Universidad de San Carlos de Guatemala
Escuela de Ingeniería en Ciencias y Sistemas, Facultad de Ingeniería
Introducción a la programación y computación 2, 1er. Semestre 2021.

Proyecto 1

201900810 – Jorge Antonio Pérez Ordóñez

Resumen

Se determinó que una solución para minimizar los costos de acceso y comunicación de un sistema de consultas de bases de datos en varios sitios es la metodología de agrupación con matrices de frecuencias de acceso.
El programa recibirá archivos XML de entrada con un set de matrices de n filas y m columnas con un set de datos los cuales serán las frecuencias de acceso. Se recolectarán los datos de las matrices y estas se almacenarán en una lista circular simplemente enlazada. Luego se procesarán por medio del método de agrupación para obtener la matriz reducida de frecuencias de acceso. Después el usuario tendrá la opción de elaborar un archivo XML de salida con la nueva matriz de frecuencia reducida en la ruta que desee.
Si el usuario lo desea, podrá elegir cualquier matriz cargada para hacer un reporte de ella para visualizar la matriz de una mejor manera en forma de gráfica.

Palabras clave

Archivos XML
Matriz de frecuencia reducida
Lista circular

Abstract

It was determined that a solution to minimize the access and communication costs of a multi-site database query system is the clustering methodology with access frequency matrices.
The program created input XML files with a set of matrices of n rows and m columns with a set of data that will be the access frequencies. The data will be collected from the arrays and these will be stored in a simply linked circular list. It will then be processed using the clustering method to obtain the reduced access frequency matrix. The user will then have the option of producing an output XML file with the new reduced frequency matrix in the path of their choice.
If the user wishes, he can choose any loaded matrix to make a report of it to better visualize the matrix in the form of a graph.

Keywords

XML Files
Reduced Frequency Matrix
Circular List

Introducción

Se requiere guardar objetos de bases de datos en sitios, de tal manera que se minimice el costo de la transmisión de datos para el procesamiento de todas las aplicaciones.

La dinámica consiste en que un set de consultas llegara a un set de sitios, los cuales interactúan con la base de datos en forma de objetos. Se crea un esquema inicial de alojamiento de objetos de bases de datos, y las frecuencias de acceso de cada consulta desde cada sitio en un período de tiempo. El objetivo del proyecto es determinar un esquema que se adapte a un nuevo patrón de uso de la base de datos y minimice los costos de transmisión. Una posible solución es la metodología de agrupamiento, que consiste en utilizar matrices binarias para obtener nuevas matrices reducidas.

Desarrollo del tema

Tipos de Datos Abstractos (TDA)

Un Tipo Abstracto de Datos es un conjunto de valores y de operaciones definidos mediante una especificación independiente de cualquier representación. La manipulación de un TDA sólo depende de su especificación, nunca de su implementación. Se pueden implementar los TDA sólo a partir de la especificación, sin saber para qué se van a usar.

Una de las técnicas para manejar tipos de datos abstractos en programación son las listas enlazadas. Las listas enlazadas son conjuntos de elementos en los que cada elemento contiene la posición. Cada elemento de la lista enlazada debe tener al menos dos campos: un campo que tiene el valor del elemento y un campo (enlace) que contiene la posición del siguiente elemento, es decir, su conexión, enlace o encadenamiento. Los elementos de una lista son enlazados por medio de los campos enlaces. De tal manera que para hacer referencia a el siguiente elemento en la lista se debe dirigir a la propiedad enlace (Fernández, 2013).

Una lista circular es una lista enlazada en la cual el ultimo elemento de la lista esta enlazado con el primero, de tal manera que no tiene fin. La ventaja de utilizar listas circulares es que cada elemento de una lista circular es accesible desde cualquier otro elemento de ella. Es decir, dado un elemento se puede recorrer toda la lista completa. En una lista enlazada de forma simple sólo es posible recorrerla por completo si se parte de su primer elemento. Además, Las operaciones de concatenación y división de listas son más eficaces con listas circulares. Sin embargo, el inconveniente de utilizar listas circulares es que se pueden producir lazos o bucles infinitos. Una forma de evitar estos bucles infinitos es disponer de un nodo especial que se encuentre permanentemente asociado a la existencia de la lista circular. Este nodo se denomina cabecera de la lista.

Archivos XML

Un archivo XML es un archivo de lenguaje de marcado extensible y se utiliza para estructurar datos para su almacenamiento y transporte. En un archivo XML, hay etiquetas y texto. Las etiquetas proporcionan la estructura y propiedades a los datos por medio de atributos. El texto del archivo que desea almacenar está rodeado por estas etiquetas, que se adhieren a pautas de sintaxis específicas. En esencia,

un archivo XML es un archivo de texto estándar que utiliza etiquetas personalizadas para describir la estructura del documento y cómo debe almacenarse y transportarse. Su propósito es principalmente guardar información estructurada (Juviler, 2020).

En Python se puede leer archivos XML por medio de librerías específicas que ayudan a extraer la información de las etiquetas. La librería ElementTree (ET) es una de las más populares para leer xml en Python. XML es un formato de datos basicamente jerárquico y la forma más natural de representarlo es con un árbol. ET tiene dos clases para este propósito: ElementTree representa todo el documento XML como un árbol y Element representa un solo nodo en este árbol. Las interacciones con todo el documento generalmente se realizan en el nivel ElementTree. Las interacciones con un solo elemento XML y sus subelementos se realizan en el nivel del elemento (Python Software Foundation, 2021).

Metodología de Agrupamiento

La metodología de agrupamiento sirve para determinar el alojamiento de datos de forma que los costos de acceso y comunicación son minimizados. Como muchos otros problemas reales, es un problema combinatorio NP-Hard. Algunas de las situaciones comunes que hemos observado cuando se resuelven instancias muy grandes de un problema NP-Hard son: Fuerte requerimiento de tiempo y fuerte demanda de recursos de memoria.

Para “nt” tuplas y “ns” sitios, el método consiste en tener la matriz de frecuencia de acceso en los sitios F[nt][ns] de la instancia objetivo, transformarla en una matriz de patrones de acceso y agrupar las tuplas con el mismo patrón. El patrón de acceso para una tupla es el vector binario indicando desde cuál sitio la tupla es accedida

Por ejemplo, en la siguiente matriz con n=5 filas y m=4 columnas.

2	3	0	4
0	0	6	3
3	4	0	2
1	0	1	5
0	0	3	1

Fuente: Elaboración propia

Se debe obtener el vector binario reemplazando con 1s en los lugares en donde hay números diferentes a cero.

1	1	0	1
0	0	1	1
1	1	0	1
1	0	1	1
0	0	1	1

Fuente: Elaboración propia

Con la matriz del vector binario se pueden ubicar las filas con el mismo patrón de acceso. En este caso las filas que comparten un patrón de acceso son la fila 1 con la 3 y la fila 2 con la 4. De manera que, al sumar las filas con el mismo patrón de acceso en la matriz original, se obtiene la siguiente matriz.

5	7	0	6
0	0	9	4
1	0	1	5

Fuente: Elaboración propia

Conclusiones

Las listas circulares son listas enlazadas en las que el último elemento de la lista esta apuntando al primer elemento de la lista y la ventaja de utilizar este tipo de lista es que cualquier elemento puede ser accesible desde cualquier otro elemento.

Los archivos XML son archivos en los que se guarda información estructurada por medio de etiquetas con atributos y sirven para describir la estructura de documentos y cómo deben almacenarse y transportarse.

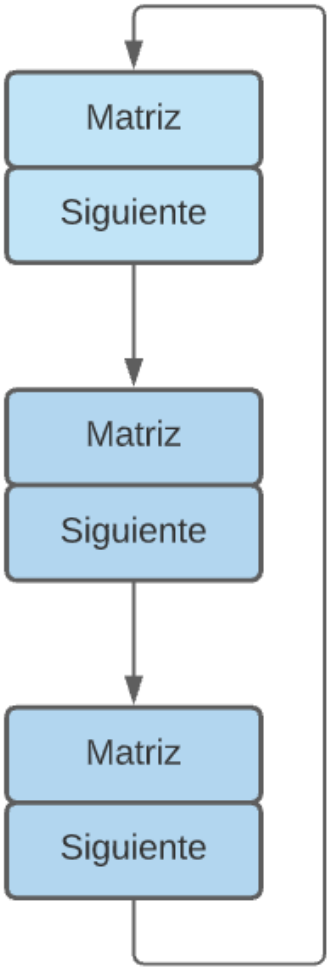
La metodología de agrupamiento es una técnica útil para determinar el alojamiento de datos en sitios distribuidos reduciendo el requerimiento de tiempo y demanda de memoria.

Referencias bibliográficas

- Fernández, J. (25 de Octubre de 2013). *sites.google.com/site/programacioniiuno*. Obtenido de https://sites.google.com/site/programacioniiuno/temario/unidad-2---tipo-abstracto-de-dato/tipo-de-dato-abstracto
- Juviler, J. (27 de Julio de 2020). *blog.hubspot.com*. Obtenido de https://blog.hubspot.com/website/what-is-xml-file
- Python Software Foundation. (08 de Marzo de 2021). *docs.python.org*. Obtenido de https://docs.python.org/3/library/xml.etree.elementtree.html

Anexo

Lista circular de matrices



Fuente: Elaboración propia