

Rapport RES Labo 01

Nous voulons pour ce laboratoire réaliser une série de test (benchmark) pour nous aider à mieux comprendre le fonctionnement de la lecture et écriture sur une mémoire persistante. Le laboratoire nous propose un code qui réalise déjà un benchmark mais sans enregistrer les valeurs de celui-ci, de plus ce code n'est pas extensible à d'autres types de tests.

Le Benchmark en question testera différents aspects des performances d'input output de la mémoire persistante :

- Lecture et écriture (READ, WRITE)
- Taille des blocs (500, 50, 5, 0)
- Avec ou sans buffer

La taille des fichiers lu et écrits par le benchmark sera de 10MB.

Pour commencer voici la configuration sur laquelle les tests seront effectués :

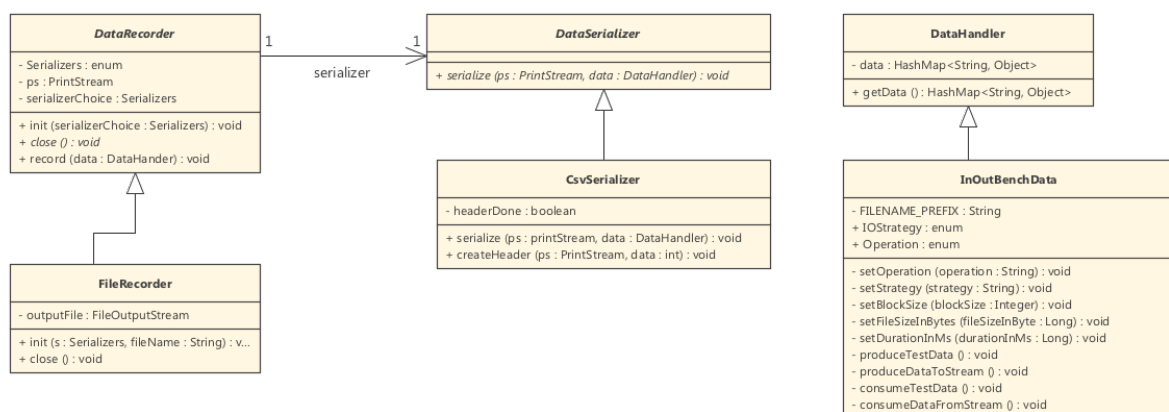
- MacBook pro retina 15'', early 2013
- Processeur : 2.4 GHz Intel Core i7
- Mémoire : 8 Go 1600 MHz DDR3
- Disque dur : SSD PCIE 512GB
- OS : OSX

Pour commencer j'ai voulu répondre à ces questions :

- What is we have 10, 20 strategies?
- What if we have 5 parameters for each strategy?
- How can we analyze the data more efficiently?

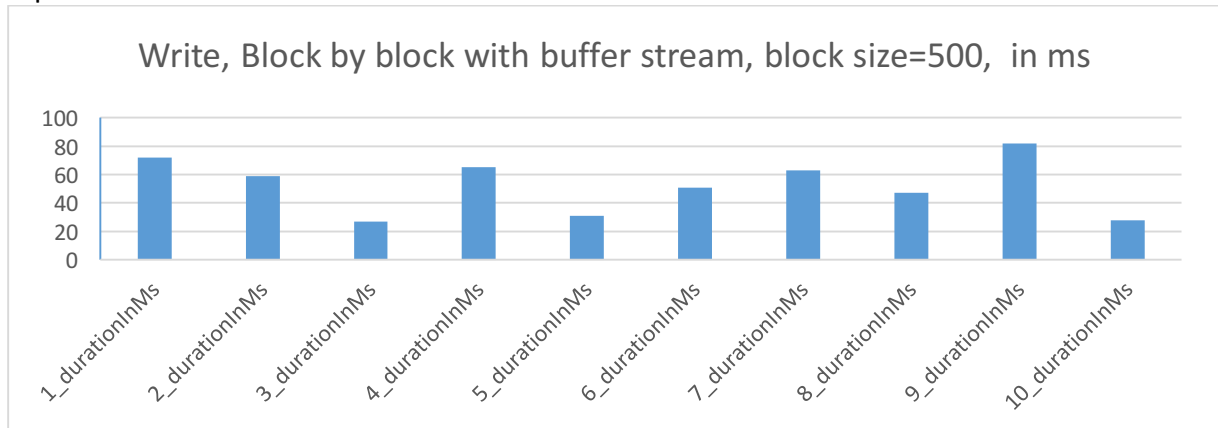
Pour répondre à toutes ces questions d'un coup je dirais qu'il faut séparer le travail de chaque fonctionnalité du programme de test en une entité qui s'occupe spécifiquement de celle-ci.

Dans notre cas voici la décomposition que j'ai choisie :



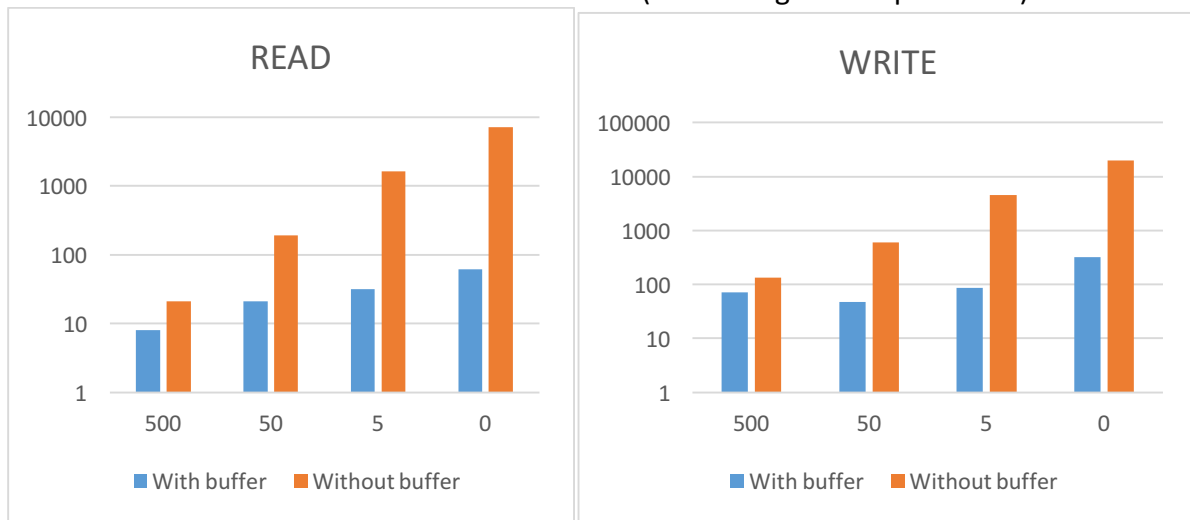
Les type recorder ont pour but d'enregistrer, plus précisément FileRecorder enregistre dans un fichier. Les serializer convertisse les données en séries. Les DataHandler contiennent les données et ici la spécification InOutBenchData permet de contenir les données spécifiques à ce labo.

Pour commencer l'analyse des données, j'ai commencé par faire le test 10 fois pour voir rapidement la « stabilité » des données :



Le rapport étant trop limité en pages pour en discuter, je dirais que, bien que les valeurs soient variables entre les tests de différentes séries, les résultats obtenus entre les différents tests de la même série sont trop différents pour que cela pose problème.

Voici les résultats obtenus dans une même série (échelle logarithmique en ms) :



Tout d'abord on peut constater que la lecture est plus rapide que l'écriture, ceci est principalement dû au disque dur lui-même. En lecture sans buffer, le temps de lecture est exponentiel et croît beaucoup plus vite qu'avec un buffer. Qui plus est avec buffer le temps d'attente est largement réduit. Pareil pour l'écriture.

Toutefois, il est intéressant de remarquer que lors de l'écriture avec buffer, le temps lors de taille de blocs 500 à 50 descend puis remonte pour 5. J'ai effectué 10 tests pour voir si l'événement se répétait, il s'est répété dans 40% des cas. Je pense que cela peut être relatif à la taille de « secteur » de mon disque dur.

Dans tous les cas, l'utilisation d'un buffer pour stocker les données en mémoire est un gain drastique de performance.