

Révision des Mots-Clés pour le Partiel

Référence : Folk, Zoellick and Riccardi – Section “Key Terms” à la fin de chaque chapitre.

Note : cette révision est seulement pour vous rafraichir la mémoire sur quelques mots-clés en relation avec les concepts étudiés dans ce cours; pour réviser chaque topic en détail il faut absolument se référer aux chapitres et aux notes du cours.

Chapitre 2 – Opérations fondamentales de traitement de fichier

Fichier physique – Fichier existant sur un moyen de stockage secondaire. C’est le fichier reconnu par le système d’exploitation de l’ordinateur et qui apparaît dans son répertoire de fichier.

Fichier logique – C’est le fichier vu par le programme. Les fichiers logiques permettent aux programmes à décrire les opérations qui vont être exécutées sur le fichier sans savoir quel est le fichier physique qui va être utilisé. Le programme peut être utilisé après pour traiter n’importe quel fichier parmi les fichiers qui ont la même structure.

Open « ouvrir » – C’est une fonction ou un appel système qui fait le fichier prêt à être utilisé. Elle peut encore relier un nom de fichier logique à un fichier physique. Ses arguments seront les noms des deux fichiers logique et physique et on peut inclure encore le type d’accès (lecture, écriture, etc.).

Close « fermer » – C’est une fonction ou un appel système qui annule la liaison entre le nom du fichier logique et le nom du fichier physique correspondant.

Read « lire » - C’est une fonction ou un appel système utilisé pour obtenir des informations d’entrée d’un fichier. Au plus bas niveau, trois arguments sont requis : (1) nom du fichier logique source correspondant à une ouverture de fichier; (2) l’adresse de destination des octets à lire; et (3) la taille ou la quantité des données à lire.

Write « écrire » - C’est une fonction ou un appel système utilisé pour fournir des capacités de sortie. Au plus bas niveau, trois arguments sont requis par cette fonction : (1) nom du fichier logique de destination correspondant à une ouverture de fichier; (2) l’adresse source des octets à écrire; et (3) la taille ou la quantité des données à écrire.

Seek « chercher » - C’est une fonction ou un appel système qui initialise le pointeur lire/écrire à une position donnée dans le fichier. Les langages qui fournissent des fonctions de recherche permettent aux programmes l’accès direct aux éléments donnés d’un fichier au lieu de faire une recherche séquentielle dès le début chaque fois un élément donné est désiré.

Chapitre 3 – Stockage Secondaire et Logiciel Système

Disques Magnétiques

Sector Organization «Organisation Secteur» - Organisation de disque qui utilise les secteurs.

Block Organization «Organisation Bloc» - Organisation de disque qui permet à l'utilisateur de définir la taille et l'organisation des blocs et puis d'accéder un bloc tout en donnant l'adresse du bloc ou la clé d'un de ses enregistrements.

Secteur – Les blocs de données de taille fixe qui forment ensemble les pistes sur certains types de disques. Les secteurs sont la plus petite unité adressable sur un disque dont les pistes sont formées de secteurs.

Pistes – Ensemble d'octets sur une surface d'un disque qui sont accédés sans recherche (sans déplacer le bras d'accès). On peut imaginer la surface d'un disque comme une série de cercles concentriques tel que chaque cercle correspond à une position donnée du bras d'accès et des têtes de lecture/écriture.

Cylindre – Ensemble de pistes, sur un disque, logés directement les uns sous les autres. Toutes les pistes d'un cylindre donné peuvent être accédées sans déplacer le bras d'accès – ils peuvent être accédés sans perdre du temps, de recherche.

Nombre Sous-Bloc (Count Subblock)- Dans une organisation bloc, un petit bloc précède chaque bloc de donnée et contient des informations sur le bloc des données, comme par exemple nombre d'octets et son adresse.

Temps de Recherche – C'est le temps requis pour déplacer le bras d'accès à un cylindre donné sur un disque.

Délai de Rotation – le temps pris par une rotation de disque pour positionner un secteur donné sous la tête de lecture/écriture.

Temps de Transfert – Une fois les données qu'on veut sont sous la tête de lecture/écriture, on doit attendre pour qu'elles passent sous la tête au moment de la lecture. Le temps requis pour ce mouvement et pour la lecture est le temps de transfert.

Groupement (cluster) – C'est l'unité minimale d'allocation d'espace sur un disque sectorisé, et consistant en un ou plusieurs secteurs contigus. L'usage de gros groupements peut améliorer le temps d'accès en assurant la lecture d'une plus grande quantité de données sans recourir à la recherche. Les petits groupements tendent à minimiser la fragmentation interne.

Étendue – C’est la représentation d’une partie (ou de la totalité) du fichier par un ou plusieurs groupements adjacents. Plus le nombre d’étendues augmente, plus le fichier est dispersé sur le disque et plus il requiert de temps de recherche en se déplaçant d’une partie du fichier à l’autre.

Bandes Magnétiques

Facteur de bloc – le nombre d’enregistrements stockés sur un bloc.

La densité d’enregistrement effective – Mesure la densité en prenant compte de l’espace interbloc, du sous-blocs sans données (vides), et autres entités consommatrices d’espace accompagnant les données.

La densité d’enregistrement nominale – Mesure la densité sur une piste de disque ou de bande magnétique sans prendre compte des effets de l’espace interbloc ou des sous-blocs vides.

Taux de transmission effective – Taux de transmission qui dépend du temps utilisé pour localiser et transmettre le bloc de donnée où l’enregistrement désiré réside.

Taux de transmission nominal des données – Taux de transmission de données d’un disque ou d’une bande sans prendre compte des effets de telles opérations comme le temps de recherche des disques et le temps de traverse d’interbloc pour les bandes.

CD-ROM

(Pas de mots-clés incluent, mais les étudiants doivent réviser ce concept).

Logiciel Système

Gestionnaire de Fichier – La partie du système d’exploitation responsable de la gestion des fichiers, contenant une collection des programmes dont la responsabilité varie de garder les traces des fichiers à invoquer des processus d’entrée/sortie qui transmettent des informations entre des unités de stockages primaire et secondaire.

Table d’allocation des fichiers – C’est une table contenant des liens aux adresses physiques de tous les groupements sur tous les fichiers sur disque.

Processeur d’entrée/sortie – Composant qui effectue les fonctions d’entrée/sortie, ce qui permet au processeur central (CPU) de concentrer sur les autres fonctions.

Contrôleur – Composant qui contrôle directement le fonctionnement d’un ou de plusieurs unités de stockages secondaires, comme les disques et les bandes magnétiques.

Tampon – C’est quand l’entrée/sortie est sauvegardée dans une mémoire tampon au lieu qu’elle soit transmise directement à sa destination. Dans les chapitres suivants, on trouve que la performance des programmes qui lisent et écrivent les données peuvent être améliorées en cas l’entrée/sortie a été sauvegardé dans une mémoire tampon.

Chapitre 4 – Concepts Fondamentaux des Structures des Fichiers

Champ – La plus petite unité logique d’information dans un fichier. Un enregistrement dans un fichier contient généralement plusieurs champs.

Enregistrement – Une collection de champs en relation. Par exemple, un enregistrement ‘individu’ dans un fichier de liste de diffusion peut contenir des champs comme le nom, l’adresse, et autres informations sur cet individu.

Enregistrement à taille fixe – Une organisation de fichier dont tous les enregistrements ont la même taille. Les enregistrements sont complétés avec des blancs, nulles ou autres caractères de façon à étendre l’enregistrement jusqu’à la taille fixe. Lorsque tous les enregistrements ont la même taille, on peut donc calculer la position initiale de n’importe quel enregistrement ce qui fait l’accès direct possible.

Enregistrement à taille variable – Une organisation de fichier dont les enregistrements n’ont pas une taille prédéterminée. Ils ont juste la taille nécessaire ce qui améliore l’usage de l’espace mémoire relativement aux enregistrements à taille fixe. Malheureusement on ne peut pas calculer l’octet-offset de ce genre d’enregistrements si on sait seulement le numéro de l’enregistrement relatif.

Chapitre 5 – Gestion des Fichiers d’Enregistrements

Clé – Une expression dérivée d’un ou de plusieurs champs d’un enregistrement et qui peut être utilisée pour localiser cet enregistrement. Les champs utilisés dans la création de clé sont parfois nommés des *champs clés*. L’accès par clé permet d’effectuer une recherche basée sur le contenu plutôt qu’une recherche basée sur la position de l’enregistrement.

Clé primaire – C’est une clé d’identification unique de chaque enregistrement. Elles sont utilisées comme la méthode primaire d’accès aux enregistrements.

Accès séquentiel – C’est l’accès aux données nécessaires en lisant le fichier dès le début. L’autre alternative est l’accès direct.

Accès direct – C'est l'accès aux données nécessaires en sautant directement à l'enregistrement donné. L'accès direct à un enregistrement à taille fixe est généralement effectué par l'usage de son numéro d'enregistrement relatif, le calcul de son octet-offset et puis par la recherche du premier octet de l'enregistrement.

Numéro d'enregistrement relatif (NER)– Un indice donnant la position d'un enregistrement relativement au début du fichier. Si un fichier est composé d'enregistrements à taille fixe le NER peut être utilisé pour calculer l'octet-offset d'un enregistrement pour que cet enregistrement soit accédé directement.

Chapitre 6 – Organisation des Fichiers pour l'amélioration de la performance

Compression de données

Compression de données – Encodage d'informations dans un fichier de façon à prendre moins d'espace.

Encodage Huffman – Encodage à taille variable où les tailles des codes sont basées sur leurs probabilités d'occurrence.

Encodage Lempel-Ziv – Technique de compression dynamique que vous avez déjà étudié en détail.

Run-length encoding – Une méthode de compression où on remplace les répétitions d'une valeur par le nombre de fois qu'elle est répétée suivi par la valeur répétée.

Récupération d'espace dans les fichiers

Fragmentation – C'est l'espace inutilisé dans un fichier. L'espace peut être verrouillé à l'intérieur d'un enregistrement (fragmentation interne) ou entre des enregistrements (fragmentation externe).

Fragmentation interne – Forme de fragmentation qui s'effectue en cas d'espace perdu dans un fichier parce qu'il était verrouillé et inutilisé à l'intérieur de l'enregistrement. Les structures des fichiers à taille fixe généralement causent des fragmentations internes.

Fragmentation externe – Forme de fragmentation qui s'effectue quand l'espace inutilisé du fichier existe en dehors ou entre des enregistrements individuels.

Avail list – Une liste d'espace vidé à l'aide d'effacement des enregistrements et qui est prêts à contenir de nouveaux enregistrements. Dans l'exemple considéré dans ce chapitre, cette liste d'espace a pris la forme d'une liste d'enregistrements effacés.

Stratégie de placement – C'est un mécanisme de sélectionner l'espace, de la liste d'espace, qui va contenir le nouvel enregistrement. (par exemple first fit, best fit, worst fit)

Tri Interne/Externe et Recherche Binaire

Recherche Binaire – C'est un algorithme qui localise une clé dans une liste triée par une répétition de la sélection de l'élément au milieu de la liste, et puis par la division de la liste en deux parties et la formation d'une nouvelle liste, plus petite, de la partie contenant la clé. Ce processus va continuer jusqu'à que l'élément sélectionné soit la clé demandée.

Tri Interne – Méthode de trier les données dans la mémoire centrale.

Tri Externe - Méthode de trier les données qui ne peuvent pas exister simultanément dans la mémoire centrale (par exemple en cas la taille des données est supérieure à celle de la mémoire centrale).

Tri par Clé

Tri par Clé – C'est une méthode de trier un fichier qui ne doit pas nécessairement exister entièrement dans la mémoire. La mémoire contient seulement les clés et les pointeurs qui lient ces clés aux enregistrements dans le fichier original. On tri les clés et puis on utilise la liste triée des clés pour construire une nouvelle version triée du fichier. L'avantage primaire de ce tri c'est qu'il requiert moins d'espace que le tri de mémoire. Le fait que la construction d'un nouveau fichier requière beaucoup de recherche d'enregistrements pose un désavantage pour ce tri.

Enregistrements fixes (pinned records) - un enregistrement est fixe si autres enregistrements ou structures de fichier le font référence à travers son adresse physique. C'est fixe de manière qu'on ne puisse pas changer son adresse physique, sinon on détruit la validité du référence physique à l'enregistrement. Ces références seront des *pointeurs pendants* inutiles.

Chapitre 7 – Construction d'Indexes

Index – C'est un outil pour trouver les enregistrements dans un fichier. Il contient un champ clé où l'indexe va être chercher et un champ de référence qui nous donne l'adresse de l'enregistrement des données associé à une clé particulière.

Inverted list – Indexes dont la clé peut être associée à une liste de champs de références qui pointent aux documents contenant la clé. Les indexes secondaires discutées à la fin de ce chapitre sont des exemples de *inverted lists*.

Fichier à entrée séquentielle – C’est un fichier dont les enregistrements sont placés selon leur ordre d’entrée dans le fichier.

Attachement – L’attachement prend place quand la clé associée à un enregistrement physique particulier dans un fichier de données. En général, l’attachement prend place au moment de la construction du fichier ou au moment de l’utilisation. Dans le premier cas, nommé, *tight binding*, les indexes contiennent des références explicites aux enregistrements physiques associés. Dans le deuxième cas, la connexion entre une clé et un enregistrement physique particulier est retardée jusqu’à que l’enregistrement soit cherché pendant l’exécution du programme.

Chapitre 8 – Traitement Co-séquentiel (jusqu’à section 8.1 seulement)

Opérations Co-séquentielles – Applications appliquées aux problèmes de performance de l’union, l’intersection et des sets d’opérations plus complexes sur deux ou plusieurs fichiers d’entrée triés pour produire un ou plusieurs fichiers de sortie construits de quelques combinaisons d’éléments de fichiers d’entrée. Les opérations co-séquentielles sont utiles pour la fusion (l’union) des éléments des deux listes, pour l’assortiment (*matching*) de deux listes (ou leur intersection) et pour les problèmes de modification des fichiers.

Assortiment (Match) – C’est le processus de formation d’un fichier de sortie trié à partir de tous les éléments communs à deux ou plusieurs fichiers d’entrée triés.

Fusion (Merge) – C’est le processus de formation d’un fichier de sortie trié à partir de l’union des éléments de deux ou plusieurs fichiers d’entrée triés.