

TP Évaluation de performances

Sebastien GOUGEAUD

2019/2020

Quelques informations avant de démarrer ce TP :

- le langage de programmation utilisé pour le traitement des données et la génération des graphes est laissé à votre choix ;
- le TP, sera à rendre par mail à l'adresse suivante : `pro.seb.gougeaud@gmail.com` ;
- le rendu du mail consistera en une archive (zip, tar, tar.gz) contenant le code permettant de traiter la **deuxième** question de l'exercice, le matériel de compilation (si nécessaire), le compte-rendu ainsi qu'un fichier `README.md` décrivant le contenu de votre archive et comment utiliser votre programme ;
- le rendu devra être fait au plus tard le 9 février 2020, 23h59 (heure française métropolitaine) ;
- les rendus seront comparés les uns aux autres, et des points seront enlevés s'il y a une correspondance suspecte entre eux.

1 Flexible I/O Tester

Nous allons utiliser pour ce TP, le logiciel `fio`¹ qui a pour but de générer des tâches I/O et de reporter dans un fichier de résultats les performances obtenues. Pour un système Ubuntu, l'installation se fait simplement à l'aide de la commande suivante. Pour les autres systèmes, veuillez vous référer à la documentation présente sur le dépôt.

```
# apt-get install fio
```

L'utilisation de `fio` se fait en ligne de commande. Une manpage est disponible² et vous décrit l'ensemble des arguments disponibles et le contenu des affichages de fin de tâche. `fio` peut prendre en argument le chemin d'un fichier de tâche, contenant les paramètres à utiliser. Des exemples sont disponibles sur le dépôt.

2 Consignes

En utilisant `fio`, vous allez devoir répondre à des énoncés vous demandant d'effectuer des comparaisons entre plusieurs tâches I/O en faisant varier un (ou plusieurs) paramètre(s). Ces comparaisons prendront la forme de graphes de bande-passante générés par vos soins et d'un texte mettant en avant les différences observées et présentant une réflexion sur la raison de ces différences. Pour l'énoncé 3, vous présenterez également un graphe de déciles des latences observées.

1. dépôt : <https://github.com/axboe/fio>

2. version online : <https://linux.die.net/man/1/fio>

3 Énoncés

Considérez les paramètres par défaut suivants (sauf si renseignés dans l'énoncé) :

PARAMÈTRE	VALEUR PAR DÉFAUT	CLÉ FIO
nombre de répétitions	5	—
temps d'exécution	60s	<code>runtime/time_based</code>
nombre de processus	1	<code>numjobs</code>
pourcentage d'écriture	33%	<code>rwmixwrite</code>
taille de requête	16ko	<code>blocksize</code>
taille de l'espace d'adressage	1Go	<code>filesize</code>
accès aux données	aléatoire	<code>randrw</code>
buffer I/O	désactivé	<code>direct</code>

- 1- Tâches avec un pourcentage d'écriture variant de 0 à 100% par pas de 12.5 pour des accès aux données séquentiels, puis aléatoires.
- 2- Tâches avec une taille de requête variant de 1ko à 1Mo par pas de puissance de 2.
- 3- Tâche de requêtes en écriture avec 30% de requêtes de 4ko, 60% de 16ko et 10% de 64ko (`bssplit`).
- 4- Tâches avec 1, 2, 4, 6 et 8 processus tournant en parallèle.