

Compte-rendu du TP C++ 4

Évaluation des performances

Arnaud Favier

Marc Gagné

Février 2016

Le plus grand facteur limitant la performance de notre programme est la lecture et écriture de fichiers (ainsi que la sortie et l'entrée standard). Pour limiter ceci, nous avons essayé plusieurs techniques, par exemple en lisant les commandes caractère par caractère et en adoptant un format de sauvegarde compact.

Pour évaluer la performance, nous avons créé le script `tests/benchmark.sh`, qui génère plusieurs très grands fichiers d'entrée de test de l'ordre du MB et mesure le temps d'exécution du programme pour trois opérations différentes :

- Création d'objets, sans sauvegarde : plusieurs centaines de milliers de commandes de création de segments et de rectangles sont effectuées, et le temps nécessaire pour leur création est mesuré.
- Création d'objets, avec sauvegarde : même opération que précédemment, mais en plus nous sauvegardons le modèle produit ; nous en déduisons une estimation du temps de sauvegarde par soustraction (pas entièrement fiable, car il peut y avoir des variations de la charge du système entre deux exécutions du programme).
- Chargement d'un fichier de sauvegarde : une simple commande de chargement est lancée avec pour paramètre le nom du fichier généré précédemment, et le temps nécessaire pour charger les figures en mémoire est mesuré...

Ces tests sont effectués plusieurs fois, avec des fichiers d'entrée différents, pour obtenir une moyenne de performance. Aussi, pour **5 tentatives**, avec création de **100 000 segments**, **100 000 rectangles**, **1 000 réunions de 100 rectangles et 100 segments**, nous avons obtenu les résultats suivants sur les machines de l'INSA :

| Tentative | Création | Création + sauvegarde | Sauvegarde | Chargement |
|----------------|--------------|-----------------------|--------------|--------------|
| Tentative 1 | 1.217 | 1.648 | 0.431 | 0.734 |
| Tentative 2 | 1.218 | 2.276 | 1.058 | 0.716 |
| Tentative 3 | 1.303 | 1.682 | 0.379 | 0.746 |
| Tentative 4 | 1.207 | 1.701 | 0.494 | 0.755 |
| Tentative 5 | 1.220 | 1.668 | 0.448 | 0.731 |
| Moyenne | 1.233 | 1.795 | 0.562 | 0.737 |

Nous voyons donc que même pour des fichiers de taille assez conséquente les temps de traitement demeurent acceptables pour l'utilisateur. Entre autres, nous voyons également que le temps de chargement d'un fichier de sauvegarde est beaucoup plus court que le temps nécessaire pour la création initiale ; ceci est en partie dû au format compact du fichier de sauvegarde et au fait que les réunions d'objets sont décrites explicitement, plutôt que sous forme de liste de noms d'objets à copier.