# Le coin des perfs...

*Jean-Yves BRUCKER*

## Lecture / Ecriture sur disques

### Mais où sont donc passées mes données ?

Lors des opérations d'accès sur disques, vous pouvez constater des variations de débit allant de quelques centaines de méga-octets par seconde à quelques centaines de kilo-octets par seconde.

C'est l'emplacement des données qui est à l'origine de ces écarts de performances qui peuvent aller jusqu'à un facteur 1000.
Voici, classés dans l'ordre de la rapidité d'accès, du plus rapide au plus lent, les endroits où peuvent se situer les données.

### Lectures sur disques

#### Lectures aléatoires de 4 Ko en mode JFS

Cache de niveau 1 (interne au processeur) ⟶ vitesse du processeur

Cache de niveau 2 ⟶ vitesse : 10 Go/s

Cache de la base de données (par exemple : SGA) (mémoire interne) ⟶ vitesse en Go/s

I/O cache d'AIX (mémoire interne) ⟶ vitesse en Go/s

Mémoire de la baie de disques ⟶ vitesse : 10 à 100 Mo/s

Cache du disque ⟶ vitesse : 40 Mo/s maxi

Disque ⟶ durée en ms

Dans le cas d'une lecture aléatoire de données d'Oracle™ sur un système de 8 Go équipé d'une baie de disques ESS (ou autre), le débit moyen sera de l'ordre de 500 Ko/s.

## Lectures séquentielles de 4 Ko en mode JFS

Les deux premières lectures correspondent à des lectures aléatoires.
Les suivantes seront principalement des lectures en mémoire, du fait de la lecture en mode "*read-ahead*".

Dans ce cas :

- Les caches de niveau 1 ne contiennent jamais les données.
- Les caches de niveau 2 sont chargés par le *read-ahead* et n'influencent que très peu les performances.

Cache de la base de données (mémoire interne)

I/O cache d'AIX (mémoire interne)

Mémoire de la baie de disques

Cache du disque

Disque

Pour des données uniquement situées sur les disques, le débit sera de l'ordre de 3 à 15 Mo/s suivant les configurations.

## Lectures séquentielles ou aléatoires de 4 Ko en mode RAW

Dans ce cas, l'impact des caches de niveaux 1 et 2 est lié au fonctionnement de la base de données.
Il est à noter qu'aucun mécanisme de cache n'est géré par AIX.

Cache de la base de données

Mémoire de la baie de disques

Cache du disque

Disque

Ces lectures seront plus lentes pour des blocs de 4Ko (absence des caches AIX et du *read-ahead*) mais plus rapides pour des blocs de taille importante (> 256 Ko).

## Accès aux données d'un disque

L'accès aux données d'un disque impose de déplacer le bras porte-têtes pour l'aligner sur le bon cylindre, puis d'attendre que les données passent devant la tête de lecture.
Pour un disque qui tourne à 15 000 tours/minute, cette attente peut durer jusqu'à 60/15000 = 0,004 s, soit 4 millisecondes, uniquement pour la rotation du disque. D'où l'intérêt du cache de disque, qui sera utilisé pour enregistrer la fin de la piste en cas de besoin ultérieur.
Les disques "modernes" contiennent un processeur qui est capable d'optimiser et d'anticiper les lectures en fonction des ordres SCSI qui sont reçus. L'installation des nouveaux microcodes de disques peut permettre d'améliorer les performances.

## Stripping or not stripping

Le *stripping*(1) peut améliorer les performances d'un traitement séquentiel, en fournissant X disques au service d'un processus.
A l'inverse, plusieurs processus utilisant les mêmes disques en mode "*stripping*" entraîneront des mouvements importants des bras ainsi que des purges des caches.

Recommandations :

- Utiliser le *stripping* AIX si un seul processus doit accéder aux données de ces disques.
- Ne jamais utiliser l'option "**stripping**" de la commande "**mklv**" (smit mklv) si les données ont déjà subi du *stripping* au niveau de l'adaptateur ou de la baie de disques (risque de conflit entre les algorithmes).
- Utiliser du *stripping* de partition (valeur : "**Range of physical volumes=maximum**" dans la création de "logical volume") pour les applications où plusieurs programmes doivent accéder aux disques simultanément. Ce *stripping* permet un équilibrage des disques physiques ou logiques (RAID 5, RAID 0 ou baie ESS). En cas d'utilisation d'un "*cache write*", il permet en outre l'utilisation de plusieurs caches pour un même disque logique.

**Note (1) :** Le terme *stripping* correspond au découpage séquentiel d'un espace de données et à sa répartition sur différents disques physiques.
Ce découpage peut prendre différentes formes :
- De 4Ko à 128Ko : *stripping* au niveau du *logical volume* avec l'option "**-S**" de la commande "**mklv**" (smit mklv).
- De 32Ko à 128Ko : *stripping* au niveau interne de l'adaptateur (RAID 5, RAID 0 et baie de type ESS). Ces valeurs sont en général non modifiables.
- De 4Mo à 128Mo : *stripping* au niveau des partitions.

## Ecritures sur disques

### Ecritures Synchrones

Dans ce cas, le programme attend que les données soient physiquement écrites dans le cache de l'adaptateur (pour les adaptateurs SSA uniquement), dans le cache-écriture de la baie ou sur le disque.

Le débit d'écriture pour des blocs de 4 Ko est de l'ordre de 300 Ko/seconde.

### Ecritures Asynchrones

Dans ce cas, les données sont écrites en mémoire et le programme peut poursuivre son traitement.

Si ces écritures impliquent un changement de taille du fichier, le jfslog enregistrera la modification en mode synchrone. L'écriture sera effectuée ultérieurement par le syncd (toutes les minutes en général).

Le débit est ici de l'ordre de plusieurs méga-octets/seconde.

# Conclusion

- Lorsque vous effectuez des mesures sur les disques, il faut à tout moment vous poser la question : "Où sont mes données et où vont-elles ?"
- Méfiez-vous aussi des *benchmarks* qui montrent d'importantes différences de performances entre différentes baies, disques ou systèmes.
- Les disques sont les éléments d'un système qui évoluent le moins vite du fait de la mécanique ; le seul moyen d'augmenter le débit est l'accès aux données en parallèle.
- Si vous avez des doutes sur les performances de vos disques, posez-vous les questions suivantes :
    - **Où sont mes données ?**
    - **Les caches sont-ils saturés ?**
    - **Mon traitement est-il séquentiel ou parallèle ?**

■