Tema 5: Introducción y Conceptos Básicos

- Estructuras Lógicas vs. Estructuras Físicas
  - Consecutividad y Concepto de Cubo
- Diseño de Ficheros
  - Diseño Lógico
  - Diseño Físico del Registro Lógico
  - Diseño Físico
- Objetivos del Diseño Físico
  - Cálculo de densidades
  - Costes
- Interacción con ficheros
- Tipología de Procesos

Tema 5.1: Punto de partida

enfoque lógico o externo

enfoque físico o interno

Tema 5.1: Punto de partida

El usuario ve ***archivos***,
que son colecciones de ***registros***,
que son agregaciones de ***datos***

Aplicación
SGBD
SGF

La máquina accede a ***ficheros***,
que son secuencias de ***bloques***,
que son conjuntos de ***bytes***

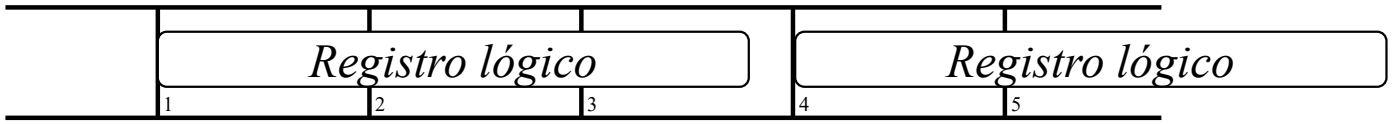
Tema 5.2: Est. Física vs Lógica

Correspondencia físico-lógica a nivel de registro:

consideración de tamaño: Rl y Rf no suelen coincidir en tamaño

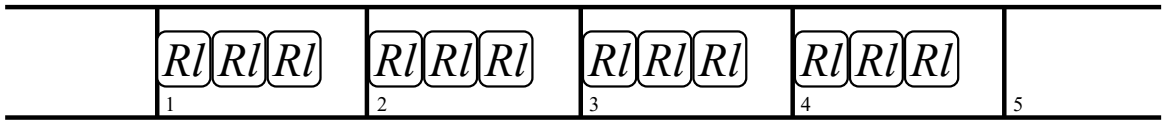
- ***Registro Expandido***

Cuando el registro lógico es mayor que el físico (ocupa varios)



- ***Bloque***

Cuando el registro físico es mayor que el lógico (caben varios)



Factor de Bloqueo: número de registros lógicos que caben en uno físico

$$f_b = \frac{T_{bq}}{T_{reg}}$$

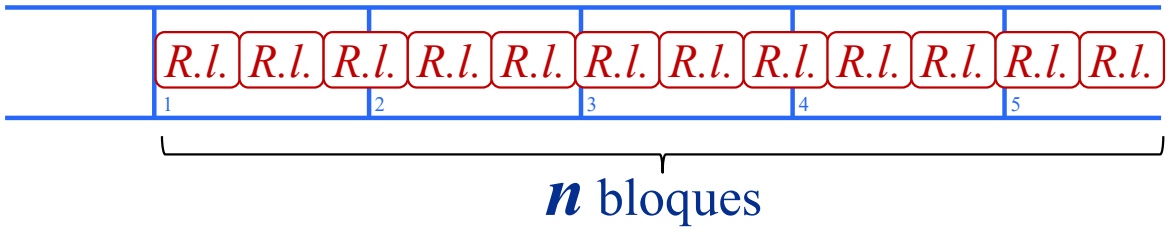
Tema 5.2: Est. Física vs Lógica

Correspondencia físico-lógica entre registros:

A este nivel, la organización de registros puede ser de dos tipos:

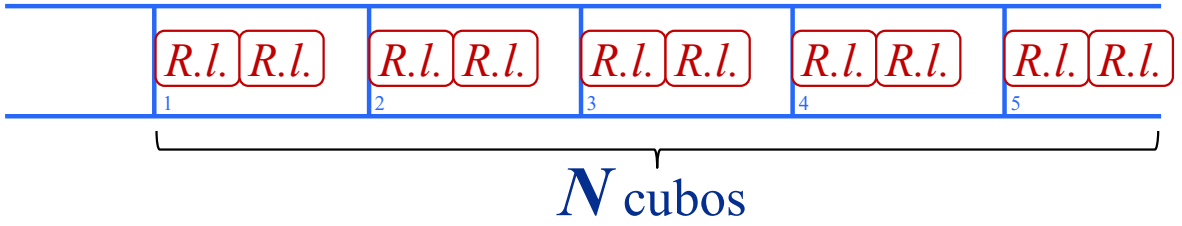
• *Organización* ***Consecutiva***

Todo registro lógico comienza siempre a continuación del anterior



• *Organización* ***No Consecutiva***

Todo registro físico comienza con el inicio de un registro lógico



* *Sólo los ficheros seriales y secuenciales pueden tener regs. consecutivos.*

Tema 5.2: Concepto de Cubo

Cubo (DB block): cjto. de bytes con una condición de acceso común
(comparten dirección, información de control, organización, espacio libre, …)

- suele procesarse como una unidad: es como un *'bloque virtual'*
- dentro del cubo, la organización de los registros es consecutiva
- entre cubos, la organización es no consecutiva

Espacio de cubo (E_c): cantidad de información asignada al cubo
(habitualmente se mide en bloques, pero puede darse en bytes, KB,…)

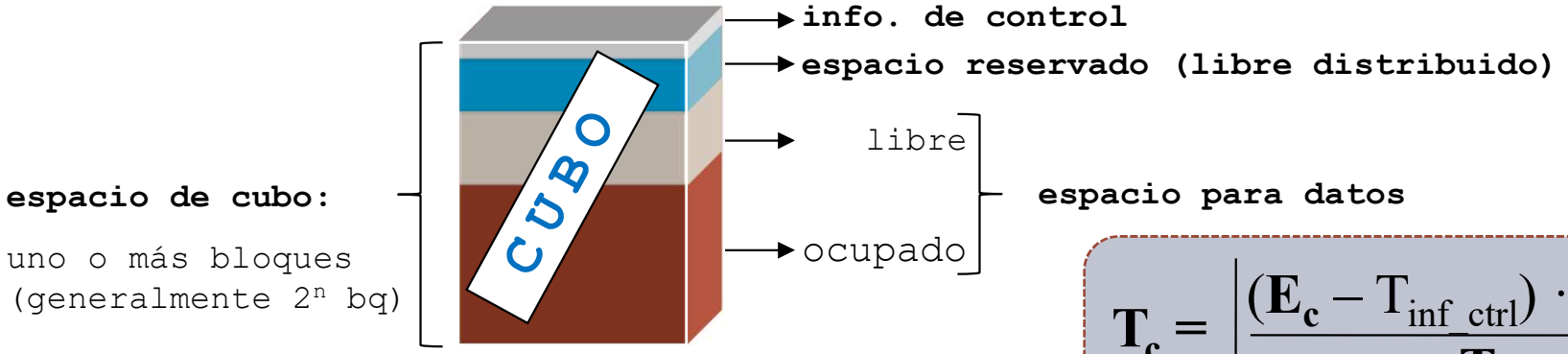
- es un múltiplo del tamaño de bloque
- si el espacio es menor de un bloque, se denomina *'celda'*
- la memoria intermedia suele paginarse al espacio de cubo

Tamaño de cubo (T_c):
cuántos registros lógicos contiene el cubo
(como el factor de bloqueo, pero registros enteros en cubos)

$$T_c = \left\lfloor \frac{E_c}{T_{reg}} \right\rfloor$$

Tema 5.2: Concepto de Cubo
Cubos y Extensiones

- El cubo precisa emplear parte de su espacio (bytes) en información de control:
  - cabecera, directorio de cubo, puntero encadenamiento, byte de desbordamiento, primera posición libre, …
- También puede dejarse reservado un espacio para determinadas operaciones (*espacio libre distribuido*). Por ejemplo, en previsión de que una modificación de registro (update) haga crecer su tamaño (aka, *pctfree)*



$$T_c = \left\lfloor \frac{(E_c - T_{inf\_ctrl}) \cdot (1 - ELD)}{T_{reg}} \right\rfloor$$

- *Otros parámetros:*
  - *Ocupación (tamaño) máxima de cubo: límite preestablecido por encima del que no se insertan registros*
  - *Ocupación mínima de cubo (pctused): umbral por debajo del que se forzarán inserciones en el cubo.*

Tema 5.3: Diseño de Ficheros

Diseño: *concepción original de un objeto u obra destinado a la producción en serie*

Diseño Físico:

"Determinación de la organización física de un fichero"

Organización física: disposición de los registros en el soporte, relativa a su implementación, orden, direccionamiento, apuntamientos, etc.

Diseño Lógico:

"Descripción de la estructura lógica de los registros de un fichero"

Estructura lógica (de un registro): descripción y disposición de los elementos de datos de un registro, que en conjunto definen un individuo.

Elementos de datos: campos, vectores, y grupos repetitivos.

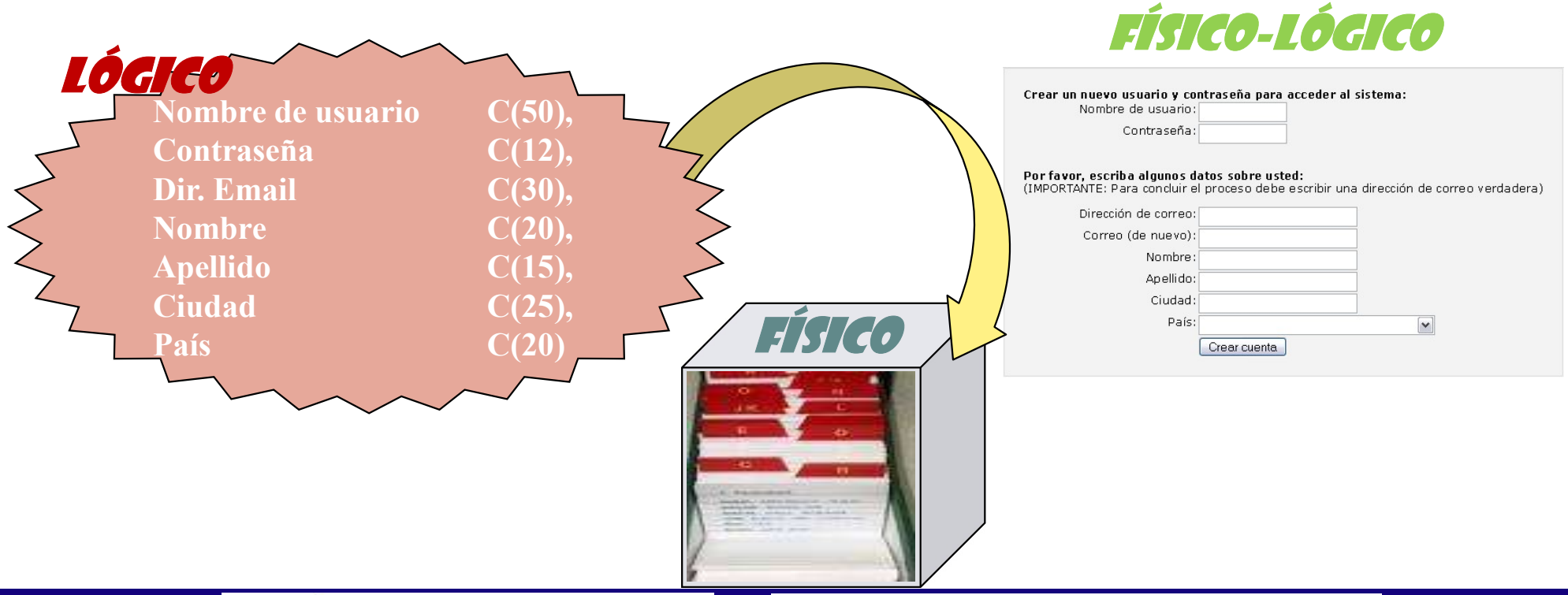
Tema 5.3: Diseño de Ficheros

Diseño Físico del registro Lógico:

"Implementación de un registro lógico en secuencias de bytes"

Diseño Físico-Lógico (registro):

descripción de las cadenas de *bytes* utilizadas para almacenar registros, y de los convenios necesarios para su interpretación.



Tema 5.3.1: Diseño Lógico
Diseño del Registro del Fichero

- **Elemento de datos (campo)**: unidad mínima e indivisible de datos que interviene en un proceso

  Notación: *campo* tipo(tamaño)

  (si es opcional, se encerrará entre corchetes [...] *)*

- **Agregado de datos**: colección de elementos (campos y/o agregados)

  - ***Vector***: agregado compuesto por un número fijo de elementos cuya interpretación es complementaria (entre todos los elementos definen un concepto).

    Notación: ($elemento_1$; $elemento_2$; ...; $elemento_n$)

  - ***Grupo Repetitivo***: agregado compuesto por un número fijo o variable de elementos cuya interpretación es común (el mismo concepto para definir cada elemento)

    Notación: (*elemento*)* para 0..N elementos

    o bien $(elemento)^+$ para 1..N elementos

Tema 5.3.1: Diseño Lógico

Ejemplo: diseño lógico

películasDVD

```
Título          C(42)
Año             N(4)
Productora      C(20)
Género          C(9)
Director        (nombre   C(20),   apellido   C(20))
Actores         (nombre   C(20),   apellido   C(20))+5
[ Autor B.S.    (nombre   C(20),   apellido   C(20))
  Formato       C(3)*3 ]
```

Tema 5.3.2: Diseño Físico-Lógico

- **Es la implementación del registro en secuencias de bytes**
- Objetivo: **eficiencia** (reducir espacio → menor número accesos)
- Debe dar cabida a todos los registros (incluso el de mayor tamaño)

Ejemplo: hallar diseño físico-lógico pésimo y su densidad ideal

- Título (el más largo tiene 42 letras, y de media 30)
- El año de su realización siempre son 4 dígitos
- La productora son 20 car. como máximo, y de media 9 car.
- El género, 9 caracteres como máximo y de media se usan 8.1
- Los nombres de personas usan, en media, 25 caracteres (máximo 40)
- Se han llegado a registrar hasta 5 actores/actrices, y de media 2.8
- El 90% de las películas tienen banda sonora, y de media se tienen 0.5 formatos de cada banda sonora existente ('Formato' siempre gasta 3 car.)

Tema 5.3.2: Diseño Físico-Lógico

- *El diseño físico-lógico pésimo reserva el mayor tamaño para cada campo*
- *Si la ocurrencia es menor, se usará un carácter de relleno (padding)*

películasDVD

```
Título          b(42)
Año             b(4)
Productora      b(20)
Género          b(9)
Director        (nombre   b(20),    apellido   b(20))
Actor/iz 1      (nombre   b(20),    apellido   b(20))
Actor/iz 2      (nombre   b(20),    apellido   b(20))
Actor/iz 3      (nombre   b(20),    apellido   b(20))
Actor/iz 4      (nombre   b(20),    apellido   b(20))
Actor/iz 5      (nombre   b(20),    apellido   b(20))
Autor B.S.      (nombre   b(20),    apellido   b(20))
Formato1        b(3)
Formato2        b(3)
Formato3        b(3)
```

Tema 5.3.2: Diseño Físico-Lógico

Objetivo: Espacio (de un registro)

Volumen y Ocupación de un registro

- **volumen**: número de caracteres necesarios para almacenarlo
- **ocupación útil**: caracteres útiles del registro

volumen ≥ ocupación útil

- **densidad ideal** (d_i) de un registro: relación entre cantidad de información útil y cantidad de información almacenada

$$d_i = \frac{\text{ocupación útil}}{\text{volumen real}} = \frac{\text{tamaño medio de un registro}}{\text{tamaño real de un registro}}$$

Disminuye al aumentar el espacio desperdiciado y/o la información de control

Tema 5.3.2: Diseño Físico-Lógico

Ejemplo de ocurrencia en el fichero de películas *(diseño físico-lógico pésimo)*

películasDVD

```
El Apartamento                                                    1960Metro Goldwyn
Mayer Comedia   Billy                          Wilder                          Jack
                Lemmon                         Shirley                         MacLa
ine             Fred                           MacMurray
                Adolph                         Deutsch                         CD
   El Nombre de la Rosa                                                1986Neue Const
```

- ocupación útil (caso medio registro) = 169.95 B
- volumen real (registro) = 364 B
- d_i = (30+4+9+8.1+25+2.8*25+0.9*(25+0.5*3)) / (42+4+20+9+7*40+3*3) ≈ **46.7%**

¿cómo mejorar la densidad?

$$\text{densidad (\%)} = \frac{\text{ocupación útil}}{\text{volumen real}}$$

¡Reduciendo el volumen!

Tema 5.3.2: Optimización Diseño F-L

Campos de Control (*marcas*): mejoran el manejo en soporte

- Las marcas de elemento de datos ahorran espacio evitando el *padding*.

Elemento de Datos

Existencia: en campos opcionales, indica si aparece o no

Longitud: en campos variables, indica el número de caracteres

Reiteración: en grupos repetitivos, indica el número de ocurrencias

Fin de Campo: en campos indefinidos, indican que acaba

Registro

Fin (inicio) de Registro: separa registros consecutivos en el soporte

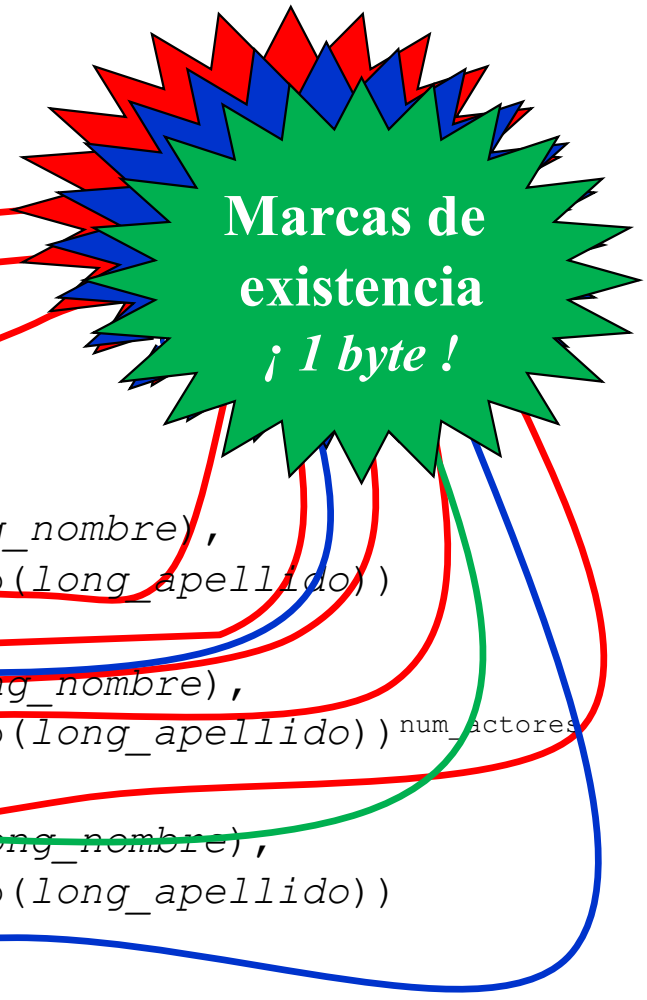
Tipo: indica el tipo de registro a continuación (f. heterogéneos)

Mapa: indica los registros que se aplican (f. heterogéneos)

Tema 5.3.2: Optimización de Espacio

Ejemplo: **películasDVD**

```
longitud_titulo   b(1)
Título            b(longitud_titulo)
Año               b(4)
long_productora   b(1)
Productora        b(long_productora)
Género            b(9)
Director          (long_nombre b(1), nombre b(long_nombre),
                   long_apellido b(1), apellido b(long_apellido))
num_actores       b(1)
Actores           (long_nombre b(1), nombre b(long_nombre),
                   long_apellido b(1), apellido b(long_apellido))num_actores
Existe_BS         b(1)
( Autor BS        (long_nombre b(1), nombre b(long_nombre),
                   long_apellido b(1), apellido b(long_apellido))
  num_formato     b(1)
  Formato         b(3)num_formato )existe_BS
```



- convenio aplicado: todas las cadenas de bytes codificadas en ASCII

$$d_i = 169.95 / (1+30+4+1+9+9+2+25+1+2.8*(2+25)+1+0.9*(2+25+1+0.5*3)) \approx \mathbf{91.8\ \%}$$

Tema 5.3.2: Diseño Físico-Lógico

Ejemplo de ocurrencia en el fichero de películas *(diseño físico-lógico mejorado)*

películasDVD

¿¡esta marca de existencia sobra?!

Oracle DB no distingue entre '' y null

$d_i \approx 92.3\%$

```
♫El Apartamento1960‼Metro Goldwyn Mayer Comedia♣Billy♠Wild
er♥♦Jack♠Lemmon•Shirley◘MacLaine♦Fred○MacMurray☺♠Adolph•Deut
sch☺CD ¶El Nombre de la Rosa1986¶Neue Constantin FilmSuspens
```

- ocupación útil (caso medio registro) = 169.95 B *(no cambia)*
- volumen real (registro) = 185.15 B *(casi la mitad)*
- $d_i \approx 91.8\%$

menor volumen → menos accesos

¿se puede reducir (aún más) el volumen del registro?

Tema 5.3.2: Optimización Diseño F-L

Codificación de campos: consiste en sustituir cierta información por otra equivalente pero de menor tamaño. Sacrifica *legibilidad*.

1. ***Codificación numérica***: no almacenar dígitos (en caracteres alfanuméricos), representar números (en base 256). Ejemplo: `tlf C(9)` → `byte(4)`

2. ***Enumerados***: equivalen a una codificación numérica natural (se puede establecer una codificación de ese tipo)

   Ejemplo: `color ∈ {blanco,amarillo,naranja,rojo,verde,azul,negro}`
   `color C(8)` → `byte(1)`

   - *Pseudoenumerados*: incluyen un valor 'otros', cuya ocurrencia implica la aparición explícita del valor a continuación en el registro.
     Se puede representar como sigue:
     ```
     color          byte(1)
      [color_raro   byte(8)]color='otros'
     ```

3. ***Fechas:*** - *Juliana*: número natural desde fecha base (1/1/-4713 12:00 pm)
   - *texto/mixto*: adaptable en granularidad + cronológico (yyyymmdd)

4. ***Agrupación de varios campos***: por ejemplo, booleanos en un mapa

Tema 5.3.2: Optimización de Espacio

Ejemplo: *(ver 1-35)* **películasDVD**

```
longitud_titulo   b(1)
Título            b(longitud_titulo)
Año               b(2)
long_productora   b(1)
Productora        b(long_productora)
Género            b(1)
Director          (long_nombre b(1), nombre b(long_nombre),
                   long_apellido b(1),  apellido b(long_apellido)
num_actores       b(1)
Actores           (long_nombre b(1),  nombre b(long_nombre),
                   long_apellido b(1),  apellido b(long_apellido)
long_nombre       b(1),
( Autor BS        (nombre b(long_nombre),
                   long_apellido b(1),  apellido b(long_apellido)
  num_formato     b(1)
  Formato         b(1)num_formato ) long_nombre>0
```

Codificación Numérica *¿cuántos bytes?*

Codificación Enumerados *¿cuántos bytes?*

ocupación útil = (30+2+9+1+25+2.8*25+0.9*(25+0.5*1)) = 159.95 B

volumen real = (1+30+2+1+9+1+2+25+1+2.8*(2+25)+1+0.9*(1+25+1+0.5*1)) = **173.25 B**

d_i = 159.95 / 173.25 ≈ **92.3 %**

Tema 5.3.2: Optimización de Espacio

Ejemplo: *(ver 1-35)* **películasDVD**

```
longitud_titulo   b(1)
Título            b(longitud_titulo)
Año               b(2)
long_productora   b(1)
Productora        b(long_productora)
Género            b(1)
Director          (long_nombre b(1), nombre b(long_nombre),
                   long_apellido b(1),  apellido b(long_apellido))
num_actores       b(1)
Actores           (long_nombre b(1),  nombre b(long_nombre),
                   long_apellido b(1),  apellido b(long_apellido))num_actores
long_nombre       b(1),
( Autor BS        (nombre b(long_nombre),
                   long_apellido b(1),  apellido b(long_apellido))
  Formato         b(1) )long_nombre>0
```

Codificación Mapa *¿cuántos bits?*

ocupación útil = (30+2+9+1+25+2.8*25+0.9*(25+1)) = 159.55 B

volumen real = (1+30+2+1+9+1+2+25+1+2.8*(2+25)+1+0.9*(1+25+1)) = **172.8 B**

d_i = 159.95 / 172.8 ≈ **92.56 %**

Tema 5.3.2: Diseño Físico-Lógico

Ejemplo de ocurrencia en el fichero de películas *(diseño físico-lógico optimizado)*

películasDVD

```
♫El Apartamento•¿‼Metro Goldwyn Mayer☺♣Billy♠Wilder♥♦Jack♠Lem
mon•Shirley◘MacLaine♦Fred○MacMurray♠Adolph•Deutsch@¶El Nombre
```

- ocupación útil (caso medio registro) = 159.95 B *(menor)*
- volumen real (registro) = **172.8 B** *(reduce más de 6%)*
- $d_i \approx 92.56\%$

menor volumen → menos accesos → ***mayor eficiencia***

(para toda org. base)

Tema 5.3.3: Diseño Físico

A partir de los *registros*, implementados en secuencias de bytes…

¿cómo **organizar** esas secuencias de bytes para …?

A) que ocupen menos espacio de almacenamiento

B) que el coste de acceso sea mínimo

Diseño Físico

Objetivos: Espacio (de un fichero)

- **densidad real (d_r)** de un fichero: relación entre cantidad de información útil y cantidad de información almacenada

$$d_r = \frac{\text{ocupación fichero}}{\text{volumen fichero}} = \frac{\text{ocupación/reg} * \text{nºregistros}}{n \text{ bloques} * T_{bq}}$$

Siempre está referida a un dispositivo (soporte) y a una organización (O_n)

Siempre se cumple que $d_i \geq d_r$

- **densidad de ocupación (d_o) de un fichero no consecutivo**: relación entre cantidad de registros almacenados y cantidad de ellos que caben, mide el porcentaje de espacio potencialmente útil no utilizado (referido siempre a una O_n)

$$d_{oc} = \frac{\text{nºregistros}}{N \text{ cubos} * T_c}$$

Tema 5.4: Diseño Físico

Objetivos: Coste

- ¿Cuál de esas organizaciones aprovecha mejor el espacio?
- ¿En cuál cuesta menos sacar un listado completo? (procesar todo)
- ¿Y si el listado debe seguir un criterio de ordenación?
- ¿Y en cuál cuesta menos encontrar un libro concreto?
- ¿Y en cuál cuesta menos insertar? ¿o borrar? ¿o modificar?
- ¿Y si tengo varios procesos distintos? …

Tema 5.4: Objetivos del Diseño Físico

Coste Global (*accesos lógicos*)

Frecuencia Relativa de un Proceso:

Todo sistema de archivos estará sometido a un cjto. de procesos $P \equiv \{P_1..P_n\}$

Cada proceso tendrá una frecuencia f_i asociada, referida a...

- a una unidad de tiempo (por ejemplo, segundos u horas)
- frecuencia relativa: al conjunto P de todos los procesos, tal que $\sum_{i=1..n} f_i = 1$

Coste Global (referido a la organización O_k y al conjunto de procesos P):

- Una organización física del Sistema de Archivos (O_k) define todas las organizaciones base de los archivos que incluye, vínculos entre los mismos, y las organizaciones auxiliares con las que cuenta.
- Cada proceso P_i tendrá asociado en O_k un **coste parcial** C_i (expresado en nº accesos o en tiempo), que dependerá directamente de la organización.
- El **coste global** de la organización se define como la media de accesos lógicos por carga o ud. tiempo (*media ponderada de los costes parciales*).

$$C(O_k,P) = \sum_{i=1..n} C_i \cdot f_i$$

Tema 5.4: Mejoras a bajo nivel

Elección del soporte

- Determinados soportes mejoran su comportamiento en acceso **serial**
  - **Lectura aleatoria:** se localiza un bloque para operarlo.
    El tiempo de localización depende del tiempo que el disco tarda en dar una vuelta (latencia) y que puede estar entorno a los 5ms.
  - **Secuencialidad de disco**: acceder al bloque siguiente (consecutivo / entrelazado) es muchísimo más rápido (hasta 500 veces)
- Los SSD carecen de esta ventaja, y son menos duraderos (<10000 ciclos). Pero suelen ser más rápidos (depende de la tecnología y la operación).

Tema 5.4: Mejoras a bajo nivel

Uso de Memorias Intermedias (buffers)

- Se pueden almacenar bloques en memoria privilegiada (en *páginas*)

- Acceso paginado:
al pedir un bloque, devuelve la dir. página buffer donde está;
si no está en buffer, vacía una página, lo lee y lo almacena allí.
- Otras mejoras: adelantar lecturas / demorar escrituras
- ventaja: velocidad (ahorra accesos al dispositivo)
- inconveniente: coste elevado → recurso muy escaso

- *hit ratio* (*hr* | φ_{buf}): porcentaje de accesos ahorrados por la M_{int}

PIO coste real (efectivo) = (1 - *hr*) * coste global **LIO**

Tema 5.5: Interacción con FF

***Clave*: campo (o conjunto de campos) con una función específica en la interacción de los usuarios (procesos) con el fichero.**

Tipos de clave:

- Clave de **Identificación** (identificador, clave identificativa o unívoca):

  identifica unívocamente cada ocurrencia en el archivo
  (toma valores diferentes para cada registro)

- Clave **No Identificativa**:

  Toma valores en un dominio. Los valores se repiten en distintos registros.

  - cardinalidad del dominio: #valores(CO)
  - Coincidencia *k*: número de registros (en media) que adoptan ese valor
    $k = r$ / #valores(CO) , ,donde r es el número de registros del fichero

Tema 5.5: Interacción con FF

Clave de **Búsqueda**:

Información utilizada (± frecuentemente) para seleccionar registros.

También entran en esta categoría los atributos proyectados (objetivo).

Clave **Privilegiada**:

La que cuenta con un mecanismo de localización (física) eficiente.

Las claves de búsqueda no privilegiadas se denominan *claves alternativas*

Existen varios sub-tipos:

- Clave de **Direccionamiento**: determina la ubicación del registro
- Clave de **Ordenación**: criterio de ordenación, ya sea físico (org) o lógico (proceso)
- Clave de **Indización (indexación)**: privilegiada a través de una estructura auxiliar
- Clave de Agrupamiento (**clusterización**): reúne registros con esa clave común.

Tema 5.5: Tipología de Procesos

- **Actualización** vs. **Recuperación**

- Tipos de Actualización:
  - Insert: incorporar datos (*selectivo vs. bulk*)
  - Delete: eliminar datos (*selectivo vs. vaciado*)
  - Update: modificar datos (*selectivo vs. a la totalidad*)

- Características de Procesos de Recuperación:
  - Proc. Selectivo vs Incondicional (acceso a la totalidad o *full scan*)
  - Proc. sobre un archivo o varios (combinación de archivos)
  - Consulta de datos agregados (analítica) vs individuales
  - Resultado Ordenado (o sin ordenar)

Tema 5.5: Procesos y Subprocesos

- **Actualización (selectiva) - Subprocesos**

conlleva coste (acc)

tipo	ámbito	subproceso	Insert	Delete	Update
comprobar restricciones	registro	validación			
	archivo	unicidad (PK/UK)	✓		✓
	esquema	referencia (FK) (cl. referenciante)	✓		✓
integridad	esquema	referencia (FK) (cl. referenciada)		~	~
localización	archivo		~	✓	✓
escritura	archivo		✓	✓	✓

Tema 5.5: Procesos de Recuperación

- Recuperación Selectiva vs Incondicional

- Selección **Identificativa** (*exact match*) vs **No Identificativa**
- Subtipos de Selección **simple** (un atributo):
  - Selección de un registro por clave identificativa (exact match)
  - Selección de *k* registros (coincidentes) por clave no identificativa
  - Selección de *k* registros en un rango (sobre cualquier clave de búsqueda)
- Subtipos de Selección **multiclave**: (varias claves **no identificativas**)
  - Selección de *k* registros en un rango (window query)
  - Proyección de pocos atributos del resultado de una WQ

Tema 5.5: Subprocesos Recuperación

- Acceso a la totalidad (full scan) vs *filtrado de cubos*
- Combinación de varios orígenes de datos (*join*)
  - recorridos anidados
  - orígenes ordenados
  - composición dispersa
- Filtrado por combinación (*consultas anidadas*)
- Agrupación / Agregación
- Ordenación (mezcla natural)
- Otros (independientes de org.): combinación resultados, manejo cursores, etc.

*¿ Proceso en **Mem. Principal*** vs. *Virtualización en **Disco** !!*