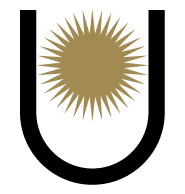
Universidad Nacional del Nordeste



Facultad de Ciencias Exactas y Naturales y Agrimensura

Licenciatura en Sistemas de Información

Base de datos l

Año: 2023

**Índices columnares SQL**

Comisión 1 grupo 5

Alumno: Bournisent, Matias

L.U. N°: 50.434

D.N.I. N°: 39.126.075

Alumno: Comba, Carlos Alfredo

L.U. N°: 44.055

D.N.I. N°: 32.335.478

Alumno:Ramirez Gonzalo Daniel

L.U. N°: 56.838

D.N.I. N°: 44.543.439

Alumno: Zini, Franco Joaquin

L.U. N°: 50.717

D.N.I. N°: 40.049.028

**Índice**

[**Capítulo I: INTRODUCCIÓN 3**](#_30j0zll)

[**Capítulo II: Marco teórico 5**](#_3znysh7)

[**Capítulo III: Metodología 9**](#_2et92p0)

[**Capítulo IV:Desarrollo/Resultados 11**](#_tyjcwt)

[**Capítulo V: Conclusión 12**](#_3dy6vkm)

# **Capítulo I: INTRODUCCIÓN**

En el ámbito de la materia Base de Datos I, se nos asignó la tarea de investigar y experimentar con un componente fundamental del diseño de bases de datos: los índices columnares. El enfoque se centra en analizar cómo los índices impactan en el rendimiento del sistema, utilizando un caso de estudio concreto.

Partiendo de un modelo de base de datos proporcionado por la cátedra, llamado "base\_consorcio", nuestra tarea es duplicar la tabla 'gasto', creando así una nueva tabla denominada 'gastoNew'. La clave de nuestro estudio es la implementación de índices columnares en esta nueva tabla, lo que nos permitió investigar cómo estos índices aceleran la recuperación de datos.

Ambas tablas se cargaron con un millón de registros para crear un escenario realista de carga de trabajo. Nuestro objetivo es evaluar y comparar las diferencias de rendimiento entre las búsquedas realizadas en una tabla con índices columnares ('gastoNew') y una sin ellos ('gasto'), para entender la mejora que ofrecen estos índices.

Este informe proporcionará una visión detallada de nuestra metodología, incluyendo los fundamentos teóricos de los índices columnares, los scripts utilizados para modificar las tablas y realizar las inserciones, así como los scripts de búsqueda. Además, se detalla el entorno técnico en el que se llevaron a cabo las pruebas, incluyendo el motor y gestor de la base de datos utilizados, así como las especificaciones de las máquinas que alojan nuestras pruebas.

Los resultados de nuestras pruebas incluyen estadísticas precisas sobre la duración de las consultas, el consumo de memoria, el uso de la CPU y el número de filas recuperadas. Estos datos ofrecerán una comprensión profunda y cuantificable del impacto de los índices columnares en el rendimiento del sistema, proporcionando una base sólida para nuestras conclusiones finales.

# **Capítulo II: Marco teórico**

En la gestión de bases de datos, el uso de índices columnares optimiza el rendimiento. Son estructuras de datos especializadas que permiten una recuperación eficiente de información en grandes conjuntos de datos. En lugar de buscar a través de todas las filas de una tabla, los índices columnares almacenan y ordenan los valores de una columna específica, actuando como rutas de acceso rápidas hacia los datos que los usuarios buscan.

Cuando se ejecutan consultas en una base de datos, estos índices permiten una búsqueda mucho más rápida y, por ende, reducen significativamente el tiempo necesario para recuperar resultados. Esto se traduce en un rendimiento general del sistema notablemente mejorado, especialmente cuando se trata de operaciones que implican grandes volúmenes de datos.

Cuando se crea un índice columnar para una columna específica, la base de datos organiza los valores de esa columna en una estructura de datos optimizada:

* **Árboles B:** Los índices columnares basados en árboles B son estructuras de datos de tipo árbol que permiten búsquedas eficientes, inserciones y eliminaciones. Estos árboles están diseñados para minimizar la altura del árbol, lo que vuelve la búsqueda más rápida.
* **Árboles B+:** Son una variante de los árboles B que mejora aún más la eficiencia en las operaciones de búsqueda. Los nodos internos de un árbol B+ contienen solo claves para la navegación, lo que aumenta la capacidad de almacenamiento de las claves y mejora la velocidad de búsqueda.
* **Estructuras de Bitmap:** Los índices columnares basados en estructuras de bitmap son eficaces para columnas que tienen un número limitado de valores únicos (por ejemplo, género o estado civil). Cada valor único se asigna a un conjunto de bits, y las operaciones de búsqueda implican operaciones lógicas en estos conjuntos de bits, lo que permite búsquedas rápidas y eficientes.
* **Índices Hash:** Los índices hash se basan en funciones hash que asignan valores de columna a ubicaciones específicas en una estructura de datos hash. Esto permite búsquedas rápidas cuando se conoce el valor exacto de la columna. Sin embargo, los índices hash no son tan eficientes para operaciones como el rango o las comparaciones de patrones.
* **Árboles Trie:** Estos árboles son útiles para índices columnares en datos de texto o cadenas. Los árboles Trie almacenan información de texto de manera jerárquica, lo que facilita la búsqueda y comparación de palabras o patrones de texto.

La elección de la estructura asociada a un índice columnar depende del tipo de datos en la columna y del tipo de operaciones que se realizarán con esa columna (por ejemplo, búsquedas exactas, búsquedas de rango, etc.). Cada estructura tiene sus ventajas y desventajas, por lo que se selecciona según los requisitos específicos de la aplicación. El motor de base de datos toma la decisión de qué estructura de índice utilizar. Los DBMS están diseñados para ser sistemas inteligentes y autónomos que optimizan el rendimiento de las consultas y operaciones en función de la estructura de los datos y el tipo de consultas que se realizan.

Ventajas de los índices columnares:

Los índices columnares ofrecen una serie de ventajas sobre los índices tradicionales, entre las que se incluyen:

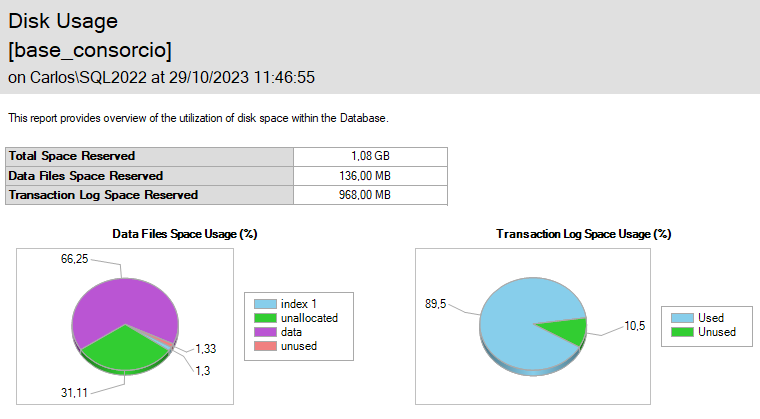
* Mejor rendimiento para consultas que requieren la comparación de valores en una sola columna.
* Reducción del uso de memoria del servidor.
* Mejor compresión de datos.
* Reducción de la fragmentación de datos.

Los índices columnares no son ideales para todos los entornos, por ejemplo estos no son ideales en el ámbito de procesamiento de transacciones en línea (OLTP) donde se realizan operaciones de inserción, actualización y eliminación con alta frecuencia. Están orientados para las situaciones en las que las operaciones de lectura son más frecuentes comparadas a las de escritura. Otra desventaja que presentan Los índices columnares es que pueden requerir más espacio de almacenamiento que los índices tradicionales basados en filas, especialmente durante el proceso de carga de datos, debido a esto el costo de almacenamiento puede incrementarse enormemente, también el uso de índices columnares puede requerir recursos adicionales del sistema, como CPU y memoria, para su procesamiento. En entornos con recursos limitados, esto puede ser un factor a considerar.

Los índices columnares se utilizan a menudo en situaciones donde se requiere un alto rendimiento para consultas analíticas y agregaciones en bases de datos que manejan grandes volúmenes de datos, las más populares son:

* Análisis de datos (OLAP)
* Data warehousing:
* Business intelligence
* Big data

Falta citas de autores bibliograficas.



# **Capítulo III: Metodología**

Sobre la base del modelo que entregó la cátedra. La estructura de base de datos y dos lotes de carga.

Donde se ejecuta:

1. ModeloDatos\_Consorcio.sql: este script crea la base de datos y sus tablas.
2. BDI\_loteDatosConsorcios.sql: permite cargar los datos de provincias, localidad, zona, conserje, administracion, consorcio, tipogasto, gasto.
3. BDI\_tabla\_inmueble\_completo.sql: crea y carga la tabla inmueble.
4. columnstore\_index\_script.sq: crea la tabla gastonew, crea los indice columnares, se insertan 1 millón de registros, y se carga la tabla gasto con los registros faltantes para llegar al millón y poder hacer las pruebas comparativas.
5. script\_busqueda.sql: tiene los select de búsqueda para ambas tablas.

Cada estudiante ejecutará el script de busqueda.sql y anotará sus resultados en la tabla de referencia. Se realizarán 4 test por búsqueda, sobre un total de 5 búsquedas que consideramos que son las que pueden mostrar diferencia en el uso de índices columnares.

Herramientas

Elementos comunes a todos:

* Script en Transact-SQL (Es el lenguaje utilizado para interactuar y gestionar bases de datos en el sistema de gestión de bases de datos Microsoft SQL Server.)

Elementos usados por Comba Carlos para las pruebas:

* Motor de base de datos SQL Server 2022 standard versión 16.0.1000.6
* Sql Managament Studio 19.1.56
* Máquina i7 11gen, 32 gb ram, 1tb ssd.

Elementos de prueba usado por Ramirez Gonzalo para las pruebas:

* Motor de base de datos SQL Server 2022
* Sql Managament Studio 19.1.56
* Equipo: Ryzen 3 3100, 16 gb de ram, Ssd 240GB.

# **Capítulo IV:Desarrollo/Resultados**

En esta sección, haremos foco en el análisis de las pruebas de rendimiento realizadas en consultas que involucran el uso de índices columnares en comparación con aquellas que no los utilizan. Nuestra principal misión es evaluar y presentar resultados concretos que permitan comprender el impacto de estos índices en términos de tiempo, costos y rendimiento. Las pruebas son realizadas en distintos escenarios para así comprobar el desempeño de los índices columnares en situaciones variadas. Estos escenarios pueden incluir consultas que involucran un gran volumen de datos, consultas de agregación, búsquedas de información y más. Esto nos permitirá obtener una visión completa de cómo los índices columnares se desempeñan en diversas circunstancias.

| Alumno | test | tabla | tiempo de respuesta | Cpu cost | row | subtree cost | i/o cost | operator cost |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Comba Carlos | test1 | gastonew |  |  |  |  |  |  |
|  | test1 | gastonew |  |  |  |  |  |  |
|  | test1 | gastonew |  |  |  |  |  |  |
|  | test1 | gastonew |  |  |  |  |  |  |
| Comba Carlos | test1 | gasto |  |  |  |  |  |  |
|  | test1 | gasto |  |  |  |  |  |  |
|  | test1 | gasto |  |  |  |  |  |  |
|  | test1 | gasto |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |

| **Alumno** | **Test** | **Tabla** | **Tiempo de respuesta** | **subtree cost** |
| --- | --- | --- | --- | --- |
| Ramirez  Gonzalo | Test 1 | gasto | 220 ms | 5,91 |
|  | Test 1 | gastonew | 7 ms | 0,24 |
|  | Test 2 | gasto | 223 ms | 5,91 |
|  | Test 2 | gastonew | 7 ms | 0,24 |
|  | Test 3 | gasto | 134 ms | 6,39 |
|  | Test 3 | gastonew | 3 ms | 0,20 |
|  | Test 4 | gasto | 5 seg | 5,3 |
|  | Test 4 | gastonew | 3 seg | 0,42 |
|  | Test 5 | gasto | 2 seg | 7,2 |
|  | Test 5 | gastonew | 1 seg | 1,35 |
|  |  |  |  |  |

# **Capítulo V: Conclusión**

Los índices columnares en SQL Server son una solución clave para optimizar el rendimiento de las bases de datos, lo que se traduce en una mayor eficiencia operativa y en la capacidad de tomar decisiones más informadas. Su implementación puede marcar la diferencia en la experiencia de los usuarios y en el éxito de las aplicaciones y sistemas que dependen de la gestión eficaz de datos, la implementación de índices columnares es una estrategia valiosa para las organizaciones que buscan maximizar el valor de sus bases de datos. Al enfocarse en columnas específicas en lugar de en la tabla completa, los índices columnares permiten un acceso más rápido a la información relevante

**Capitulo VI: Bibliografía**

1. SQL Performance Explained" por Markus Winand
2. Fundamentos de Base de Datos" de Korth H. y Silberschatz A. (1993, McGraw-Hill, ISBN: 84-481-0079-4)
3. Microsoft. (2023). Introducción a los índices de almacén de columnas.