

Structures de données avancées : Hachage dynamique

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

Hachage dynamique

Introduction

- Hachage classique limité aux fichiers statiques → domaines d'utilisation restreints.
- La plupart des applications utilisent les fichiers dynamiques → maintenir un bon accès et un bon chargement du fichier quand le fichier se développe ou se rétrécit.

Hachage dynamique

Avant les années 80

- Arbres B et ses variantes.
(2 à 3 accès disque, Facteur de chargement aux environs de 70%,
Fichier dynamique et ordonné)
- Ces propriétés font des arbres B l'une des structures de fichiers
les plus populaires.

Hachage dynamique

A partir des années 80

- Les fichiers dynamiques peuvent être construits avec le hachage
- La nouvelle approche d'utilisation du hachage est appelée
hachage dynamique (hachage extensible ou hachage virtuel)
- Les possibilités de cette approche sont étonnantes car incapable
de les croire avant les années 80 (55 à 80).

Hachage dynamique

Du hachage classique au hachage dynamique

- Le hachage classique (au sens de KNUTH) suppose que la fonction de hachage est statique.
- Toute recherche de données est basée sur la même fonction dont le nombre d'adresses possibles est constant.
- Toute collision conduit à la création d'un débordement faisant ainsi décroître les performances d'accès.
- Tout algorithme apparu sur le hachage a permis tout simplement de minimiser cette détérioration.

Hachage dynamique

Du hachage classique au hachage dynamique

- Les débordements restaient toujours une fonction croissante du nombre d'insertions.
- La conception du hachage dynamique avait mis fin à ces idées admises depuis vingt cinq années (55 à 80).
- L'idée principal : changement dynamique de la fonction d'accès

Hachage dynamique

Du hachage classique au hachage dynamique

- Une insertion ou une suppression peut commencer par une fonction de hachage et terminer par une autre.
- En plus, l'espace des adresses possibles n'est plus limité.
- Le hachage dynamique fournit de bonnes performances de recherche et un facteur de chargement élevé indépendamment de la taille du fichier et sans réorganisation périodique.

Hachage dynamique

Hachage dynamique

- Naissance dès l'année 78 (Litwin et Larson)
- Depuis, plusieurs méthodes ont été développées
- Certaines de ces méthodes préservent également l'ordre des articles.

Hachage dynamique

Quelques résultats remarquables

- 1. Quelque soit le nombre d'insertions, une clé peut être trouvée en 1 accès (LH : Hachage linéaire)
- 2. Quelque soit le nombre d'insertions, une clé peut être trouvée en 1 accès, alors que le facteur de chargement peut rester égal à une valeur arbitraire dépassant les 95%.
(LH : Hachage linéaire contrôlé)
- 3. On peut construire des fichiers dynamiques ordonnés et de fichiers multidimensionnels sans index et une recherche prend 1 accès
(IH : Hachage par interpolation)

Hachage dynamique

Quelques résultats remarquables

- 4. On peut construire de grands fichiers dynamiques et ordonnés dans lesquels une recherche prend au plus 1 accès.
(TH : Hachage digital)
- 5. On peut construire de très grands fichiers dynamiques et ordonnés dans lesquels une recherche prend au plus 2 accès.
(Hachage digital multi-niveaux)

Hachage dynamique

Notion de fonction de split (division)

- Soit C l'espace des clés. Soit h_0 une fonction de H-code produisant N adresses : $N > 0$. Les fonctions de H-codes h_1, h_2, \dots, h_n sont des fonctions de split (division) pour h_0 si pour toute clé c
 - $h_i : C \rightarrow \{0, 1, \dots, 2^i \cdot N - 1\}$
 - $h_i(c) = h_{i-1}(c)$
ou bien
 $h_i(c) = h_{i-1}(c) + 2^{i-1} N$

Exemple : $h_i(c) = c \text{ Mod } 2^i N$

Hachage dynamique

Utilisation d'un arbre digital

- Fonction d'accès : arbre digital (structure de donnée compacte)
- Aussi longtemps que l'arbre peut rester en mémoire, toute recherche d'enregistrement aboutit en un accès disque au plus.