

**Universidad de Buenos Aires**

**Facultad de Ingeniería**

**Departamento de Informática**

***Organización de Datos (75.06)***

**Voto Electrónico**

**Documentación General**

Cuatrimestre y año: 2do Cuatrimestre 2011

Docente a cargo del TP:Nicolás Pablo Fernández Theillet

Grupo: Lamas

Fecha de Entrega:

Integrantes: 2011-10-22

|  |  |  |
| --- | --- | --- |
| ***Padrón*** | ***Nombre*** | ***Email*** |
| 91187 | Gonzalez Durand, Juan Manuel | jmanuel.gonzalez.durand@gmail.com |
| 90762 | Ostrowsky, Gabriel | gaby.osto@gmail.com |
| 90728 | Schenkelman, Damián | damian.schenkelman@gmail.com |
| 91045 | Torrado, Alejandro | aletorrado@gmail.com |
| 90884 | Zamudio, Gonzalo | ahogadosderazon@gmail.com |

# Especificación de Clases

La siguiente sección provee la documentación de las clases utilizadas en la solución implementada. En la misma, se detallan el nombre de la clase, la carpeta en la que se encuentra y una breve explicación del rol de la clase.

|  |
| --- |
| **Nota:** Esta sección no considera a las clases utilizadas para pruebas. |

**Carpeta: BPlusTree**

En esta carpeta se guardan las clases que están específicamente relacionadas con el funcionamiento del árbol B+.

|  |  |
| --- | --- |
| **Clase** | **Especificación** |
| **FreeBlockManager** | Es utilizada para mantener actualizada la lista de bloques libres en un archivo separado al de datos. La misma utiliza un conteo incremental, siempre que no haya bloques que hayan sido liberados, en cuyo caso recurre a ellos primero. |
| **IndexTreeBlock** | Hereda de TreeBlock. Representa un bloque de la zona índice del árbol B+. Es usada por InternalNode. Guarda punteros a nodos hijo e índices (usando claves) hacia ellos. |
| **InternalNode** | Hereda de Node. Representa un nodo interno del árbol e implementa las operaciones de lectura, alta, baja y modificación de forma recursiva (llama a la función para el hijo determinado según el valor de la clave). |
| **LeafNode** | Hereda de Node. Representa un nodo hoja del árbol y todas las operaciones de lectura o escritura finalmente actúan sobre ellos. |
| **Node** | Representa, de forma abstracta, un nodo del árbol. Provee una interfaz común para operaciones como alta, baja, modificación, lectura y agrupa funcionalidad común a nodos hoja e internos. |
| **OverflowParameter** | Es utilizada para guardar información cuando un bloque entra en overflow. Se pasa como parámetro auxiliar a la función de inserción, para que los padres tengan información para actualizarse correctamente ante el split de sus hijos. |
| **SequenceTreeBlock** | Hereda de TreeBlock. Representa un bloque de la zona índice del árbol B+. Es usada por Leaf. Guarda los registros de datos de longitud variable y un puntero al nodo siguiente, para permitir el recorrido secuencial. |
| **Tree** | Agrupa la funcionalidad del árbol. De esta forma, se abstrae al usuario de la implementación a niveles más bajos del árbol y se provee una interfaz de alto nivel la cual es fácil de utilizar. Guarda un puntero a la raíz, y se encarga de guardar el último bloque leído. |
| **TreeBlock** | Representa, de forma abstracta, un bloque del árbol. Provee una interfaz común para operaciones como alta, baja, modificación, lectura y agrupa funcionalidad común a bloques de índice y secuencia. |
| **TreeBlockFile** | Hereda de BaseVariableBlockFile. Es utilizada por Tree y los nodos para cargar y guardar bloques de longitud variable al archivo. Adicionalmente, al momento de descender por el árbol para las distintas operaciones, y a fin de mantener los bloques del actual recorrido en memoria, provee una pila para ir acumulando a los mismos en orden. |

**Carpeta: Indexes**

En esta carpeta se guardan las clases relacionadas con los índices adicionales a los archivos de datos requeridos para diferentes funcionalidades.

|  |  |
| --- | --- |
| **Clase** | **Especificación** |
| **CountId** | Guarda información sobre el identificador de un Conteo. Es utilizada para indexar conteos. |
| **DistrictCounts** | Guarda una lista de conteos relacionados con un distrito. A partir de esta clase se generan los registros del índice de conteos por distrito. |
| **DistrictCountsIndex** | Provee una abstracción sobre el árbol B+ para indexar y des-indexar conteos relacionados con un distrito. Adicionalmente, permite recuperar una entrada indexada a partir de un distrito determinado. |
| **DistrictCountsMethods** | Hereda de RecordMethods. Implementa los métodos de esa clase para DistrictCounts. |
| **DistrictElections** | Guarda una lista de elecciones relacionadas con un distrito. A partir de esta clase se generan los registros del índice de elecciones por distrito. |
| **DistrictElectionsIndex** | Provee una abstracción sobre el árbol B+ para indexar y des-indexar elecciones relacionadas con un distrito. Adicionalmente, permite recuperar una entrada indexada a partir de un distrito determinado. |
| **DistrictElectionsMethods** | Hereda de RecordMethods. Implementa los métodos de esa clase para DistrictElections. |
| **ElectionId** | Guarda información sobre el identificador de una Elección. Es utilizada para indexar conteos. |

**Carpeta: VariableBlocks**

En esta carpeta se guardan las clases que implementan funcionalidad relacionada con todos los archivos que usan registros de longitud variable.

|  |  |
| --- | --- |
| **Clase** | **Especificación** |
| **BaseVariableBlock** | Representa de manera abstracta un bloque de un archivo con registros de longitud variable. Guarda el espacio libre disponible, la posición donde comienzan los registros y provee una interfaz común para algunas operaciones como altas, bajas y modificaciones de registros. Adicionalmente, implementa operaciones comunes a todos los bloques, en algunos casos en forma total y en otros como Template Method. |
| **BaseVariableBlockFile** | Provee una interfaz común para todos los archivos que usan bloques de longitud variable, así como también implementaciones base de métodos relacionados con posicionamiento relativo en el archivo. |
| **Constants** | Guarda constantes relacionadas con archivos de longitud variable. Estas son:   * Tamaño del espacio libre del bloque. * Tamaño del espacio reservado para campo de control de la longitud de un registro. * Tamaño del espacio reservado para campo de control de la longitud de un campo. |
| **RecordMethods** | Provee una interfaz común que es usada por las diferentes estructuras que usan registros de longitud variable. Es una alternativa al pasaje de punteros a funciones. Permite, de forma genérica:   * Comparar registros * Obtener la clave a partir de un registro de datos * Imprimir clave * Imprimir registro |
| **SimpleVariableBlock** | Hereda de BaseVariableBlock. Es el tipo de bloque utilizado por SimpleVariableBlockFile. Maneja el espacio libre encadenando listas de espacio libre que se encuentran en diferentes bloques. |
| **SimpleVariableBlockFile** | Hereda de BaseVariableBlockFile. Es una implementación de un archivo secuencial para guardar registros de longitud variable y organizado en bloques. Es usado por el Hash para guardar los bloques de desborde. |
| **VariableRecord** | Abstracción utilizada para que cualquier entidad pueda ser guardada en el árbol B+ y Hash sin necesidad de tener una implementación particular para ella. Permite guardar los registros como bytes y permite acceder a estos bytes a su tamaño. |