

FUNDAMENTOS DE ORGANIZACIÓN DE DATOS

Clase 4

Agenda

Búsqueda de información

- Secuencial
- Directa

Busqueda binaria



- Costo de orden

Clasificación

- alternativas

Archivos - Búsqueda

Búsqueda de información (costo)

- # de comparaciones (operaciones en memoria)
 - Se pueden mejorar con algoritmos más eficientes.
- # de accesos (operaciones en disco)

Buscar un registro

- + rápido si conocemos el NRR (directo)
- Secuencia debe buscarse desde el principio
- Trataremos de incorporar el uso de claves o llaves.

Archivos - Búsqueda

Búsqueda binaria □ precondiciones

- Archivo ordenado por clave
- Registros de longitud fija




Búsqueda □ partir el archivo a la mitad y comparar la clave,

- puedo acceder al medio por tener long. Fija
- Si N es el # de registros, la performance será del orden de $\log_2 N$
- Se mejora la performance de la búsqueda secuencial.

Archivos □ Clasificación

Búsqueda binaria

- acota el espacio para encontrar información
- costo □ mantener orden  do el archivo

Como clasificar (ordenar) un archivo

- En RAM
- Claves en RAM
- Archivos Grandes?

Archivos □ Clasificación

Llevar el archivo a RAM

- Eficiencia?

Llevar las claves a RAM 

- Eficiencia?

Si no caben en RAM las claves

- Ordenar sobre disco?
 - Eficiencia?
- Alternativa

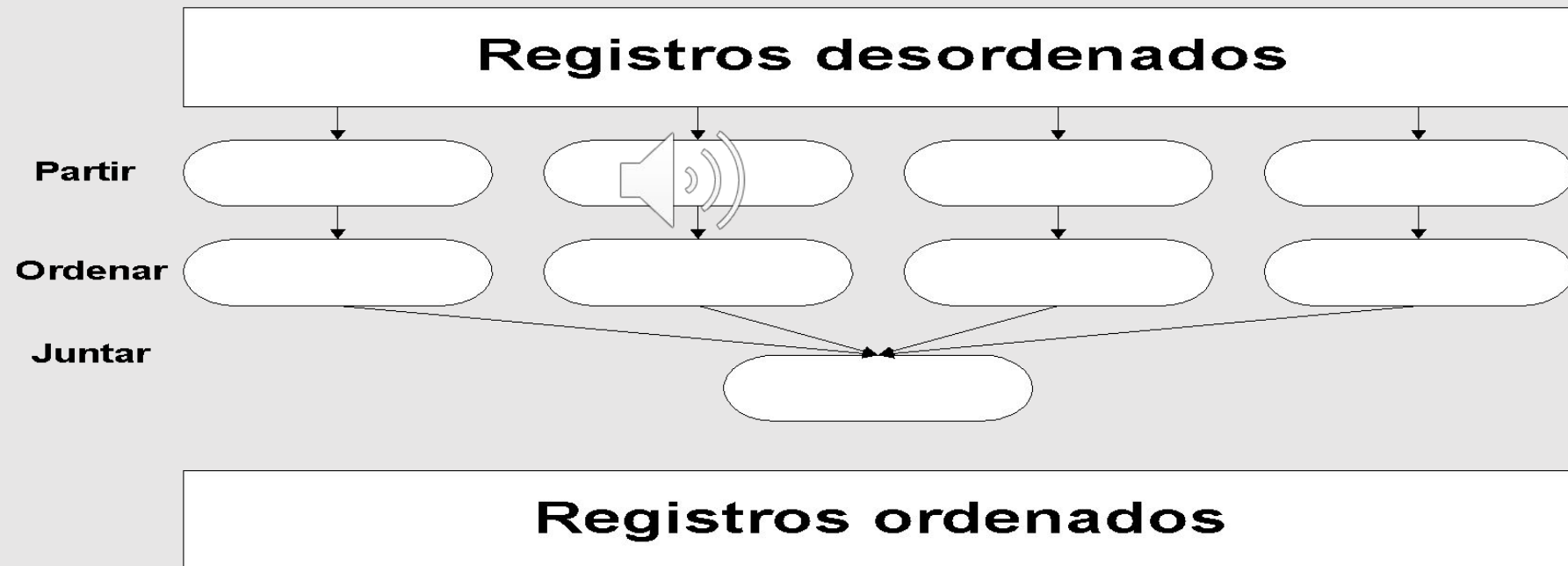
Archivos □ clasificación

Archivos demasiado grandes
para caber en memoria Ram



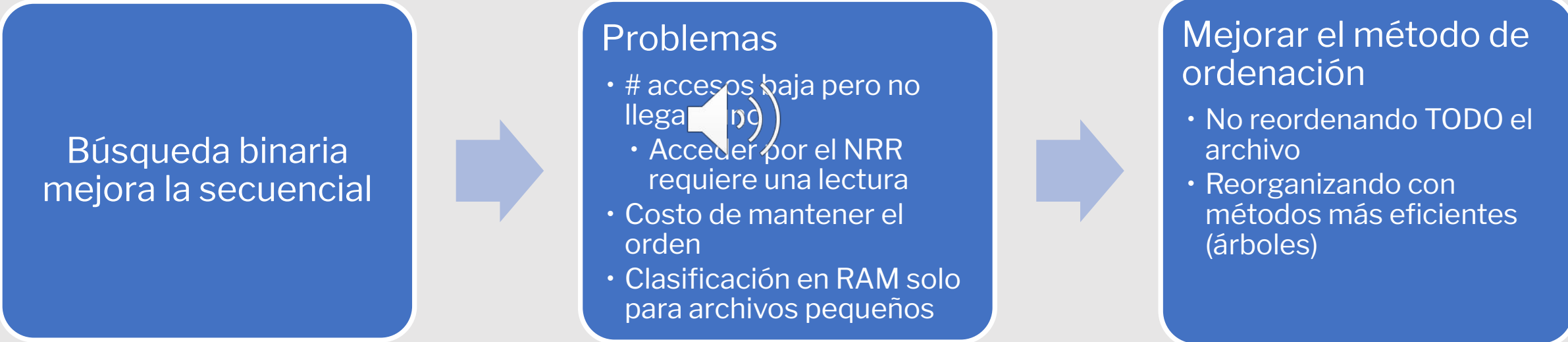
- Partir el archivo
- Ordenar cada parte
- Juntar las partes ordenadas (merge)

Archivos – Clasificación



Archivos □ Algunas conclusiones

Búsqueda binaria
mejora la secuencial



```
graph LR; A[Búsqueda binaria mejora la secuencial] --> B[Problemas]; B --> C[Mejorar el método de ordenación];
```

Problemas

- # accesos baja pero no llega a 100
- Acceder por el NRR requiere una lectura
- Costo de mantener el orden
- Clasificación en RAM solo para archivos pequeños

Mejorar el método de ordenación

- No reordenando TODO el archivo
- Reorganizando con métodos más eficientes (árboles)