

الجمهورية الجزائرية الديمقراطية الشعبية

République Algérienne Démocratique et Populaire

وزارة التعليم العالي و البحث العلمي

*Ministère de l'Enseignement Supérieur et de la Recherche
Scientifique*

*Ecole Supérieure d'Informatique
De Sidi Bel Abbès*



مدرسة العليا للإعلام الإلكتروني
بسيدي بلعباس

ORGANISATION DES FICHIERS

Présenté par Dr.Nabil KESKES.

Année 2016-2017.

PLAN

- Introduction.
- Rappel.
- Chargement initial
- Recherche dichotomique
- Suppression logique
- Insertion
- Conclusion

Ecole Supérieure d'Informatique
De Sidi Bel Abbés



مدرسة العليا للإعلام الآلي
بمسيدى بلعباس

Objectifs du Chapitre

L'objectif de notre cours consiste donc à répondre aux deux questions suivantes :
Comment définir :

- ✓ l'organisation du fichier : séparation des champs, des articles, ..
- ✓ la méthode d'accès : comment localiser un article sur le fichier ?

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي و البحث العلمي
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Ecole Supérieure d'Informatique
De Sidi Bel Abbès



مدرسة العليا للإعلام الآلي
بمسيدى بلعباس

Ca revient donc à étudier les opérations de base :

- recherche
- insertion
- suppression
- requête à intervalle

sous différentes formes d'accès qui sont :

- ✓ fichier vu comme un tableau
- ✓ fichier vu comme une liste linéaire chaînée

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي و البحث العلمي
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Ecole Supérieure d'Informatique
De Sidi Bel Abbès



مدرسة العليا للإعلام الآلي
بمبدي بلعباس

Rappel

✓ Un fichier physique est un fichier logique sur un support.

✓ Organisation

Combinaison d'une **implantation physique** et d'une **méthode d'accès**.

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي و البحث العلمي
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Ecole Supérieure d'Informatique
De Sidi Bel Abbès



مدرسة العليا للإعلام الآلي
بسيدي بلعباس

✓ **Implantation**

Manière d'organiser les informations sur le support.

✓ **Organisation contiguë (Consécutive)**: les enregistrements sont placés consécutivement sur le support il s'agit d'une obligation pour les supports non adressables.

✓ **Organisation chaînée (Dispersé)** : ne peut s'effectuer que sur des supports **adressables**. A chaque enregistrement, on ajoute un pointeur sur l'adresse de l'enregistrement suivant et précédent. Les deux méthodes d'accès (séquentiel ou direct) sont possibles.

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي و البحث العلمي
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Ecole Supérieure d'Informatique
De Sidi Bel Abbés



مدرسة العليا للإعلام الآلي
بمبدي بلعباس

✓ **Méthode d'accès**

Moyen de retrouver un enregistrement sur le support.

✓ **Séquentiel**

Pour accéder à l'enregistrement m, il faut tout d'abord accéder aux enregistrements précédents. Cette méthode d'accès est utilisable quelque soit le type de mémoire de stockage (adressable ou non).

✓ **Direct**

Accès direct à l'enregistrement grâce à son adresse – ce qui implique une obligation d'utiliser des mémoires adressables.

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي و البحث العلمي
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Ecole Supérieure d'Informatique
De Sidi Bel Abbès



مدرسة العليا للإعلام الآلي
بمبدي بلعباس

Typologie des organisations

L'organisation est une composition de l'implantation et de la méthode d'accès.

- ✓ **Séquentielle**: implémentation consécutive – accès séquentiel
- ✓ **Séquentielle indexée** : combinaison d'implémentation consécutive et accès séquentiel ou sélectif
Et directe
- ✓ **Directe** : implémentation directe – accès direct

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي و البحث العلمي
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Ecole Supérieur d'Informatique
De Sidi Bel Abbés



مدرسة العليا للإعلام الآلي
بمبيدي بلعباس

Organisation séquentielle indexée

Chaque fois qu'un article est écrit ,une clé lui est associée et une adresse de l'article dans le fichier sont inscrites dans une table appelée table d'indexe

Lors d'un accès au fichier, on sélectionne l'enregistrement dans la table d'index puis on effectue un accès direct

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي و البحث العلمي
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Ecole Supérieur d'Informatique
De Sidi Bel Abbés



مدرسة العليا للإعلام الآلي
بمبيدي بلعباس

Organisation directe

L'indicatif de chaque article est traduit en adresse physique en appliquant une fonction de randomisation – algorithme de calcul appliqué lors de la création du fichier ; il y a un risque de collision, c'est à dire que deux indicatifs différents "donnent" le même enregistrement.

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي و البحث العلمي
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Ecole Supérieure d'Informatique
De Sidi Bel Abbés



مدرسة العليا للإعلام الآلي
بمسيسي بلعباس

Exemple 1

fichier de type « TOF » (fichier vu comme tableau, ordonné avec enregistrements à taille fixe)

- ✓ L'opération du chargement initial consiste à construire un fichier ordonné avec n enregistrements initiaux, en laissant un peu de vide dans chaque bloc. Ce qui permettra de minimiser les décalages pouvant être provoqués par les futures insertions.
- ✓ La recherche d'un enregistrement est dichotomique (rapide).

Déclaration du fichier:

const

b = 30; // capacité maximale des blocs (en nombre d'enregistrements)

type

Tenreg = structure // Structure d'un enregistrement :

effacé : boolean; // booléen pour la suppression logique

cle : typeqlq; // le champs utilisé comme clé de recherche

champ2 : typeqlq; // les autres champs de l'enregistrement,

champ3 : typeqlq; // sans importance ici.

...

Fin;

Tbloc = structure // Structure d'un bloc :

tab : tableau[1..b] de Tenreg; // un tableau d'enreg d'une capacité b

NB : entier; // nombre d'enreg dans tab (<= b)

Fin;

var // variables globales (F et buf)

F : Fichier de Tbloc Buffer buf Entete (entier, entier);

/* Description de l'entête du fichier F :

L'entête contient deux caractéristiques de type entier.

- la première sert à garder la trace du nombre de bloc utilisés (ou alors le numéro logique du dernier bloc du fichier)
- la deuxième servira comme un compteur d'insertions pour pouvoir calculer rapidement le facteur de chargement, et donc voir s'il y a nécessité de réorganiser le fichier.

*/



Chargement initial

Le chargement initial d'un fichier ordonné consiste à construire un nouveau fichier contenant dès le départ n enregistrements. Ceci afin de laisser un peu de vide dans chaque bloc, qui pourrait être utilisé plus tard par les nouvelles insertions tout en évitant les décalages inter-blocs (très coûteux en accès disque)

Chargement_Initial(nomfich : chaine; n : entier; u : reel)

// u est un réel compris entre 0 et 1 et désigne le taux de chargement voulu au départ

var

e : Tenreg;

i,j,k : entier;

DEBUT

Ouvrir(F, nomfich, 'N'); *// un nouveau fichier*

i := 1; *// num de bloc à remplir*

j := 1; *// num d'enreg dans le bloc*

ecrire('Donner les enregistrements en ordre croissant suivant la clé : ');

POUR $k:=1, n$

lire(e);

SI $j \leq u*b$ *// ex: si $u=0.5$, on remplira les bloc jusqu'à $b/2$ enreg*

buf.tab[j] := e

$j := j+1$;

SINON *// $j > u*b$: buf doit être écrit sur disque*

buf.NB = $j-1$;

EcrireDir(F, i, buf);

buf.tab[1] := e ; *// le kème enreg sera placé dans le prochain bloc, à la position 1*

$i := i+1$;

$j := 2$;

FSI

FP

// à la fin de la boucle, il reste des enreg dans buf qui n'ont pas été sauvegardés sur disque

buf.NB := j-1;
EcrireDir(F, i, buf);

// mettre à jour l'entête (le num du dernier bloc et le compteur d'insertions)

Aff-entete(F, 1, i);

Aff-entete(F, 2, n);

Fermer(F)

FIN // chargement-initial

La recherche d'un enregistrement est dichotomique (rapide).

Module de recherche: (dichotomique)

en entrée la clé à chercher et le nom externe du fichier.

en sortie le booleen Trouv, le num de bloc (i) contenant (c) et le déplacement (j)

Rech(c:typeqlq; nomfich:chaine; var Trouv:bool; var i,j:entier)

var

bi, bs, inf, sup : entier;

trouv, stop : booleen;

DEBUT

Ouvrir(F, nomfich, 'A');

bs := entete(F,1); *// la borne sup (le num du dernier bloc de F)*

bi := 1; *// la borne inf (le num du premier bloc de F)*

Trouv := faux; stop := faux; j := 1;

```

TQ ( bi <= bs et Non Trouv et Non stop )
i := (bi + bs) div 2; // le bloc du milieu entre bi et bs
LireDir( F, i, buf );
SI ( c >= buf.tab[1].cle et c <= buf.tab[buf.NB].cle )
// recherche dichotomique à l'intérieur du bloc (dans la variable buf)...
inf := 1; sup := buf.NB;
TQ inf <= sup et Non Trouv
    j := (inf + sup) div 2;
    SI c = buf.tab[j].cle: Trouv := vrai
    SINON
        SI c < buf.tab[j].cle: sup := j-1
        SINON inf := j+1
    FSI
FSI

FTQ
SI ( Non Trouv )
    j := inf
FSI
// fin de la recherche interne. j indique l'endroit où devrait se trouver c
stop := vrai

```

```

SINON // non (  $c \geq \text{buf.tab}[1].cle$  et  $c \leq \text{buf.tab}[\text{buf.NB}].cle$  )
SI (  $c < \text{buf.tab}[1].cle$  )
    bs := i-1
SINON //  $c > \text{buf.tab}[\text{buf.NB}].cle$ 
    bi := i+1

FSI
FSI
FTQ
SI ( Non Trouv )
    i := bi

FSI
fermer( F )
FIN

```

La suppression logique consiste à rechercher l'enregistrement et positionner le champs

'effacé' à vrai :

Suppression(c:typeqlq; nomfich:chaine)

var

trouv : booleen;

i,j : entier;

DEBUT

// on commence par rechercher la clé c pour localiser l'emplacement (i,j) de l'enreg à supprimer

Rech(c, nomfich, trouv, i, j);

// ensuite on supprime logiquement l'enregistrement

SI (trouv)

Ouvrir(F,nomfich, 'A');

LireDir(F, i, buf);

buf.tab[j].effacé := VRAI;

EcrireDir(F, i, buf);

Fermer(F)

FSI

FIN *// suppression*

```

Insérer( e:Tenreg; nomfich:chaîne )
var
trouv : boolean;
i,j,k : entier;
e,x : Tenreg;
DEBUT
// on commence par rechercher la clé e.cle avec le module précédent pour localiser l'emplacement
(i,j)
// où doit être insérer e dans le fichier.
Rech( e.cle, nomfich, trouv, i, j );
SI ( Non trouv ) // e doit être inséré dans le bloc i à la position j
Ouvrir( F,nomfich, 'A'); // en décalant les enreg j, j+1, j+2, ... vers le bas
continu ← vrai;
// si i est plein, le dernier enreg de i doit être inséré dans i+1
TQ ( continu et i<= entete(F,1) ) // si le bloc i+1 est aussi plein son dernier enreg sera
LireDir( F, i, buf ); // inséré dans le bloc i+2, etc ... donc une boucle TQ.
// avant de faire les décalages, sauvegarder le dernier enreg dans une var x ...
x ← buf.tab[buf.NB];
// décalage à l'intérieur de buf ...
k ← buf.NB;
TQ k > j
buf.tab[k] ← buf.tab[k-1];
k ← k-1
FTQ
// insérer e à la pos j dans buf ...
buf.tab[j] ← e;

```

```

// si buf n'est pas plein, on remet x à la pos NB+1 et on s'arrête ...
SI ( buf.NB < b ) // b est la capacité max des blocs (une constante)
buf.NB ← buf.NB+1;
buf.tab[buf.NB] ← x;
EcrireDir( F, i, buf );
continu ← faux;
SINON // si buf est plein, x doit être inséré dans le bloc i+1 à la pos 1
...
EcrireDir( F, i, buf );
i ← i+1;
j ← 1;
e ← x; // cela se fera (l'insertion) à la prochaine itération du TQ
FSI // non ( buf.NB < b )
FTQ

```



```
// si on dépasse la fin de fichier, on rajoute un nouveau bloc
contenant un seul enregistrement e
SI i > entete( F, 1 )
  buf.tab[1] ← e;
  buf.NB ← 1;
  EcrireDir( F, i, buf ); // il suffit d'écrire un nouveau bloc à cet
  emplacement
  Aff-entete( F, 1, i ); // on sauvegarde le num du dernier bloc dans
  l'entete 1
FSI
Aff-entete( F, 2 , entete(F,2)+1 ); // on incrémente le compteur
d'insertions
Fermer( F );
FSI
FIN
```

الجمهورية الجزائرية الديمقراطية الشعبية

République Algérienne Démocratique et Populaire

وزارة التعليم العالي و البحث العلمي

*Ministère de l'Enseignement Supérieur et de la Recherche
Scientifique*

Ecole Supérieure d'Informatique

De Sidi Bel Abbés



مدرسة العليا للإعلام الآلي
بسيدي بلعباس

4. Conclusion

✓ Le fichier peut être vu comme un tableau, c'est à dire constitué de M blocs contigus. Le fichier peut être ordonné ou pas. S'il n'est pas ordonné, les ajouts d'articles se font à la fin du fichier. L'accès est alors séquentiel. S'il est ordonné, les ajouts d'articles causent des décalages. L'accès est alors soit dichotomique soit séquentiel.

الجمهورية الجزائرية الديمقراطية الشعبية

République Algérienne Démocratique et Populaire

وزارة التعليم العالي و البحث العلمي

*Ministère de l'Enseignement Supérieur et de la Recherche
Scientifique*

Ecole Supérieure d'Informatique

De Sidi Bel Abbés



مدرسة العليا للإعلام الآلي
بسبيدي بلعباس

4. Conclusion

✓ Le fichier peut être vu comme une liste linéaire chaînée de blocs, c'est à dire constitué de M blocs non contigus. Le fichier peut être ordonné ou pas. S'il n'est pas ordonné, les ajouts d'articles se font à la fin du fichier. S'il est ordonné, les ajouts d'articles causent des insertions de nouveaux blocs (allocation dynamique). Dans les deux cas, l'accès est exclusivement séquentiel.

الجمهورية الجزائرية الديمقراطية الشعبية

République Algérienne Démocratique et Populaire

وزارة التعليم العالي و البحث العلمي

*Ministère de l'Enseignement Supérieur et de la Recherche
Scientifique*

Ecole Supérieure d'Informatique

De Sidi Bel Abbès



مدرسة العليا للإعلام الآلي
بسيدي بلعباس

4. Conclusion

L'organisation du fichier est quelconque : séparation des champs, des articles, format fixe ou variable. Si le fichier est ordonné, on fait d'abord un chargement initial en remplissant les blocs à 60-70%. On économise ainsi les décalages intra-blocs.

الجمهورية الجزائرية الديمقراطية الشعبية

République Algérienne Démocratique et Populaire

وزارة التعليم العالي و البحث العلمي

*Ministère de l'Enseignement Supérieur et de la Recherche
Scientifique*

Ecole Supérieure d'Informatique

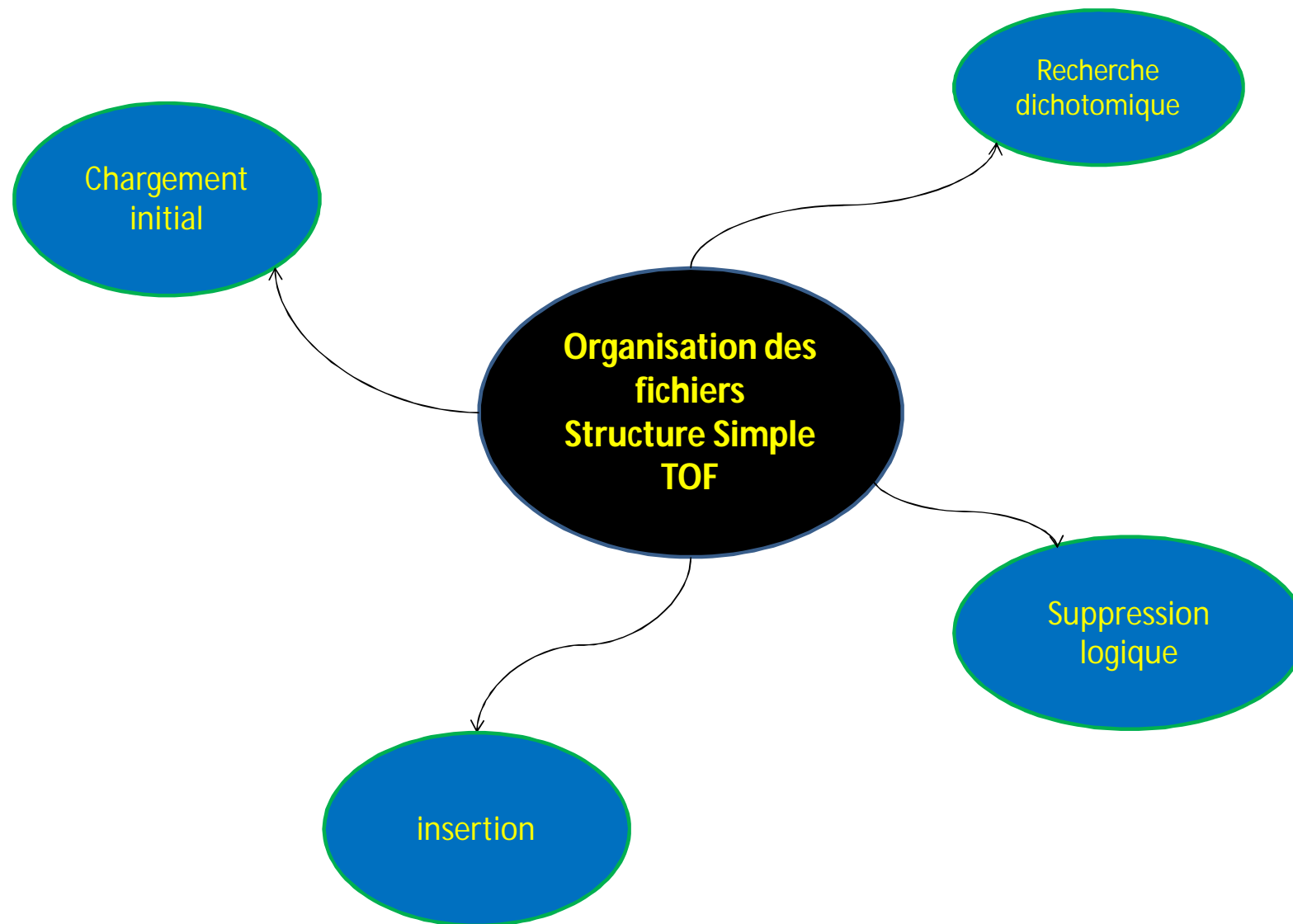
De Sidi Bel Abbès



مدرسة العليا للإعلام الآلي
بسيدي بلعباس

4. Conclusion

Ces techniques sont très efficaces pour le traitement séquentiel, c'est à dire quand le traitement consiste uniquement à appliquer une opération pour chaque élément du fichier. Si l'opération de recherche d'éléments est très fréquente, on préfère de loin utiliser un tableau ordonné avec format fixe. Ceci bien entendu pour pouvoir pratiquer la recherche dichotomique.



Une Carte Mentale de l'exposé

Références

Sitographie

<https://sites.google.com/a/esi.dz/hidouci/competences-professionnelles/algo2>

<http://zegour.esi.dz/Publication/Livre2/Partie-sdf/Part2-ss.htm>