|  |
| --- |
| **OPTIMIZADOR DE ACCESOS Y TRANSNMICIÓN DE DATOS** |
| **201901772 – Daniel Reginaldo Dubón Rodríguez** |

**Resumen**

El ensayo presenta la solución a un problema de distribución en una base de datos donde el costo total de acceso y transmisión de datos para el procesamiento es demasiado alto, se soluciona el problema obteniendo un nuevo esquema replicado de alojamiento adaptando un nuevo patrón de uso de la base de datos que minimiza los costos de acceso y transmisión, aplicando una metodología de agrupamiento, dado que los datos se pueden representar como una matriz de “n” filas y “m" columnas, el método consiste en obtener matrices de frecuencias de acceso y agrupar todas las tuplas que repitan el mismo patrón dando como resultado una matriz reducida de frecuencias de accesos, representando el nuevo esquema para el nuevo patrón de uso de la base de datos. Para la solución se optó por el uso de un lenguaje de programación para crear los algoritmos requeridos para realizar la metodología antes mencionada.

**Palabras clave**

Python, Tipo de Dato Abstracto, Lenguaje de Marcado Extensible, Programación Orientada a Objetos, Graphviz.

***Abstract***

*The essay presents the solution to a distribution problem in a database where the total cost of access and data transmission for processing is too high, the problem is solved by obtaining a new replicated hosting scheme adapting a new pattern of use of the database that minimizes access and transmission costs, applying a grouping methodology, since the data can be represented as a matrix of “n” rows and “m” columns, the method consists of obtaining matrices of access frequencies and grouping all the tuples that repeat the same pattern resulting in a reduced matrix of access frequencies, representing the new scheme for the new pattern of use of the database. For the solution, the use of a programming language was chosen to create the algorithms required to carry out the aforementioned methodology.*

***Keywords***

*Python, Abstract Data Type, Extensible Markup Language, Objetct Oriented Programming, Graphviz.*

**Introducción**

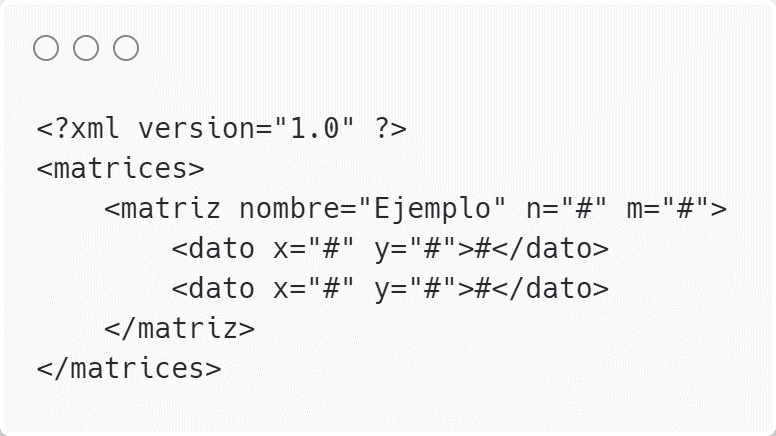
Se realizó un optimizador de accesos y transmisión de datos, con el cual se obtiene un nuevo esquema replicado de alojamiento adaptando un nuevo patrón de uso de la base de datos que minimiza los costos de acceso y transmisión, se realizó a través del lenguaje de programación Python para construir los diferentes algoritmos para obtener la solución así mismo se hizo uso del paradigma de programación orientada a objetos para abstraer el problema y lograr plasmarlo en el lenguaje de programación, también se usaron tipos de datos abstractos para crear una estructura en la cual se almacena los datos a procesar así mismo se hizo uso de la herramienta graphviz para crear el diseño de diagrama y generar una gráfica en el cual se pueden visualizar los datos obtenidos de un archivo con estructura XML.

**Desarrollo del tema**

El problema planteado consiste en alojar objetos de bases de datos en sitios distribuidos, de manera que el costo total de la transmisión de datos para el procesamiento de todas las aplicaciones sea minimizado.

Se ha optado el uso del lenguaje de programación Phyton debido a que es multiplataforma y por su facilidad de uso, además también es soporta varios paradigmas de programación.

Para la elaboración de la solución se provee un archivo con extensión y estructura xml, el cual trae información y modelo de los patrones de acceso que se desea optimizar y este archivo tiene la estructura de la siguiente forma:



*Figura 1.* Estructura del archivo XML dado.

Fuente: elaboración propia

donde:

matrices = es la etiqueta padre.

matriz=es la etiqueta que indica una nueva matriz creada para el respectivo análisis.

nombre = es el identificador de la matriz.

n = indica las filas que conformara la matriz.

m = indica las columnas que conformara la matriz

x = indica la fila de la matriz.

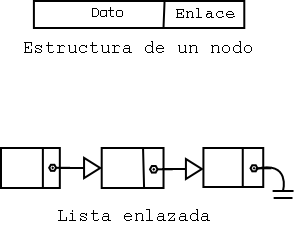
y = indica la columna de la matriz.

# = indica un valor numérico

Obtenido ese formato se necesitaba de una estructura en la cual se guardarían la información del archivo dado, por lo cual se llevó al análisis del uso de un tipo de dato abstracto ya que permite el uso de memoria dinámica ya que el programa lo requiere debido a que no se sabe con certeza la cantidad de datos que puede contener.

Los tipos de datos abstractos implementados para la solución fueron, lista simplemente enlazada y lista enlazada circular.

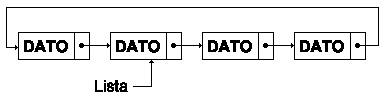
En la lista simple enlazada se emulo una matriz que contendrá “n” filas y “m” columnas, para el almacenamiento de los datos, se opto por esta estructura debido a su fácil implementación y utilización.



*Figura 2.* Diagrama de una lista enlazada simple.

Fuente: Diana Molina.

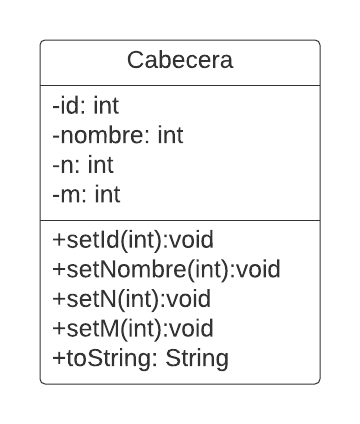
En la lista enlazada circular se uso para almacenar la información de cada matriz junto a sus atributos, esta lista también hace referencia a la lista simple enlazada donde se referencia con los datos que se compone la matriz.



*Figura 3.* Diagrama de una lista circular enlazada.

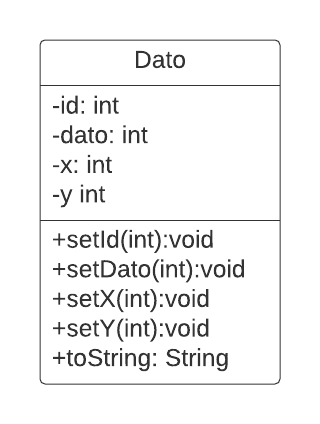
Fuente: Alfonzo Rodríguez.

Con el paradigma de programación orientada a objetos se emulo los diferentes tipos de datos para almacenarlos en la estructura de datos correspondientes, esto abrió la posibilidad de trabajar más eficientemente y optimización de memoria del programa ya que se tienen varios datos y se necesitan agruparlos entre ellos para su manipulación.



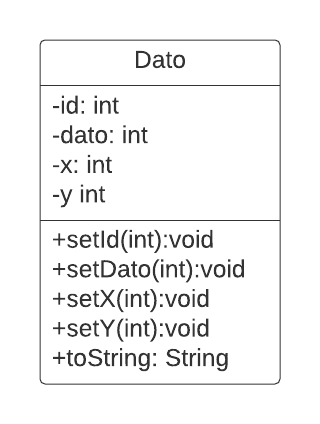
*Figura 4.* Modelo del tipo de dato Cabecera.

Fuente: Elaboración propia.



*Figura 5.* Modelo del tipo de dato Dato.

Fuente: Elaboración propia



**Conclusiones**

Esta sección debe orientarse a evidenciar claramente las principales ideas generadas, propuestas que deriven del análisis realizado y si existen, expresar las conclusiones o aportes que autor quiera destacar.

Enfatizando, lo importante es destacar las principales posturas fundamentadas del autor, que desea transmitir a los lectores.

Adicionalmente, pueden incluirse preguntas abiertas a la reflexión y debate, temas concatenados con el tema expuesto o recomendaciones para profundizar en la temática expuesta.

**Referencias bibliográficas**

C. J. Date, (1991). *An introduction to Database Systems.* Addison-Wesley Publishing Company, Inc.