

Compte rendu - réunion - 29 Novembre 2011

December 4, 2011

Présents

- Florent NICART (Client)
- Zakaria ADDI
- Baptiste DOLBEAU
- Zineb ISSAAD
- Pierre LUCE-VEYRAC
- Emmanuel MOCQUET
- Claire SMETS

Absents

Aucun

Titre

Réunion

Ordre du jour

1. Utilisation du cache
2. Présence d'un swap
3. Duplication de certains blocs
4. Gestion du cache
5. Niveau d'abstraction
6. Fichiers "fragmentés"
7. Option dans la gestion du cache
8. Travailler avec un HDD
9. Déléguer
10. Gestion des blocs défectueux

1 Utilisation du cache

On se servira d'une partie du SSD pour servir de cache non volatile pour suppléer le cache du disque dur.

2 Présence d'un swap

L'utilisation d'un swap n'a pas été demandée par le client, rendant donc son implantation inutile sur le SSD.

Cependant il pourrait être intéressant de pouvoir partitionner le système hybride, afin éventuellement d'y mettre un swap **gérée par l'OS**.

Il faut le moins de couches possible entre le swap et le système.

3 Duplication de certains blocs

Certains blocs pourront être présents à la fois sur le SSD et sur le HDD :

On veut un système hybride rapide et économe : pour cela on ne souhaite allumer le HDD que quand on a besoin. Il faut donc stocker le plus d'informations possible sur le SSD.

4 Gestion du cache

Quand des informations sont lues sur la mémoire de masse, il faut les mettre dans le cache si elles n'y sont pas déjà. Et quand il y a écriture il faut le faire seulement sur le SSD. Ne pas oublier que le SSD, bien que considéré comme une mémoire cache, est non volatile. Il n'y a donc pas besoin de faire une sauvegarde systématique sur le HDD.

Mais il y a quand même un mécanisme de synchronisation à mettre en place : quand le SSD sera saturé, il faudra transférer des données vers le HDD afin de libérer de la mémoire flash. À noter que pour tester le mécanisme que l'on mettra en place, on pourra utiliser une petite partition du SSD pour pouvoir la saturer plus vite.

Afin de savoir quels blocs transférer vers le HDD il faudra aussi maintenir une liste des accès : lesquels sont les plus souvent utilisés, lesquels sont les plus récents, ... tout dépendra de l'algorithme que l'on choisira.

Il est inutile de marquer certains blocs du SSD comme permanents : s'il est assez grand, l'algorithme mis en œuvre gèrera automatiquement le fait que certains fichiers resteront en permanence sur le SSD.

5 Niveau d'abstraction

À notre niveau nous ne savons pas ce qu'est un fichier. On demande simplement des blocs et leurs adresses logiques. Nous implanterons donc un périphérique de type blocs. Des recherches sur le sujet sont nécessaires.

Il y aura des conversions d'adresses à faire. Et il faudra réimplanter les fonctions read et write.

6 Fichiers ”fragmentés”

Cette section n’a sûrement pas sa place ici étant donné que nous ne faisons que manipuler des blocs et des adresses mémoires.

Les fichiers peuvent n’être que partiellement chargés sur le SSD : si le fichiers est très gros et qu’on y fait des accès séquentiels, il devient coûteux et en espace et en temps, et inutile de le charger entièrement sur le SSD (s’il n’y était pas déjà). De plus comme dit précédemment nous ne maîtrisons pas à le niveau le concept de fichier.

C’est par exemple particulièrement vrai pour les machines virtuelles.

Les fichiers peuvent donc ne pas être stockés intégralement sur l’un ou l’autre des périphériques.

7 Option dans la gestion du cache

Il pourrait être intéressant que quelle que soit leur priorité, certains blocs soient constamment présents sur le SSD. =>faire un marquage à la volée?

Cette option n’est cependant pas demandée par le client.

8 Travailler avec un HDD

Il faudrait avoir la possibilité de pouvoir travailler avec le HDD seul. Pour cela il faudrait s’assurer que l’intégralité du contenu du SSD soit effectivement stockée sur le HDD. Il y a trois opérations à effectuer :

1. rediriger les écritures vers le disque mécanique uniquement
2. copier les blocs qui sont présents uniquement sur le SSD vers le disque mécanique
3. désassocier le SSD de notre système pour le rendre prêt à être démonter

Ces trois opérations doivent se faire obligatoirement de façon atomique.

On peut enlever le SSD mais pas le HDD : la mémoire du HDD est plus importante. Cependant même si le HDD fonctionne seul il utilisera quand même le logiciel développé par nos soins : il y aura des informations supplémentaires par rapport au système de fichiers. Il nous faudra donc réserver un espace sur le périphérique pour les stocker.

9 Déléguer

On ne travaillera pas avec le SSD directement. La plupart du temps on déléguera et on s’appuiera sur des routines déjà présentes.

10 Gestion des blocs défectueux

On ne gèrera les blocs défectueux que quand les pool des périphériques seront vides. Avant on ne verra pas passer les erreurs.