Série d'exercices #15

IFT-2245

7 avril 2019

15.1 Systèmes de fichiers

Soit un système de fichiers de style "Unix" tel que le FFS de BSD ou ext2 de Linux, avec des blocs indexés.

1. Lister tous les blocs qu'il faut modifier sur le disque lors de l'exécution de la fonction `open` qui crée un fichier (de taille zéro).
2. Spécifier l'ordre dans lequel ces opérations devraient être exécutées pour minimiser l'impact potentiel d'un crash à mi-course.
3. Sur la base de l'ordre précédent, indiquer après chaque opération quels problèmes apparaîtraient en cas de crash à ce moment.
4. Si le système de fichier est modifié pour garder un *journal*, indiquer quelles données devraient être écrites dans le journal pour l'opération précédente.
5. Estimer le coût en performance d'un tel journal pour cette opération ; i.e. indiquer de combien de pourcents l'opération serait-elle ralentie.

15.2 Mount multiples

Quels problèmes peuvent apparaître quand on autorise un système de fichiers à être *monté* à plusieurs endroits en même temps.

15.3 VFS

Discuter de l'usage d'une abstraction nommée *VFS* pour permettre à un système d'exploitation d'utiliser facilement plusieurs sortes de systèmes de fichiers.

15.4 Blocs libres

Soit un système où l'espace libre est maintenu dans une liste chaînée de blocs libres.

1. Supposons que le pointeur sur le premier bloc libre est perdu. Le système peut-il reconstruire la liste des blocs libres ?

15.5 Optimisation et pannes

Discuter comment les optimisations de performance pour les systèmes de fichiers peuvent introduire des problèmes de cohérence en cas de panne.

15.6 Indexage indirect progressif

Soit un système de fichiers de type "Unix File System" avec des blocs de 8KB, où un pointeur sur un bloc occupe 4 bytes, et où chaque inode contient 12 pointeurs sur des blocs directs, 1 pointeur sur un bloc indirect, 1 pointeur sur un bloc doublement indirect, et 1 pointeur sur un bloc triplement indirect.

— Quelle est la taille maximum d'un fichier ?
— Combien d'accès disques sont nécessaires (en présumant que le cache est vide) pour accéder au contenu d'un petit fichier dans `/a/b/c` ?