

# Base de datos

1er. Cuatrimestre 2006

Proyecto: Minibase  
Informe: Diseño Detallado

12 de octubre de 2006

Número de Grupo:  
Nombre del Grupo: ?

## Integrantes

| Apellido y Nombre  | L.U.   | Mail                       |
|--------------------|--------|----------------------------|
| Leandro Groisman   | 222/03 | gleandro@gmail.com         |
| Fernando Rodriguez | XXX/XX | ferrod20@gmail.com         |
| Guillermo Amaral   | 522/98 | Guillermo.AMARAL@total.com |
| Facioni Francisco  | 004/04 | fran6co@fibertel.com.ar    |

| Instancia | Corrector | Nota |
|-----------|-----------|------|
| Entrega   |           |      |
| Reentrega |           |      |

Comentarios del corrector:

## Índice

|   |          |
|---|----------|
| <b>1. Introducción a Minibase</b>   | <b>1</b> |
| <b>2. BufferManager-Heap</b>  | <b>2</b> |
| 2.1. Descripción general . . . . .  | 2        |
| 2.2. Clases principales y sus protocolos principales . . . . .                                  | 2        |
| 2.3. Interacción con otros componentes . . . . .  | 2        |
| 2.4. Ejemplo de uso . . . . .   | 2        |
| 2.4.1. Diagramas de secuencia . . . . .   | 2        |
| 2.4.2. Script de ejemplo . . . . .  | 2        |
| 2.5. Evaluación del componente (opinión acerca de la calidad de código, diseño, etc.) . . . . . | 2        |
| <b>3. DiskManager</b>   | <b>3</b> |
| 3.1. Descripción general . . . . .  | 3        |
| 3.2. Clases principales y sus protocolos principales . . . . .                                  | 3        |
| 3.3. Interacción con otros componentes . . . . .  | 3        |
| 3.4. Ejemplo de uso . . . . .   | 3        |
| 3.4.1. Diagramas de secuencia . . . . .   | 3        |
| 3.4.2. Script de ejemplo . . . . .  | 3        |
| 3.5. Evaluación del componente (opinión acerca de la calidad de código, diseño, etc.) . . . . . | 3        |
| <b>4. Catalog</b>   | <b>4</b> |
| 4.1. Descripción general . . . . .  | 4        |
| 4.2. DER . . . . .  | 4        |
| 4.3. Clases principales y sus protocolos principales . . . . .                                  | 4        |
| 4.4. Interacción con otros componentes . . . . .  | 4        |
| 4.5. Ejemplo de uso . . . . .   | 4        |
| 4.5.1. Diagramas de secuencia . . . . .   | 4        |
| 4.5.2. Script de ejemplo . . . . .  | 4        |
| 4.6. Evaluación del componente (opinión acerca de la calidad de código, diseño, etc.) . . . . . | 4        |
| <b>5. Iterator</b>  | <b>5</b> |
| 5.1. Descripción general . . . . .  | 5        |
| 5.2. Clases principales y sus protocolos principales . . . . .                                  | 5        |
| 5.2.1. FileScan . . . . .   | 5        |
| 5.2.2. NestedLoopsJoins . . . . .   | 5        |
| 5.2.3. SortMerge . . . . .  | 5        |
| 5.2.4. Sort . . . . .   | 6        |
| 5.3. Interacción con otros componentes . . . . .  | 6        |
| 5.4. Ejemplo de uso . . . . .   | 6        |
| 5.4.1. Diagramas de secuencia . . . . .   | 6        |
| 5.4.2. Script de ejemplo . . . . .  | 6        |

|   |          |
|---|----------|
| 5.5. Evaluación del componente . . . . .  | 6        |
| <b>6. Index-BTree</b>   | <b>7</b> |
| 6.1. Descripción general . . . . .  | 7        |
| 6.2. Clases principales y sus protocolos principales . . . . .                                  | 7        |
| 6.3. Interacción con otros componentes . . . . .  | 7        |
| 6.4. Ejemplo de uso . . . . .   | 7        |
| 6.4.1. Diagramas de secuencia . . . . .   | 7        |
| 6.4.2. Script de ejemplo . . . . .  | 7        |
| 6.5. Evaluación del componente (opinión acerca de la calidad de código, diseño, etc.) . . . . . | 7        |
| <b>7. Optimizer (no existente!)</b>   | <b>8</b> |
| 7.1. Descripción general . . . . .  | 8        |
| 7.2. Clases principales y sus protocolos principales . . . . .                                  | 8        |
| 7.3. Interacción con otros componentes . . . . .  | 8        |
| 7.4. Ejemplo de uso . . . . .   | 8        |
| 7.4.1. Diagramas de secuencia . . . . .   | 8        |
| 7.4.2. Script de ejemplo . . . . .  | 8        |
| 7.5. Evaluación del componente (opinión acerca de la calidad de código, diseño, etc.) . . . . . | 8        |
| <b>8. Tests</b>   | <b>9</b> |
| 8.1. Descripción . . . . .  | 9        |
| 8.2. Resultados obtenidos . . . . .   | 9        |
| 8.3. Ejemplos de uso . . . . .  | 9        |
| <b>9. Herramienta de carga de datos</b>   | <b>9</b> |
| 9.1. Descripción . . . . .  | 9        |
| 9.2. Ejemplos de uso . . . . .  | 9        |
| <b>10. Conclusiones generales</b>   | <b>9</b> |
| <b>11. Apendices</b>  | <b>9</b> |
| <b>12. Código fuente</b>  | <b>9</b> |
| <b>13. Referencias/Bibliografía</b>   | <b>9</b> |

## **1. Introducción a Minibase**

## **2. BufferManager-Heap**

### **2.1. Descripción general**

### **2.2. Clases principales y sus protocolos principales**

### **2.3. Interacción con otros componentes**

### **2.4. Ejemplo de uso**

Si se justifica

#### **2.4.1. Diagramas de secuencia**

#### **2.4.2. Script de ejemplo**

### **2.5. Evaluación del componente (opinión acerca de la calidad de código, diseño, etc.)**

### **3. DiskManager**

#### **3.1. Descripción general**

#### **3.2. Clases principales y sus protocolos principales**

#### **3.3. Interacción con otros componentes**

#### **3.4. Ejemplo de uso**

Si se justifica

##### **3.4.1. Diagramas de secuencia**

##### **3.4.2. Script de ejemplo**

#### **3.5. Evaluación del componente (opinión acerca de la calidad de código, diseño, etc.)**

## **4. Catalog**

### **4.1. Descripción general**

### **4.2. DER**

### **4.3. Clases principales y sus protocolos principales**

### **4.4. Interacción con otros componentes**

### **4.5. Ejemplo de uso**

Si se justifica

#### **4.5.1. Diagramas de secuencia**

#### **4.5.2. Script de ejemplo**

### **4.6. Evaluación del componente (opinión acerca de la calidad de código, diseño, etc.)**

## 5. Iterator

TODO: Revisar los arboles binarios como estan construido y hacer un analisis mas profundo

TODO: Realizar una descripcion mas profunda de cada clase

TODO: Mencionar las clases de excepciones que estan desperdigadas por ahi

TODO: Revisar con mas detenimiento las clases IoBuf y OBuf, q parecen un poco al pedo

TODO: Analizar mas profundamente las clases de eval y projection, y en la parte de uso explicar como se introduce una condicion

### 5.1. Descripción general

La componente iterator es el punto de acceso a las tablas y a sus registros. Como tal ofrece, tambien, las operaciones basicas sobre tablas, como join, proyeccion y seleccion. La iterfaz usada, como dice el nombre, es la de un iterador, el cual se inicializa en base a otros iteradores o heapfiles y despues se va accediendo elemento por elemento en un orden definido por el iterador.

### 5.2. Clases principales y sus protocolos principales

Como dice el nombre de la componente, la interfaz que ofrece es la de un iterador. Esta interfaz es implementada usando una clase abstracta de la cual heredan todas las clases que ofrecen acceso a un conjunto de registros.

#### 5.2.1. FileScan

FileScan permite iterar sobre los registros de un heapfile que cumplen una condicion de seleccion. Esta condicion puede ser nula, permitiendo iterar sobre todos los registros del heapfile. Tambien se puede especificar los atributos de las tuplas de salidas, permitiendo hacer una proyeccion.

La evaluación de la condicion de seleccion y la proyeccion son proveidas por las clases PredEval y Projection, respectivamente.

#### 5.2.2. NestedLoopsJoins

NestedLoopsJoins permite realizar un join entre un iterador y un heapfile. El algoritmo utilizado es el más simple de los implementados, es un doble loop donde, en el cuerpo del loop interno, se verifica la condicion de join. Tambien se puede realizar una proyeccion en la salida.

#### 5.2.3. SortMerge

SortMerge realiza un join utilizando el algoritmo de merge sort. Como NestedLoopsJoins permite realizar una proyeccion en la salida.



Para realizar el sort utiliza varias clases auxiliares, como Sort (que se encarga de iterar de una manera ordenada un heapfile) y IoBuf (que permite almacenar en memoria paginas).

Esta implementacion no elimina los registros duplicados.

#### **5.2.4. Sort**

Sort permite iterar de una manera ordenada la salida de otro iterador. Utiliza un arbol binario ordenado para establecer el orden de la salida.

### **5.3. Interacción con otros componentes**

Esta clase es el punto de acceso principal a los registros. En general con esta clase es con la cual se realiza toda interaccion con la base de datos (salvo la modificacion de las tablas o de los registros).

Respecto a la utilizacion de las otras componentes, Iterator se limita a utilizar heapfile para la interaccion con las tablas.

### **5.4. Ejemplo de uso**

TODO: algun ejemplo de uso q muestre claramente los detalles

#### **5.4.1. Diagramas de secuencia**

No se justifica

#### **5.4.2. Script de ejemplo**

?

### **5.5. Evaluación del componente**

Iterator ofrece una interfaz poco practica para el uso frecuente. Mucha funcionalidad esta repetida, como la proyeccion, que simplemente se podria implementar como otro iterador. Tambien resulta incomoda la construccion de un iterador por la cantidad de estructuras de datos que hay que generar. Igualmente esto se debe a pobre diseño de las clases y su modularización a favor de un estilo que se encuentra, por lo general, en programas escritos en C.

## **6. Index-BTree**

### **6.1. Descripción general**

### **6.2. Clases principales y sus protocolos principales**

### **6.3. Interacción con otros componentes**

### **6.4. Ejemplo de uso**

Si se justifica

#### **6.4.1. Diagramas de secuencia**

#### **6.4.2. Script de ejemplo**

### **6.5. Evaluación del componente (opinión acerca de la calidad de código, diseño, etc.)**

## **7. Optimizer (no existente!)**

### **7.1. Descripción general**

### **7.2. Clases principales y sus protocolos principales**

### **7.3. Interacción con otros componentes**

### **7.4. Ejemplo de uso**

Si se justifica

#### **7.4.1. Diagramas de secuencia**

#### **7.4.2. Script de ejemplo**

### **7.5. Evaluación del componente (opinión acerca de la calidad de código, diseño, etc.)**

## **8. Tests**

### **8.1. Descripción**

### **8.2. Resultados obtenidos**

### **8.3. Ejemplos de uso**

## **9. Herramienta de carga de datos**

### **9.1. Descripción**

### **9.2. Ejemplos de uso**

## **10. Conclusiones generales**

## **11. Apendices**

## **12. Código fuente**

## **13. Referencias/Bibliografía**