

Structures de données avancées :

Introduction

Pr ZEGOUR DJAMEL EDDINE
Ecole Supérieure d'Informatique (ESI)
www.zegour.uniq.com
email: d_zegour@esi.dz

Structures de données avancées

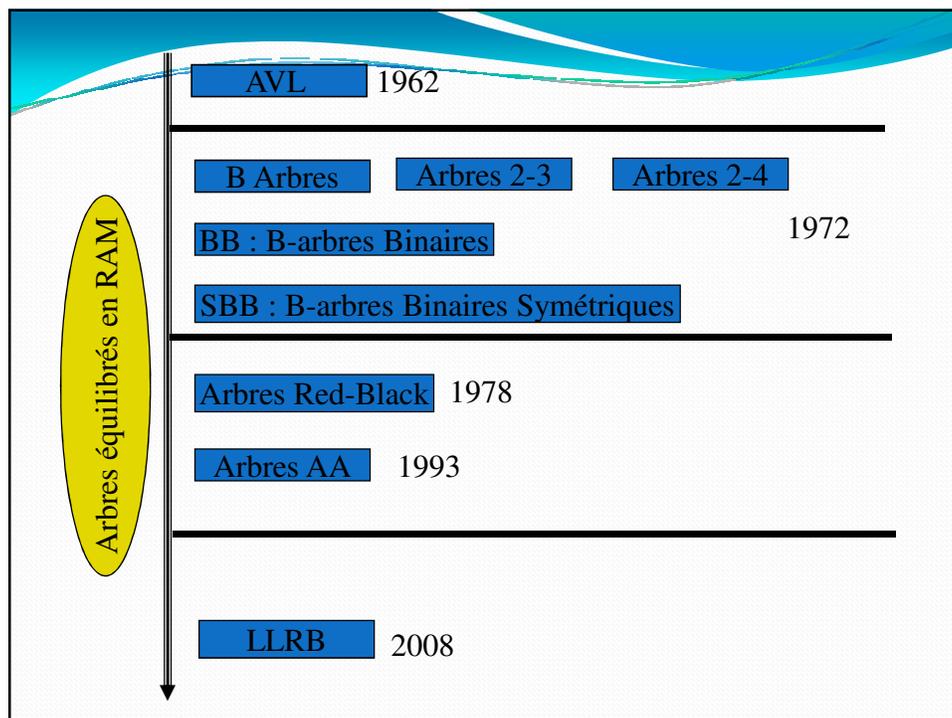
Deux parties

- **Structures de données en RAM : Les arbres équilibrés en RAM**
- **Structures de fichiers**

Structures de données avancées

Structures de données en RAM : Les arbres équilibrés

- Arbres AVL
- Arbres 2-3
- Arbre 2-4
- Arbres Red Black
- Arbres AA
- Arbres LL Red Black



Structures de données avancées

Structures de fichiers

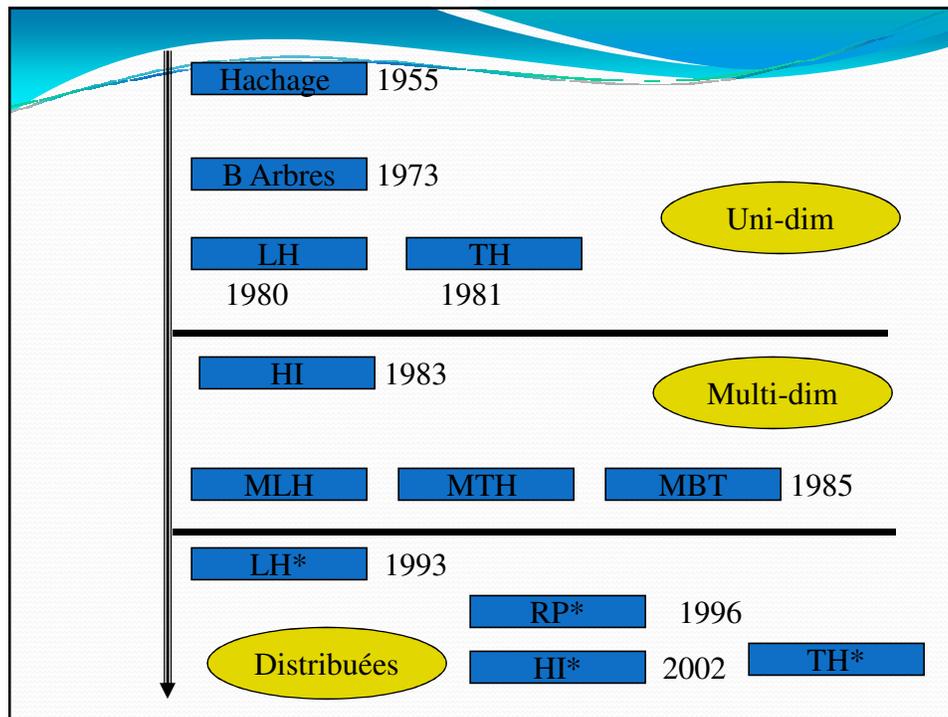
Plusieurs points de vue

- Organisation directe (table d'index ou hash-code) ou organisation d'arbre (tables d'index hiérarchisées)
- Simple attribut, plusieurs attributs
- Statique ou dynamique
- Préservation de l'ordre ou pas
- Local ou distribué

Structures de données avancées

Présentation des structures avancées de fichiers

- *Accès uni-dimensionnel* : Hachage, Arbres B ; Hachage Linéaire (LH), Hachage digital (TH)
- *Accès multi-dimensionnel* : MLH, MTH, MBT
- *Accès distribué* : LH*, TH*, RP*



Structures de données avancées

Pré-requis

- O-notation
- Pré-requis 1 : Présentation des structures de données en RAM.
- Pré-requis 2 : Introduction aux structures de fichiers et présentation des structures de fichiers de base.

O-notation

Structures de données avancées

O-notation

- Notation universelle pour mesurer des algorithmes dans le cas le plus défavorable.
- $T(n)$ est $O(f(n)) \iff$ Il existe $c, n_0 > 0, \forall n > n_0 \quad T(n) \leq c \cdot f(n)$
- Exemples $O(n^2)$, $O(\log_2(n))$, $O(n^{1.59})$,

Structures de données avancées

Mesure des algorithmes

- Cas des algorithmes itératifs
existence des règles (règle du produit, règle de la somme)

- Cas des algorithmes récursifs
 - déterminer l'équation de récurrence
 - résoudre l'équation
 - Equation différentielle
 - Dilatation de la récursivité
 - Principe de récurrence

- Cas des algorithmes à "Goto"
pratiquement impossible quand le degré de chevauchement est grand

Structures de données en RAM

Structures de données avancées

Allocation mémoire

- **Statique** : espace alloué avant l'exécution du programme
- **Dynamique** : espace alloué pendant l'exécution
- **Mixte** : mélange des deux (cas réel)

Structures de données avancées

Structures de données en RAM

- **Tableau** : suite contiguë d'entités
- **Listes** linéaires chaînées : ensemble de maillons alloués dynamiquement chaînés entre eux
- **Arbres** : structure de données hiérarchisée
- **Graphe** : structure plus complexe représentant des relations entre éléments
- **Hachage** : transformer une donnée (par une fonction) en une adresse où sera logée la donnée
Les techniques diffèrent par leur MRC (Méthodes de résolution des collisions)

Structures de données avancées

Stratégie et méthodologie

- Deux stratégies importantes
Les files d'attente : FIFO, Les piles : LIFO
- Méthodologie
 - *Analyse descendante* (modularité) : découper le problème en sous problèmes, re-découper chaque sous problème, s'arrêter au niveau de détail le plus bas ==> arborescence
 - *Récurivité* : définir un cas trivial, définir un cas général --> l'algorithme est généralement une alternative

Solution récursive plus élégante, solution plus chère

Structures de données avancées

Modèle ou type de données abstrait

- Écrire des algorithmes indépendamment de la représentation mémoire. Donc définition d'une machine abstraite pour chaque structure de données (Modèle) avec un ensemble d'opérations
- Implémentation des machines abstraites
Choix d'une représentation mémoire et la traduction des opérations du modèle dans cette représentation
- Exemple
Pile : Créerpile, Empiler, Depiler, Pilevide, Pilepleine

Introduction aux structures de fichiers

Structures de données avancées

Introduction

Pourquoi utiliser des mémoires secondaires?

Différence entre RAM et Disque

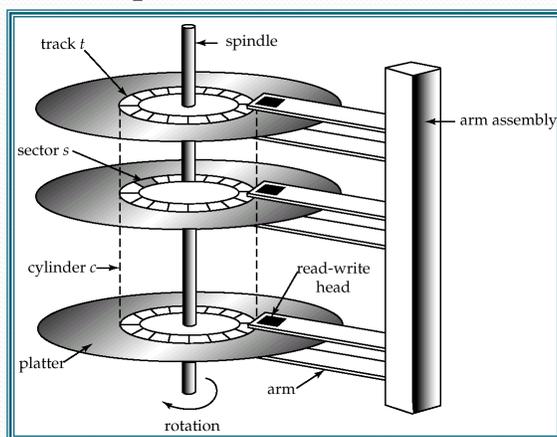
Fichier, fichier statique, fichier dynamique

Opérations fondamentales sur les fichiers

- . Fichiers physiques et fichiers logiques
- . Ouverture, Création et fermeture
- . Lecture et écriture
- . Détection de la fin de fichier
- . L'opération 'Seek'
- . Etc.

Structures de données avancées

Structure d'un disque



Structures de données avancées

Organisation des informations sur les disques

- **Secteur** : la plus petite unité de transfert.
- **Clusters** :
 - . nombre fixe de secteurs contigus.
 - . lecture sans bouger la tête de lecture/écriture.
 - . L'idéal : avoir des grands clusters.
- **Étendue** : portion du disque contenant des clusters contigus.
préférable d'avoir le fichier en une seule étendue.
- En général : fichier distribué sur plusieurs étendues.
- Les performances se dégradent quand le nombre d'étendues augmente

Structures de données avancées

Organisation des informations sur les disques

- **Bloc** : référence un groupe d'articles qui sont rangés ensemble sur le disque et traités comme une unité d'E/S.
L'idéal : un bloc = un cluster ou n secteurs
- **Fragmentation interne** : Présence de "trous" dans les blocs
 - * La taille d'un bloc n'est pas un multiple de la taille d'un article & l'article n'est rangé dans le bloc que si la place est disponible (apparition de "trous")
 - * L'article peut être à cheval sur deux bloc & suppressions ultérieures
- **Fragmentation externe** : Présence de blocs entièrement vides
Suppression de blocs, Fusion, etc.

Structures de données avancées

Organisation des informations sur bandes magnétiques

- Beaucoup moins importantes que les disques
- Restent efficace pour les traitements séquentiels

Structures de données avancées

Structures des articles

- Article = Ensemble de champs formant une même entité
- Format fixe ou variable
- Chevauchement ou pas des articles sur les blocs
- **Si format variable :**
 - Indicateur de longueur (nombre d'octets)
 - Délimiteur à la fin de chaque article

Structures de données avancées

Méthodes d'accès et organisation de fichier

- Méthode d'accès du fichier :
Approche utilisée pour localiser un article dans le fichier.
- Organisation du fichier :
Comment distinguer un article d'un autre
- Structure du fichier :
Méthode d'accès + Organisation du fichier

Structures de données avancées

Objectifs

- Combiner l'organisation du fichier et la méthode d'accès
 - Étudier les opérations de base :
recherche, insertion, suppression, requête à intervalle
- sous différentes méthodes d'accès :
- les méthodes classiques (Structures simples, méthodes d'index, méthodes d'arbre, méthode de hachage)
 - les nouvelles (LH, TH, MTH, RP*, ...)

Structures de données avancées

Critères de mesure d'une structure de fichiers

- **Taux d'occupation**=nombre d'articles / (nombre de blocs * capacité d'un bloc).
70 % : un bon compromis.
- Nombre d'accès disque (viser un accès)
- Encombrement et complexité des algorithmes
- Réaction de la méthode aux pannes systèmes