

Projet programmation répartie

Calcul parallèle

Partie 1 : **Préambule**

On peut effectuer du calcul en parallèle, en répartissant la tâche sur plusieurs ordinateurs. Par exemple, si nous devons calculer la somme de tous les entiers d'un grand tableau, nous pouvons le diviser en sous-tableaux, les envoyer à différentes machines pour effectuer les calculs, puis additionner les résultats partiels pour obtenir la somme totale. Cela permet d'accélérer le traitement global en répartissant la charge de calcul sur plusieurs machines.

Cette partie pourra être réalisée à l'aide d'une architecture en étoile, où un serveur connaît un ensemble de noeuds, et le serveur répartit la charge sur ses noeuds. Le serveur reçoit sur une certaine forme une partie du résultat par chaque processeur, et les assemble pour obtenir le résultat.

Si un des noeud ne répond pas, on peut demander à un noeud qui a finalisé son traitement de travailler la partie manquante. Le serveur ne retourne qu'une erreur s'il n'a aucune machine pouvant répondre à ses requêtes.

Partie 2 : **Le tracé de rayon (*raytracing*)**

1. Tester le programme en modifiant ses paramètres (sur la ligne de commande).

On peut mettre 3 paramètres : le nom du fichier, sa hauteur, sa longueur

2. Observer le temps de d'exécution en fonction de la taille de l'image calculée.
Vous pouvez faire une courbe (temps de calcul et tailles d'image).

Image de base (512x512): 1761 ms (Pc desktop de l'IUT, i5) et 2044 ms (Pc portable, i7 de 11e génération, sur batterie. Il est possible que branché, le traitement aurait été très proche du CPU du PC de l'IUT)

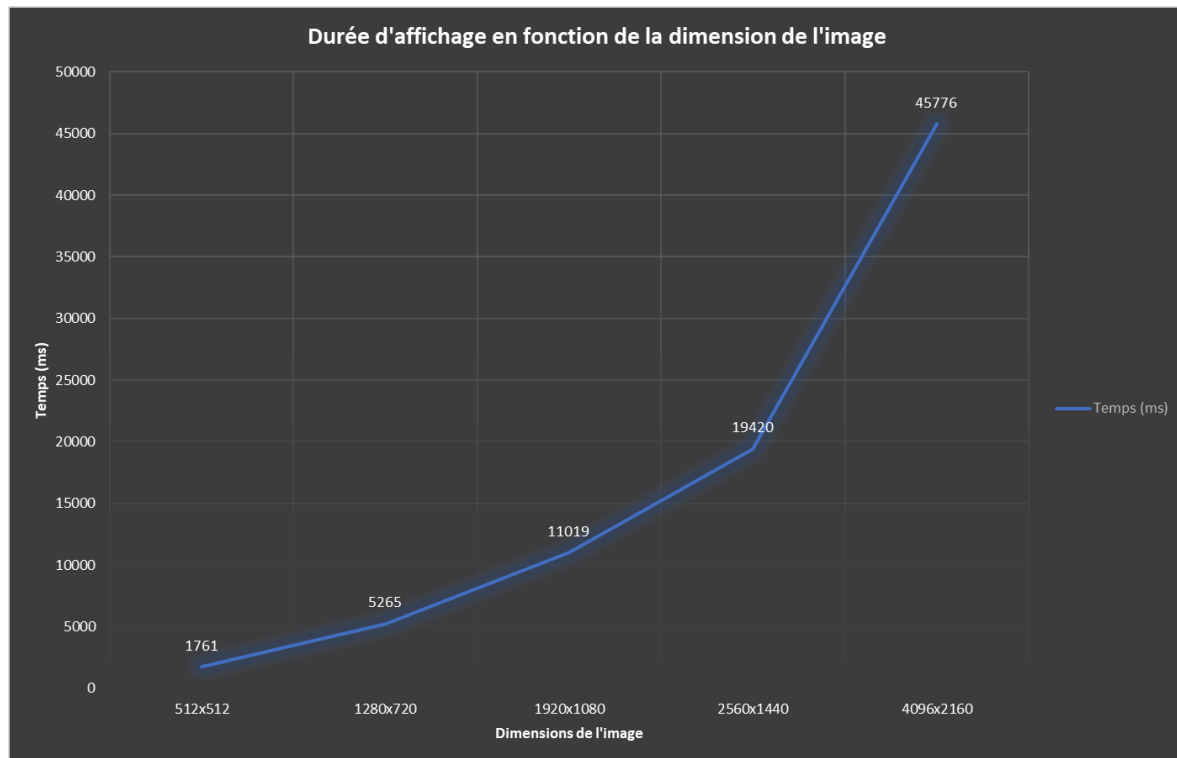
image en 1280x720 : 5265 ms (Pc de l'IUT)

image en 1920x1080 : 11019 ms (Pc de l'IUT, i5) et 7600 ms (Pc portable, i7 de 11e génération, branché)

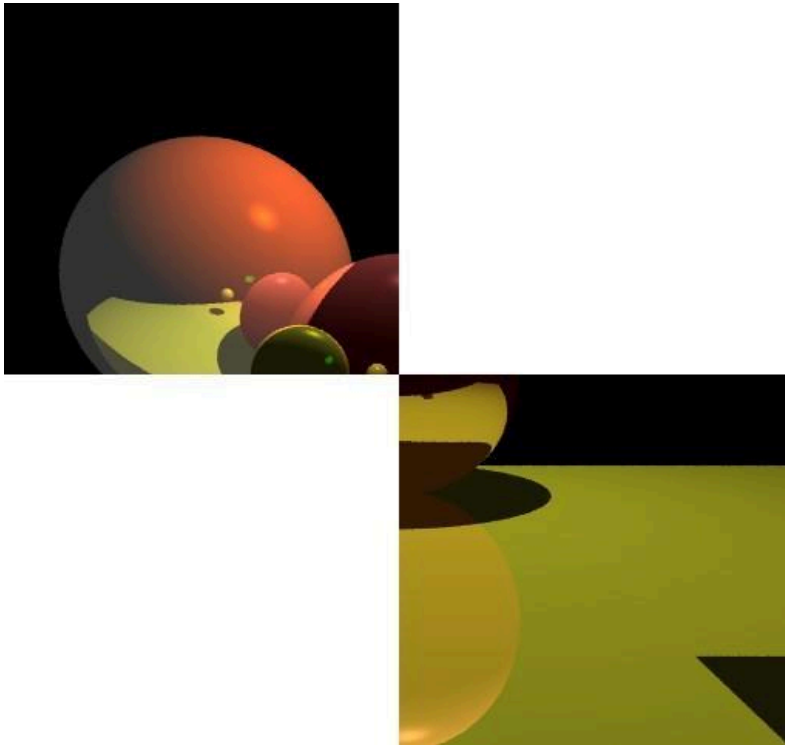
image en 2560x1440 : 19420 ms (Pc de l'IUT)

image en 4096x2160 : 45776 ms (Pc de l'IUT)

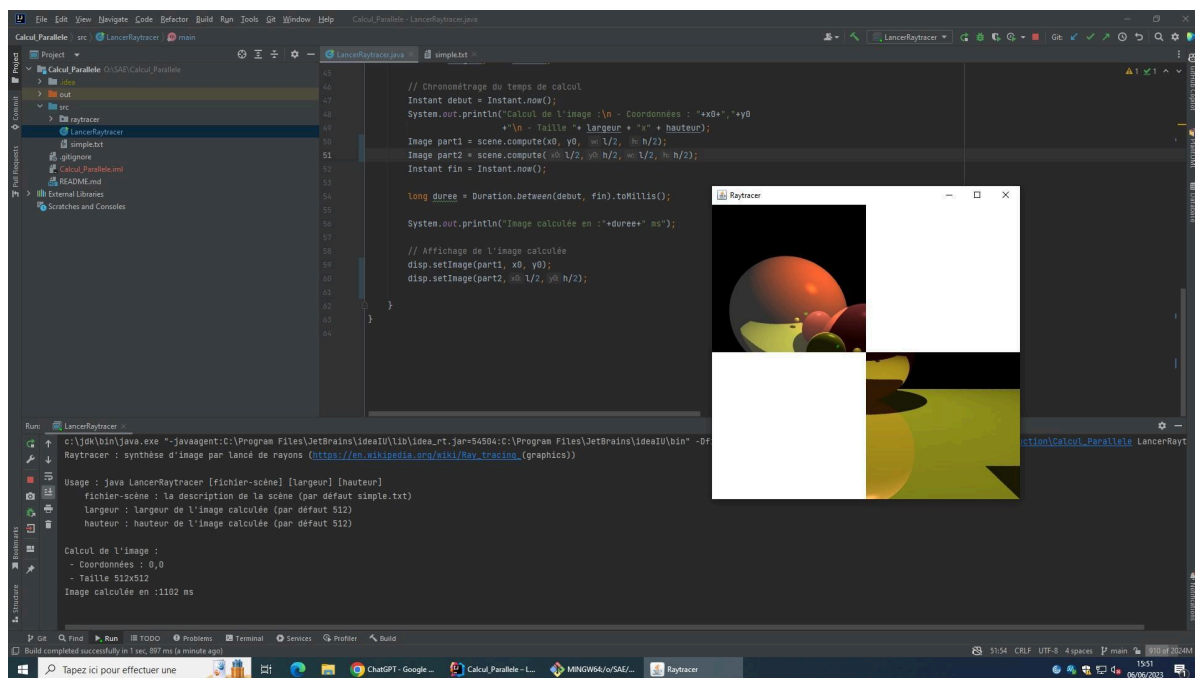
Image en 10000x10000 : 8 minutes et 26 secondes (Pc portable, i7 de 11e génération, branché)



3. Essayer de reproduire l'image suivante :

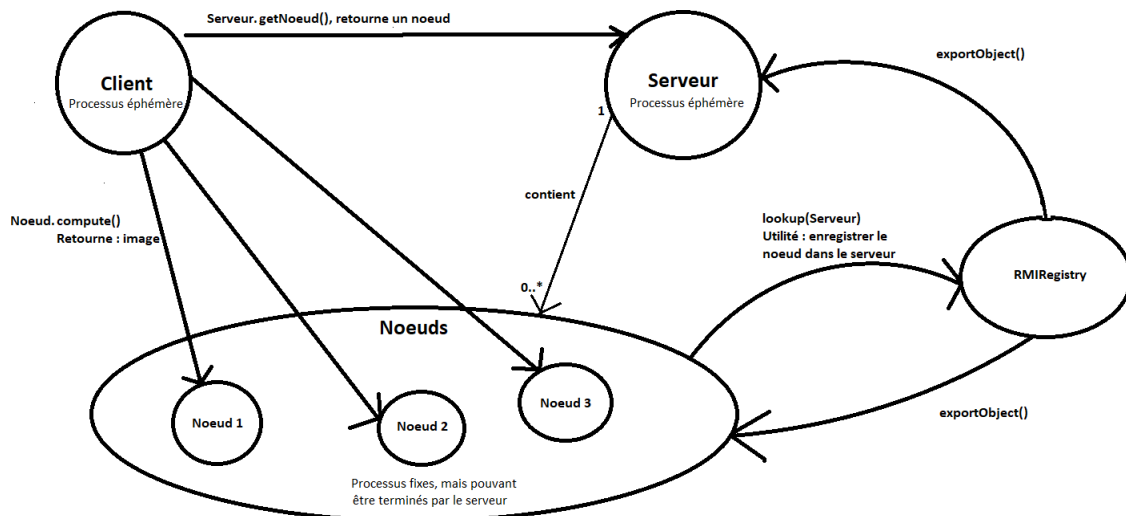


Nous avons :



Partie 3 : Accélérer les choses

- Faire un petit schéma de cette architecture en identifiant les choses suivantes :
 1. Le/les processus fixes (ceux qui écoutent sur un port choisi) et les processus éphémères ? (ceux qui entrent et sortent à leur guise) ?
 2. Les types de données échangées entre les processus



- Si on veut que les calculs se fassent en parallèle, que faut-il faire ?

Il faut exécuter le programme du ray-tracing en ne transférant que la méthode `compute` à exécuter sur le processus de calcul distant, soit le nœud, et le processus de calcul distant renvoie l'image, que le client affiche.

Il faut donc un serveur, qui enregistre des nœuds disponibles pour traitement. Le client appellera le serveur, qui renvoie la liste de nœuds. Ces nœuds auront une charge répartie de travail.

On peut appeler la classe `Thread` pour que chaque thread soit associé à un nœud, pour qu'il suive son traitement.

https://github.com/SkyPurl/calcul_parallele