20 veces
5000 registros x 10 veces	10,000 registros x 3 veces

Se medirá el tiempo de ejecución de cada consulta, desde su inicio hasta su finalización, utilizando el gestor de base de datos que proporciona esta información a través de la clase llamada "profiles".

Integridad de datos

En la siguiente prueba, se realizaron las pruebas de la Tabla 4 de manera individual para detallar los resultados lo más posible. Esto permitió identificar errores y observar los resultados con precisión.

Tabla 4: Pruebas para la integridad de datos.

Tabla de préstamos	Tabla de archivos
200 inserciones	200 actualizaciones

Estos son los criterios que se consideraron en esta prueba:

- Cumplimiento de restricciones de integridad (claves primarias, claves foráneas, campos unitarios, etc.).
- Consistencia en los datos de entrada durante las operaciones de inserción y actualización.
- Validación de campos para garantizar que los datos ingresados cumplan con ciertos requisitos (por ejemplo, formato de fecha válido, valores dentro de un rango específico, etc.).

Usabilidad, escalabilidad y mantenibilidad

En el contexto de las siguientes pruebas, se utilizaron las inserciones de las Tablas 3 permitiendo evaluar los aspectos correspondientes de estas pruebas.

Estos son los criterios que se consideraron en esta prueba:

- Usabilidad: Se examinará la estructura de cada consulta, considerando factores como el uso de subconsultas, la cantidad de uniones (joins), y la presencia de operaciones anidadas, para determinar su impacto en el rendimiento y la facilidad de comprensión.
- Escalabilidad: Se evaluará que tan factible son los métodos de consulta para crecer con el sistema a lo largo del tiempo.
- Mantenibilidad: Se evaluará qué tan fácil es actualizar y modificar las consultas a lo largo del tiempo. Esto incluye la claridad de la lógica de la consulta, la documentación disponible y la modularidad del código.

Seguridad

En la siguiente prueba se evaluó la efectividad de la protección contra inyecciones SQL y manipulación de datos de la Tabla 4. Se intentaron inyecciones SQL maliciosas y manipulación de parámetros para verificar la resistencia a estos ataques y la capacidad de limitar el acceso a datos no autorizados.

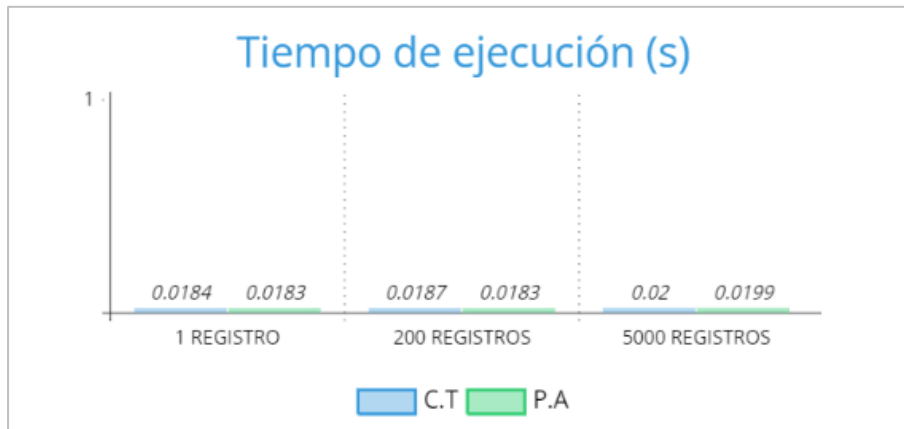
Además, se evaluó la complejidad de las consultas en términos de estructura, lógica y eficiencia para identificar posibles áreas de mejora en la aplicación.



RESULTADOS

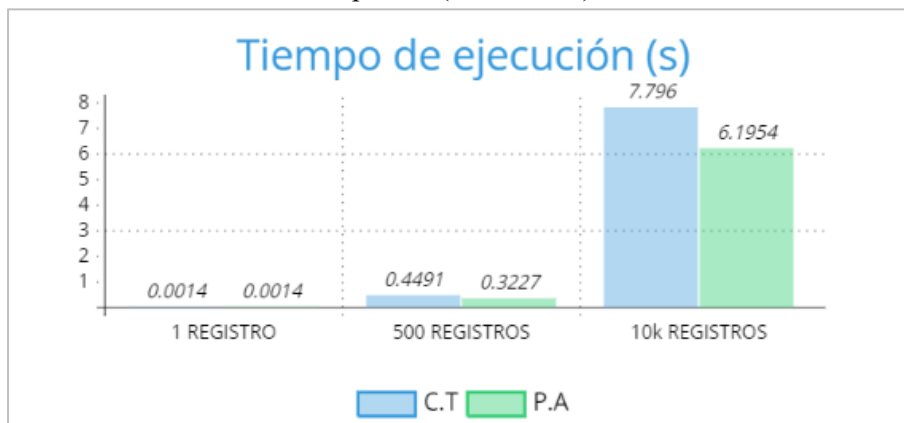
Tiempo de ejecución:

Ilustración 3: Resultados de prueba (Consultas).



En la Ilustración 3 se observa que tanto las consultas tradicionales (C.T) como los procedimientos almacenados (P.A) presentan tiempos de ejecución casi idénticos, independientemente del número de registros. Esto sugiere que el SGBD optimiza de manera similar ambos enfoques, resultando en un rendimiento prácticamente equivalente.

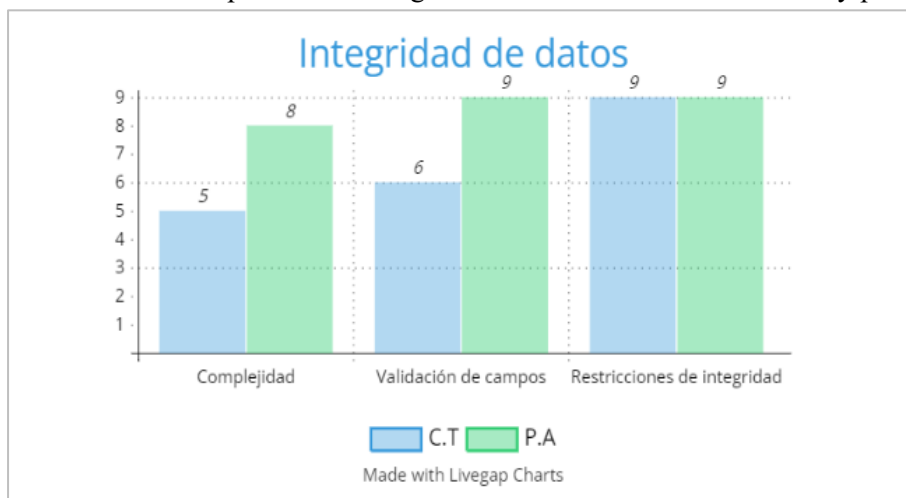
Ilustración 4: Resultados de prueba (Inserciones).



En la Ilustración 4 se puede observar que ambos tipos de métodos exhiben un comportamiento similar en términos de tiempo de respuesta cuando se trabaja con un número moderado de registros. Sin embargo, a medida que aumenta la cantidad de datos, los procedimientos almacenados demuestran un mejor rendimiento en comparación con las consultas tradicionales.

Integridad de datos

Ilustración 5: Comparativa de integridad entre consultas tradicionales y procedimientos almacenados.



Fuente: Elaboración propia.

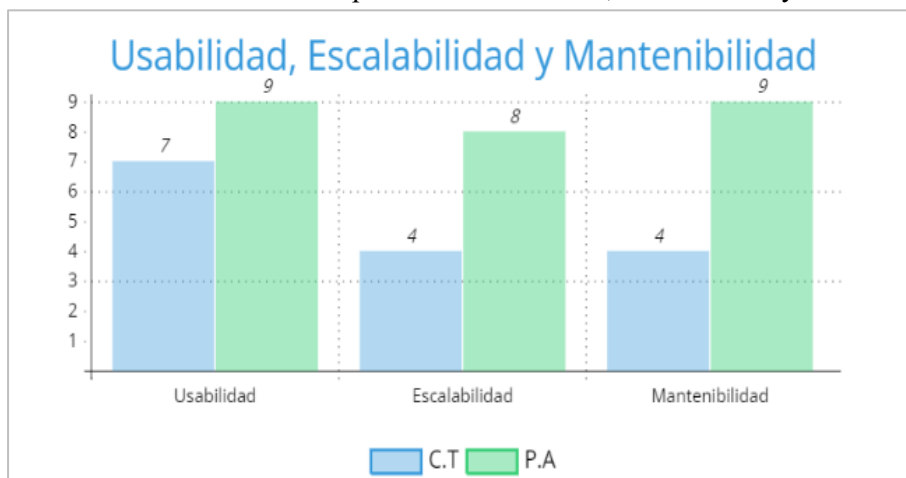
En la Ilustración 5 se representa el desempeño de ambos tipos de consulta en esta prueba y se puede observar que las consultas tradicionales, son más simples y directas de implementar, pero pueden volverse complejas al manejar múltiples restricciones y validaciones. Dependiendo mucho del manejo de errores en la aplicación, esto puede llevar a inconsistencias y dificultades en el seguimiento de errores. Las validaciones suelen ser dispersas entre la aplicación y la base de datos, lo que puede causar redundancias y omisiones. Las restricciones, aunque efectivas con restricciones en la base de datos, pueden ser menos robustas sin apoyo de lógica adicional en la aplicación.

Los procedimientos almacenados, aunque más complejos de escribir y mantener, permiten una lógica de negocio más centralizada y robusto. Ofrecen un manejo de errores más centralizado y coherente, mejorando la capacidad para gestionar excepciones.

Además, pueden incorporar validaciones complejas antes de ejecutar operaciones de datos, garantizando una mayor integridad y reduciendo la duplicidad de lógica. Finalmente, permiten la implementación de verificaciones de integridad adicionales, complementando las restricciones de la base de datos y mejorando la robustez y seguridad de los datos.

Usabilidad, Escalabilidad y Mantenibilidad.

Ilustración 6: Resultados de pruebas de usabilidad, escalabilidad y mantenibilidad.



Fuente: Elaboración propia.

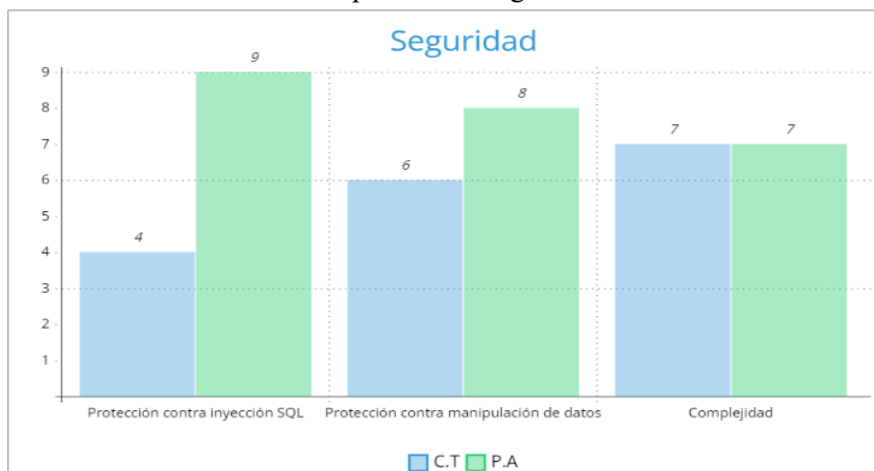
En la Ilustración 6 se muestra el desempeño comparativo de ambos tipos de consulta. Los procedimientos almacenados (P.A) presentan un mejor desempeño en todas las evaluaciones, especialmente en entornos con alta complejidad y cargas de trabajo repetitivas. Esto se debe a que los P.A son sentencias precompiladas, lo que les permite ejecutarse de manera más eficiente sin verse afectados por la frecuencia de ejecución, mejorando el rendimiento y reduciendo el tiempo de procesamiento.

Por otro lado, las consultas tradicionales (C.T) son más eficaces en escenarios de baja complejidad y menos repetitivos. Su simplicidad y flexibilidad permiten una implementación rápida y eficiente en consultas más directas, pero su rendimiento puede verse afectado en tareas más demandantes o cuando se requieren ejecuciones repetidas.

Por lo tanto, la elección entre ambos tipos de consulta dependerá en gran medida de la naturaleza y la complejidad del entorno de aplicación.

Seguridad

Ilustración 7: Resultados de pruebas de seguridad.



Fuente: Elaboración propia

Como se observa en la Ilustración 7 las consultas tradicionales ofrecen flexibilidad, pero pueden ser vulnerables a inyecciones SQL si no se implementan medidas adecuadas como sentencias preparadas. Además, al permitir mayor control directo sobre los datos, están más expuestas a manipulaciones accidentales o intencionales, lo que requiere un manejo cuidadoso. Aunque son más fáciles de escribir y gestionar en entornos simples, su mantenimiento se vuelve complicado a medida que crece la complejidad de las consultas.

Los procedimientos almacenados, al ser precompilados, proporcionan una mejor protección contra inyecciones SQL, lo que los hace más seguros en entornos con alta frecuencia de ejecución. Su estructura encapsulada también reduce las posibilidades de manipulación directa de los datos, brindando mayor control y seguridad. Aunque requieren más esfuerzo inicial en su desarrollo, resultan más fáciles de mantener y escalar en sistemas grandes y complejos, lo que los convierte en una opción ideal para entornos de alta complejidad.

CONCLUSIONES

En conclusión, este estudio comparativo fue desarrollado con el propósito de destacar los puntos fuertes y débiles de las consultas tradicionales y los procedimientos almacenados en áreas clave como tiempo de ejecución, integridad de datos, escalabilidad, usabilidad, mantenibilidad y seguridad, contribuyendo así a una mejor toma de decisiones para la implementación futura en los módulos de Archivos y Préstamos del sistema de CECyTEG.

Se ha demostrado que los procedimientos almacenados (P.A) presentan ventajas significativas frente a las consultas tradicionales (C.T) en aspectos como seguridad, escalabilidad y mantenibilidad, particularmente en entornos con alta demanda y complejidad. No obstante, en cuanto al tiempo de respuesta, ambos métodos ofrecieron un rendimiento muy similar, con la excepción de escenarios de mayor carga, donde los P.A mostraron una ligera ventaja.

Asimismo, los P.A, al ser precompilados, brindan mayor protección frente a inyecciones SQL y permiten una gestión más eficiente de operaciones repetitivas y complejas. En contraste, las C.T sobresalen por su simplicidad y efectividad en consultas menos complejas, siendo una opción adecuada para situaciones con menor demanda o consultas puntuales. La elección entre ambos enfoques debe ajustarse a los requisitos específicos del sistema, priorizando la seguridad y eficiencia en sistemas más robustos y la flexibilidad en contextos menos exigentes.

De cara al futuro, optar por procedimientos almacenados podría ser clave para mejorar la eficiencia y seguridad en los demás módulos del sistema del CECyETG.

REFERENCIAS BIBLIOGRAFICAS

- Antònia Mas, E. A. (2005). La mejora de los procesos de software en las pequeñas y medianas empresas (pyme). Un nuevo. *REICIS. Revista Española de Innovación*, 1(2), 7-29.
- Beaulieu, A. (2020). *Learning SQL*. O'Reilly Media.
- Ben-Gan, I. (2020). *SQL Server 2019: Querying*. Microsoft Press.
- Date, C. (2020). SQL and Relational Theory: How to Write Accurate SQL Code. En C. Date, *SQL and Relational Theory: How to Write Accurate SQL Code*. O'Reilly Media.
- Dávila, C. (2017). *SQL para todos*. RC Libros.
- Fernández, J. S. (2006). *Administración de bases de datos con MySQL*. RA-MA Editorial.
- Forta, B. (2019). SQL en 10 minutos. En B. Forta, *SQL en 10 minutos*. Pearson Education.
- Gil, J. (2021). Introducción a los procedimientos almacenados en SQL Server. *Revista de Ingeniería y Computación*, 10(1), 123-134.
- González, E. C. (Junio de 2017). Desarrollo de una notación y herramienta de visualización de datos para mejorar la capacidad de análisis de procedimientos almacenados SQL. *Desarrollo de una*

notación y herramienta de visualización de datos para mejorar la capacidad de análisis de procedimientos almacenados SQL. Costa Rica: Tesis. Multimedia

González, J. R. (2003). *SQL: El lenguaje universal de las bases de datos.* UNED.

Kellenberger, K. (2022). *SQL. Guía práctica para el programador.* Anaya.

Kevin Kline, L. G. (1999). *Transact-SQL Programming.* O'Reilly Media.

López, A. (2020). Optimización de consultas SQL: Buenas prácticas. *Revista Iberoamericana de Tecnologías de la Información*, 15(3), 45-58.

Martín, L. (2019). Cómo mejorar el rendimiento de las consultas SQL. *Anales de Informática y Tecnología*, 8(2), 87-99.

Muhammad Zohaib Khan, F. U. (2022). Comparative Case Study: An Evaluation of Performance Computation Between SQL And NoSQL Database. *SJHSE Sindh Journal of Headways in Software Engineering*, 1(2).

Navathe, R. E. (2017). Bases de datos: Fundamentos y aplicaciones. En R. E. Navathe, *Bases de datos: Fundamentos y aplicaciones* (p. 900). Addison-Wesley.

Pérez, A. (2018). Procedimientos almacenados en MySQL: Ventajas y usos. *Revista Técnica de MySQL y Base de Datos*, 6(4), 112-125.

Randolph West, W. A. (2019). *SQL Server 2019 Administration Inside Out.* Microsoft Press.

Septiadi, A. D. (2020). Comparative Analysis of Database Query Storage Performance. *International Journal of Informatics and Information System*, 3(2), 66.

Walther. (24 de agosto de 2023). *SQL vs NoSQL: Ventajas y Desventajas Detalladas de Cada Sistema.*
Dongee: <https://www.dongee.com/tutoriales/sql-vs-nosql-ventajas-y-desventajas-detalladas-de-cada-sistema/>

Ward, B. (2019). SQL Server 2019 Revealed. En B. Ward, *SQL Server 2019 Revealed.* Apress .

Winand, M. (2012). *SQL Performance Explained.* Markus Winand.