

Ressource R3.05 - TD 2

Fichiers

1. Dans un système de fichiers **Ext2** (blocs de 4ko, pointeur de bloc sur 4 octets), on désire créer un fichier contenant un total de 20.000.000 octets. Quelle est la fragmentation interne totale ("espace gaspillé") sur le disque résultant de la création de ce fichier ?
2. **File** est un fichier régulier d'un système **Ext2**. Il occupe 458 blocs (**en tout**) de 1024 octets. (On suppose les numéros de blocs occupent 4 octets)
 - (a) Quel peut être la taille du fichier ?
 - (b) Quelle est la taille maximale d'un fichier dans un tel système ?
 - (c) Combien d'accès nécessite la lecture séquentielle du 421-ième bloc de données par un processus sachant que la lecture se fait par bloc de 1024 octets (on suppose que le système utilise un cache) ?
 - (d) Même question pour un accès direct ?
3. Un processus lit séquentiellement un fichier **Ext2** de **8Mo**, par tranche de 256 octets. Les blocs du système de fichier sont de 1 Ko, et un numéro de bloc occupe 4 octets. Le temps d'accès au disque est de 50ms.
 - (a) Sans cache, donnez le nombre d'accès disque et le temps de lecture totale du fichier.
 - (b) On suppose que le système maintient en cache les 100 blocs les plus récemment accédés. Donnez le nombre total d'accès disque, et le temps de lecture totale du fichier.
4. On veut créer :

- un fichier **p1.c** de 2000 octets à la racine.
 - un répertoire **Rep1** à la racine.
 - un répertoire **Rep2** dans **Rep1**.
 - un répertoire **Rep3** dans **Rep2**.
 - un fichier **p2.c** de 800 octets dans **Rep3**.

/

|__ p1.c

|__ Rep1

|__ Rep2

|__ Rep3

|__ p2.c

Les inodes libres sont alloués dans l'ordre suivant : 202, 25, 17, 182, 232. Les blocs de données libres de 512 octets sont alloués dans l'ordre suivant : 5600, 4587, 14123, 7821, 5551, 12899, 4536, 8786, 125. Le répertoire racine se trouve dans le bloc 8251.

- (a) Pour chaque inode, donnez la liste des blocs utilisés par le fichier ou répertoire représenté par cet inode.
- (b) Pour chaque bloc de données de répertoire, donnez le contenu de ce répertoire.
- (c) Quelles manipulations et dans quel ordre faut-il effectuer au niveau des inodes, bitmaps, direntries, blocs etc. pour exécuter la commande **ls /p1.c**?
- (d) Même question pour **rm /Rep1/Rep2/Rep3/p2.c**.
- (e) Même question pour **mv /p1.c /Rep1/**.

5. Voici le contenu d'un inode d'un système de fichiers ext3

```
Inode: 12   Type: directory   Mode: 0755   Flags: 0x0
Generation: 3389598991   Version: 0x00000007
User:      0   Group:      0   Size: 1024
File ACL: 0
Links: 4   Blockcount: 2
Fragment: Address: 0   Number: 0   Size: 0
ctime: 0x5c5077d9 -- Tue Jan 29 16:57:13 2019
atime: 0x5c5078e0 -- Tue Jan 29 17:01:36 2019
mtime: 0x5c5077d9 -- Tue Jan 29 16:57:13 2019
BLOCKS:
(0):1538
TOTAL: 1
```

Voici le contenu du bloc 1538

```
00000000  0c 00 00 00 0c 00 01 02 2e 00 00 00 02 00 00 00 |.....|
00000010  18 00 02 02 2e 2e 00 00 0e 00 00 00 0c 00 03 02 |.....|
00000020  53 72 63 00 0f 00 00 00 0c 00 04 02 44 6f 63 73 |Src.....Docs|
00000030  10 00 00 00 10 00 06 01 70 72 6f 67 2e 63 00 00 |.....prog.c..|
00000040  11 00 00 00 14 00 09 01 70 68 6f 74 6f 2e 6a 70 |.....photo.jp|
00000050  67 00 00 00 12 00 00 00 ac 03 07 02 2e 63 6f 6e |g.....con|
00000060  66 69 67 00 00 00 00 00 00 00 00 00 00 00 00 00 |fig.....|
00000070  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |.....|
*
00000400
```

Donnez la **liste de toutes les entrées** du répertoire (inode et nom) correspondant. Voici la structure de chaque entrée.

```
struct ext3_dir_entry_2 {
    __u32      inode;           /* Inode number */
    __u16      rec_len;         /* Directory entry length */
    __u8       name_len;        /* Name length */
    __u8       file_type;
    char       name[EXT3_NAME_LEN]; /* File name */
};
```
