

SGBD

Simulación de Disco y SGBD

Creación del disco

Se define un disk.cfg externo y uno interno, si se actualiza el externo se formatea el disco y se pierde toda su info:

```
~ /sgbd main  
> cat disk.cfg  
platos=2  
pistas=4  
sectores=4  
block_size=4096  
blocks_per_sector=4
```

Información del disco

```
> make
> disk_info
==== Información del Disco ====
Nro. de Platos: 2
Nro. de Superficies por plato: 2
Nro. de Pistas por superficie: 4
Nro. de Sectores por pista: 4
Nro. Bloques por sector: 4
Capacidad del sector (KB): 16
Capacidad del sector (MB): 0.015625
Nro. de Bloques por pista: 16
Nro. de Bloques por plato: 128
Capacidad del disco (MB): 1
Capacidad del bloque (bytes): 4096
```

```
disk
├── disk.cfg
├── plato0
│   ├── superficie0
│   │   ├── pista0
│   │   │   ├── sector0
│   │   │   ├── sector1
│   │   │   ├── sector2
│   │   │   └── sector3
│   │   ├── pista1
│   │   │   ├── sector0
│   │   │   ├── sector1
│   │   │   ├── sector2
│   │   │   └── sector3
│   │   └── pista2
│   │       ├── sector0
│   │       ├── sector1
│   │       ├── sector2
│   │       └── sector3
│   └── superficie1
│       ├── pista0
│       │   ├── sector0
│       │   ├── sector1
│       │   ├── sector2
│       │   └── sector3
│       ├── pista1
│       │   ├── sector0
│       │   ├── sector1
│       │   ├── sector2
│       │   └── sector3
│       ├── pista2
│       │   ├── sector0
│       │   ├── sector1
│       │   ├── sector2
│       │   └── sector3
│       └── pista3
│           ├── sector0
│           ├── sector1
│           ├── sector2
│           └── sector3
└── plato1
    ├── superficie0
    │   ├── pista0
    │   │   ├── sector0
    │   │   ├── sector1
    │   │   ├── sector2
    │   │   └── sector3
    │   └── pista1
    │       ├── sector0
    │       ├── sector1
    │       ├── sector2
    │       └── sector3
    └── superficie1
        ├── pista0
        │   ├── sector0
        │   ├── sector1
        │   ├── sector2
        │   └── sector3
        ├── pista1
        │   ├── sector0
        │   ├── sector1
        │   ├── sector2
        │   └── sector3
        ├── pista2
        │   ├── sector0
        │   ├── sector1
        │   ├── sector2
        │   └── sector3
        └── pista3
            ├── sector0
            ├── sector1
            ├── sector2
            └── sector3
```

off 1:nvim- 2:[tmux]*

Info del disco se obtiene con el comando
“disk_info”

Muestra información del disco y llama a
tree para mostrar el árbol

Inicio del sistema

Se carga el catálogo y el bitmap a memoria, se guardan en disco cada vez que sufren cambios

```
SGBD::SGBD(Disk &disk_) : disk(disk_), bitmap(disk_), catalog(disk_) {  
    if (bitmap.load()) {  
    } else {  
        std::cout << "Bitmap no encontrado. Inicializando..." << std::endl;  
        bitmap.set(0, true);  
        bitmap.set(1, true);  
        bitmap.save();  
    }  
  
    catalog.load();  
}
```

Bitmap

Abstracción para obtener índices de data contigua a partir de la estructura del disco:

```
4 #pragma once
3 #include "disk.h"
2 #include <vector>
1
5 class Bitmap {
1   std::vector<bool> bits;
2   int total_blocks;
3   Disk &disk;
4
5 public:
6   Bitmap(Disk &disk_);
7   void set(int index, bool value);
8   bool get(int index) const;
9   bool load();
10  void save() const;
11  int size();
12  int getFreeBlock() const;
13 };
```

Calculo de posición por índice

```
3 BlockPos Disk::blockPosFromIndex(int idx) {
2   int bloques_por_plato =
1     num_superficies * num_pistas * num_sectores * blocks_per_sector;
204
1   int plato = idx / bloques_por_plato;
2   idx = idx % bloques_por_plato;
3
4   int bloques_por_superficie = num_pistas * num_sectores * blocks_per_sector;
5   int superficie = idx / bloques_por_superficie;
6   idx = idx % bloques_por_superficie;
7
8   int bloques_por_pista = num_sectores * blocks_per_sector;
9   int pista = idx / bloques_por_pista;
10  idx = idx % bloques_por_pista;
11
12  int bloques_por_sector = blocks_per_sector;
13  int sector = idx / bloques_por_sector;
14  int bloque = idx % bloques_por_sector;
15
16  return {plato, superficie, pista, sector, bloque};
17 }
```

Garantiza bloques
contiguos en el
sector

Sectores
contiguos en la
pista

Pistas contiguas
en el plato, etc

Siguiendo esta
jerarquía

Adicionar registros

Disco vacío sin registros

bloques 0 y 1 reservados para bitmap y catálogo

```
> status
```

```
Estado del Bitmap (primeros 80 bits):
```

```
11000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
```

```
Catálogo:
```

```
> █
```

Primer bloque asignado al crear la relación

```
> add_from_csv titanic titanic_small.csv fix
Nueva relación añadida...
Nombre: titanic
Tipo: Fijo
Campos:
  - PassengerId (int, 6)
  - Survived    (int, 2)
  - Pclass      (int, 2)
  - Name        (string, 75)
  - Sex         (string, 7)
  - Age         (int, 5)
  - SibSp       (int, 3)
  - Parch       (int, 3)
  - Ticket      (string, 15)
  - Fare        (float, 10)
  - Cabin       (string, 8)
  - Embarked    (string, 2)
Bloques asignados: 2
```

Bloques 2 y 3 luego de insertar registros

```
> schema titanic
Nombre: titanic
Tipo: Fijo
Campos:
  - PassengerId (int, 6)
  - Survived    (int, 2)
  - Pclass      (int, 2)
  - Name        (string, 75)
  - Sex         (string, 7)
  - Age         (int, 5)
  - SibSp       (int, 3)
  - Parch       (int, 3)
  - Ticket      (string, 15)
  - Fare        (float, 10)
  - Cabin       (string, 8)
  - Embarked    (string, 2)
Bloques asignados: 2, 3
```


Ubicación física de los bloques de la relación

```
> rel_block_info titanic
```

Bloques de la relacion titanic:

Bloque 2 ubicado en plato 0, superficie 0, pista 0, sector 0, bloque 2

Bloque 3 ubicado en plato 0, superficie 0, pista 0, sector 0, bloque 3

```
> cat disk/plato0/superficie0/pista0/sector0
titanic fix 12
PassengerId int 6
Survived int 2
Pclass int 2
Name string 75
Sex string 7
Age int 5
SibSp int 3
Parch int 3
Ticket string 15
Fare float 10
Cabin string 8
Embarked string 2
2 3
-0010138002900291      0 3 Braund, Mr. Owen Harris
                        female 38  1 0 PC 17599      71.2833  C85
1 1 Futrelle, Mrs. Jacques Heath (Lily May Peel)
ale 35  0 0 373450      8.05      S 6      0 3 Moran,
thy J      male 54
9      21.075      S 9      1 3 Johnson, Mrs. Oscar W (Elisab
                        female 14  1 0 237736
S 12      1 1 Bonnell, Miss. Elizabeth
male 20  0 0 A/5. 2151      8.05      S 14
rom, Miss. Hulda Amanda Adolfina
0 0 248706      16      S 17      0 3 Rice, Master. F
ne      male 0 0 24
18      S 20      1 3 Masselmani, Mrs. Fatima
male 35  0 0 239865      26
23      1 3 McGowan, Miss. Anna Annie
male 28  0 0 113788      35.5      A6      S 25      0 3 F
. Carl Oscar (Selma Augusta Emilia Johansson)      female
2631      7.225      C 28      0 1 Fortune, Mr. Charles A
female 0 0 330959
6      7.8958      S 31      0 1 Uruchurtu, Don. Manuel E
```


Ya que estan el el primer sector del primer bloque

Al usar “cat” podemos visualizar la data de los bloques 0 y 1 (bitmap y catálogo) junto con la data de la relación, ya que se ubican en el mismo sector físico (en el mismo archivo)

Select all

```
> select all titanic
```

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
1	0	3	Braund, Mr. Owen Harris	male	22	1	0	A/5 21171	7.25		S
2	1	1	Cummings, Mrs. John Bradley (Florence Briggs Thayer)	female	38	1	0	PC 17599	71.2833	C85	C
3	1	3	Heikkinen, Miss. Laina	female	26	0	0	STON/O2. 310128	7.925		S
4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35	1	0	113803	53.1	C123	S
5	0	3	Allen, Mr. William Henry	male	35	0	0	373450	8.05		S
6	0	3	Moran, Mr. James	male	0	0	0	330877	8.4583		Q
7	0	1	McCarthy, Mr. Timothy J	male	54	0	0	17463	51.8625	E46	S
8	0	3	Palsson, Master. Gosta Leonard	male	2	3	1	349909	21.075		S
9	1	3	Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)	female	27	0	2	347742	11.1333		S
10	1	2	Nasser, Mrs. Nicholas (Adele Achem)	female	14	1	0	237736	30.0708		C
11	1	3	Sandstrom, Miss. Marguerite Rut	female	4	1	1	PP 9549	16.7	G6	S
12	1	1	Bonnell, Miss. Elizabeth	female	58	0	0	113783	26.55	C103	S
13	0	3	Saunderscock, Mr. William Henry	male	20	0	0	A/5. 2151	8.05		S
14	0	3	Andersson, Mr. Anders Johan	male	39	1	5	347082	31.275		S
15	0	3	Vestrom, Miss. Hulda Amanda Adolfina	female	14	0	0	350406	7.8542		S
16	1	2	Hewlett, Mrs. (Mary D Kingcome)	female	55	0	0	248706	16		S
17	0	3	Rice, Master. Eugene	male	2	4	1	382652	29.125		Q
18	1	2	Williams, Mr. Charles Eugene	male	0	0	0	244373	13		S
19	0	3	Vander Planke, Mrs. Julius (Emelia Maria Vandemoortele)	female	31	1	0	345763	18		S
20	1	3	Masselmani, Mrs. Fatima	female	0	0	0	2649	7.225		C
21	0	2	Fynney, Mr. Joseph J	male	35	0	0	239865	26		S
22	1	2	Beesley, Mr. Lawrence	male	34	0	0	248698	13	D56	S
23	1	3	McGowan, Miss. Anna Annie	female	15	0	0	330923	8.0292		Q
24	1	1	Sloper, Mr. William Thompson	male	28	0	0	113788	35.5	A6	S
25	0	3	Palsson, Miss. Torborg Danira	female	8	3	1	349909	21.075		S
26	1	3	Asplund, Mrs. Carl Oscar (Selma Augusta Emilia Johansson)	female	38	1	5	347077	31.3875		S
27	0	3	Emir, Mr. Farred Chehab	male	0	0	0	2631	7.225		C
28	0	1	Fortune, Mr. Charles Alexander	male	19	3	2	19950	263	C23 C25	S
29	1	3	O'Dwyer, Miss. Ellen Nellie	female	0	0	0	330959	7.8792		Q
30	0	3	Todoroff, Mr. Lallio	male	0	0	0	349216	7.8958		S
31	0	1	Uruchurtu, Don. Manuel E	male	40	0	0	PC 17601	27.7208		C
32	1	1	Spencer, Mrs. William Augustus (Marie Eugenie)	female	1	0	0	PC 17569	146.5208	B78	C
33	1	3	Glynn, Miss. Mary Agatha	female	0	0	0	335677	7.75		Q
34	0	2	Wheadon, Mr. Edward H	male	66	0	0	C.A. 24579	10.5		S
35	0	1	Meyer, Mr. Edgar Joseph	male	28	1	0	PC 17604	82.1708		
36	0	1	Holverson, Mr. Alexander Oskar	male	42	1	0	113789	52		
37	1	3	Mamee, Mr. Hanna	male	0	0	0	2677	7.2292		
38	0	3	Cann, Mr. Ernest Charles	male	21	0	0	A./5. 2152	8.05		
39	0	3	Vander Planke, Miss. Augusta Maria	female	18	2	0	345764	18		
40	1	3	Nicola-Yarred, Miss. Jamila	female	14	1	0	2651	11.2417		

 Spectacle

Región rectangular

Se ha guardado una captura de pantalla en el portapapeles.

Insertar registro

```
> insert titanic 77 0 2 "Ivan Lopez Zegarra(OFF)" male 20 0 0 "PP 2323" 88393 "C77" S
Bloque_3 ubicado en plato 0, superficie 0, pista 0, sector 0, bloque 3
```

Insertado en el bloque 3 porque aún hay espacio

30	0	3	Todooroff, Mr. Lalio	male	0	0	349216	7.8958		S
31	0	1	Uruchurtu, Don. Manuel E	male	40	0	PC 17601	27.7208		C
32	1	1	Spencer, Mrs. William Augustus (Marie Eugenie)	female	1	0	PC 17569	146.5208	B78	C
33	1	3	Glynn, Miss. Mary Agatha	female	0	0	335677	7.75		Q
34	0	2	Wheadon, Mr. Edward H	male	66	0	C.A. 24579	10.5		S
35	0	1	Meyer, Mr. Edgar Joseph	male	28	1	PC 17604	82.1708		C
36	0	1	Holverson, Mr. Alexander Oskar	male	42	1	113789	52		S
37	1	3	Mamee, Mr. Hanna	male	0	0	2677	7.2292		C
38	0	3	Cann, Mr. Ernest Charles	male	21	0	A./5. 2152	8.05		S
39	0	3	Vander Planke, Miss. Augusta Maria	female	18	2	345764	18		S
40	1	3	Nicola-Yarred, Miss. Jamila	female	14	1	2651	11.2417		C
77	0	2	Ivan Lopez Zegarra(OFF)	male	20	0	PP 2323	88393	C77	S

off 1:nvim- 2:make*

Se intenta insertar en cada bloque, si ningún bloque tiene espacio se pide un nuevo bloque y se guarda en el catálogo, ambos se guardan en disco persistente

```
for (int block_idx : rel.blocks) {  
    if (insertRecord_fix(block_idx, record)) {  
        return true;  
    }  
}  
  
int new_block = bitmap.getFreeBlock();  
if (new_block == -1) {  
    std::cerr << "No hay bloques libres disponibles para insertar" << std::endl;  
    return false;  
}  
bitmap.set(new_block, true);  
initializeBlockHeader_fix(new_block, record_size);  
  
bitmap.save();  
catalog.save();
```

Relación más grande

Ahora cargamos toda la relación titanic para mostrar la contigüidad física de los bloques

```
> delete titanic  
> add_from_csv titanic titanic.csv fix
```

```
> schema titanic  
Nombre: titanic  
Tipo: Fijo  
Campos:  
  - PassengerId (int, 6)  
  - Survived    (int, 2)  
  - Pclass      (int, 2)  
  - Name        (string, 75)  
  - Sex         (string, 7)  
  - Age         (int, 5)  
  - SibSp       (int, 3)  
  - Parch       (int, 3)  
  - Ticket      (string, 15)  
  - Fare        (float, 10)  
  - Cabin       (string, 8)  
  - Embarked    (string, 2)  
Bloques asignados: 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32
```



```
> rel_block_info titanic
```

```
Bloques de la relacion titanic:
```

```
Bloque 2 ubicado en plato 0, superficie 0, pista 0, sector 0, bloque 2
Bloque 3 ubicado en plato 0, superficie 0, pista 0, sector 0, bloque 3
Bloque 4 ubicado en plato 0, superficie 0, pista 0, sector 1, bloque 0
Bloque 5 ubicado en plato 0, superficie 0, pista 0, sector 1, bloque 1
Bloque 6 ubicado en plato 0, superficie 0, pista 0, sector 1, bloque 2
Bloque 7 ubicado en plato 0, superficie 0, pista 0, sector 1, bloque 3
Bloque 8 ubicado en plato 0, superficie 0, pista 0, sector 2, bloque 0
Bloque 9 ubicado en plato 0, superficie 0, pista 0, sector 2, bloque 1
Bloque 10 ubicado en plato 0, superficie 0, pista 0, sector 2, bloque 2
Bloque 11 ubicado en plato 0, superficie 0, pista 0, sector 2, bloque 3
Bloque 12 ubicado en plato 0, superficie 0, pista 0, sector 3, bloque 0
Bloque 13 ubicado en plato 0, superficie 0, pista 0, sector 3, bloque 1
Bloque 14 ubicado en plato 0, superficie 0, pista 0, sector 3, bloque 2
Bloque 15 ubicado en plato 0, superficie 0, pista 0, sector 3, bloque 3
Bloque 16 ubicado en plato 0, superficie 0, pista 1, sector 0, bloque 0
Bloque 17 ubicado en plato 0, superficie 0, pista 1, sector 0, bloque 1
Bloque 18 ubicado en plato 0, superficie 0, pista 1, sector 0, bloque 2
Bloque 19 ubicado en plato 0, superficie 0, pista 1, sector 0, bloque 3
Bloque 20 ubicado en plato 0, superficie 0, pista 1, sector 1, bloque 0
Bloque 21 ubicado en plato 0, superficie 0, pista 1, sector 1, bloque 1
Bloque 22 ubicado en plato 0, superficie 0, pista 1, sector 1, bloque 2
Bloque 23 ubicado en plato 0, superficie 0, pista 1, sector 1, bloque 3
Bloque 24 ubicado en plato 0, superficie 0, pista 1, sector 2, bloque 0
Bloque 25 ubicado en plato 0, superficie 0, pista 1, sector 2, bloque 1
Bloque 26 ubicado en plato 0, superficie 0, pista 1, sector 2, bloque 2
Bloque 27 ubicado en plato 0, superficie 0, pista 1, sector 2, bloque 3
Bloque 28 ubicado en plato 0, superficie 0, pista 1, sector 3, bloque 0
Bloque 29 ubicado en plato 0, superficie 0, pista 1, sector 3, bloque 1
Bloque 30 ubicado en plato 0, superficie 0, pista 1, sector 3, bloque 2
```

Ocupa toda una pista antes
de pasar a la siguiente

disk_cap

Usamos casi todo el espacio disponible de los bloques con datos

Al adicionar registros se llenaran estos bloques, luego el sistema solicita más al disco y actualiza el catálogo con esa info

```
> disk_cap
Cantidad total de bloques: 256
Capacidad total del disco: 1048576 bytes
Capacidad libre: 913408 bytes
Capacidad ocupada: 135168 bytes
Capacidad de bloques con datos (relaciones): 126976 bytes
Capacidad usada en sectores con datos (registros activos): 122958 bytes
■
```

Para cada bloque de la relación se intenta insertar el registro

Retorna false si al leer el header ve que el bloque está lleno

Si no puede insertar en ningún bloque pide un nuevo bloque, lo guarda en el catálogo y luego inserta en el

```
bool SGBD::insertRecord_fix(int block_idx, const std::vector<char> &record) {
    std::vector<char> block = disk.readBlockByIndex(block_idx);

    int free_list_head = std::stoi(std::string(block.begin(), block.begin() + 4));
    int record_size =
        std::stoi(std::string(block.begin() + 4, block.begin() + 8));
    int capacity = std::stoi(std::string(block.begin() + 8, block.begin() + 12));
    int active_records =
        std::stoi(std::string(block.begin() + 12, block.begin() + 16));

    if (record.size() != (size_t)record_size) {
        std::cerr << "Error: tamaño del registro no coincide" << std::endl;
        return false;
    }

    if (active_records >= capacity && free_list_head == -1) {
        return false;
    }
}
```