

# Projet Programmation répartie

Scheffer Benjamin

Unternehrr David

Lien vers le dépôt git : [https://github.com/BenjaminScheffer/projet\\_SETI/tree/main/service](https://github.com/BenjaminScheffer/projet_SETI/tree/main/service)

## Question :

En utilisant votre meilleur outil, votre imagination, décrivez et illustrez comment cela pourrait être réalisé, sans rentrer dans les détails JAVA, que vous n'allez pas tarder à mettre en œuvre.

Selon nous, pour mettre en place cette outil, il faudra :

- Un Service Central :

Le Service Central aura pour rôle d'enregistrer les machines qui feront les calculs mais aussi de récupérer et répartir les calculs que les clients souhaitent réaliser sur les différentes machines dont il dispose. Donc il disposera de deux méthodes : enregistrerClient(ServiceClient s) et faireCalcul(calcul). Il disposera d'un annuaire.

- Des machines de calcul :

Ces machines s'enregistreront au près du service centrale en fournissant leur service dans la même foulée. Elles doivent donc lancer un service en locale mais sans annuaire.

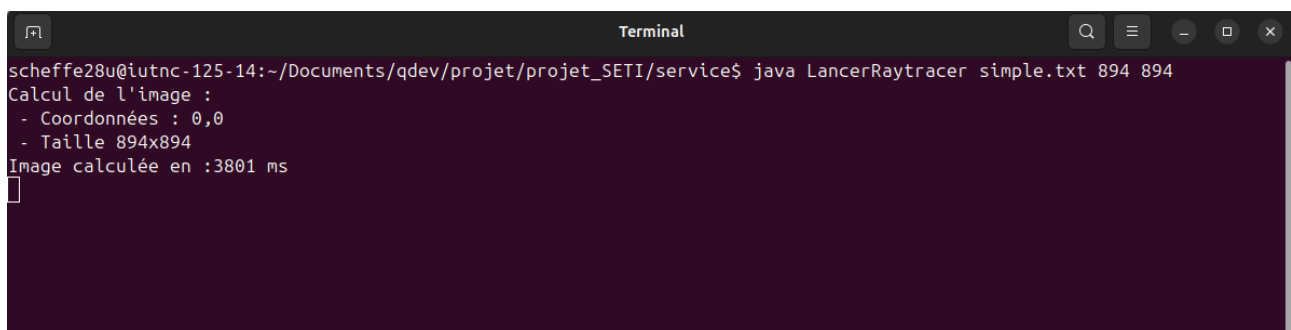
- Des clients :

Les clients enverront les calculs qu'ils souhaitent réalisés au service central grace a l'annuaire.

## Questions

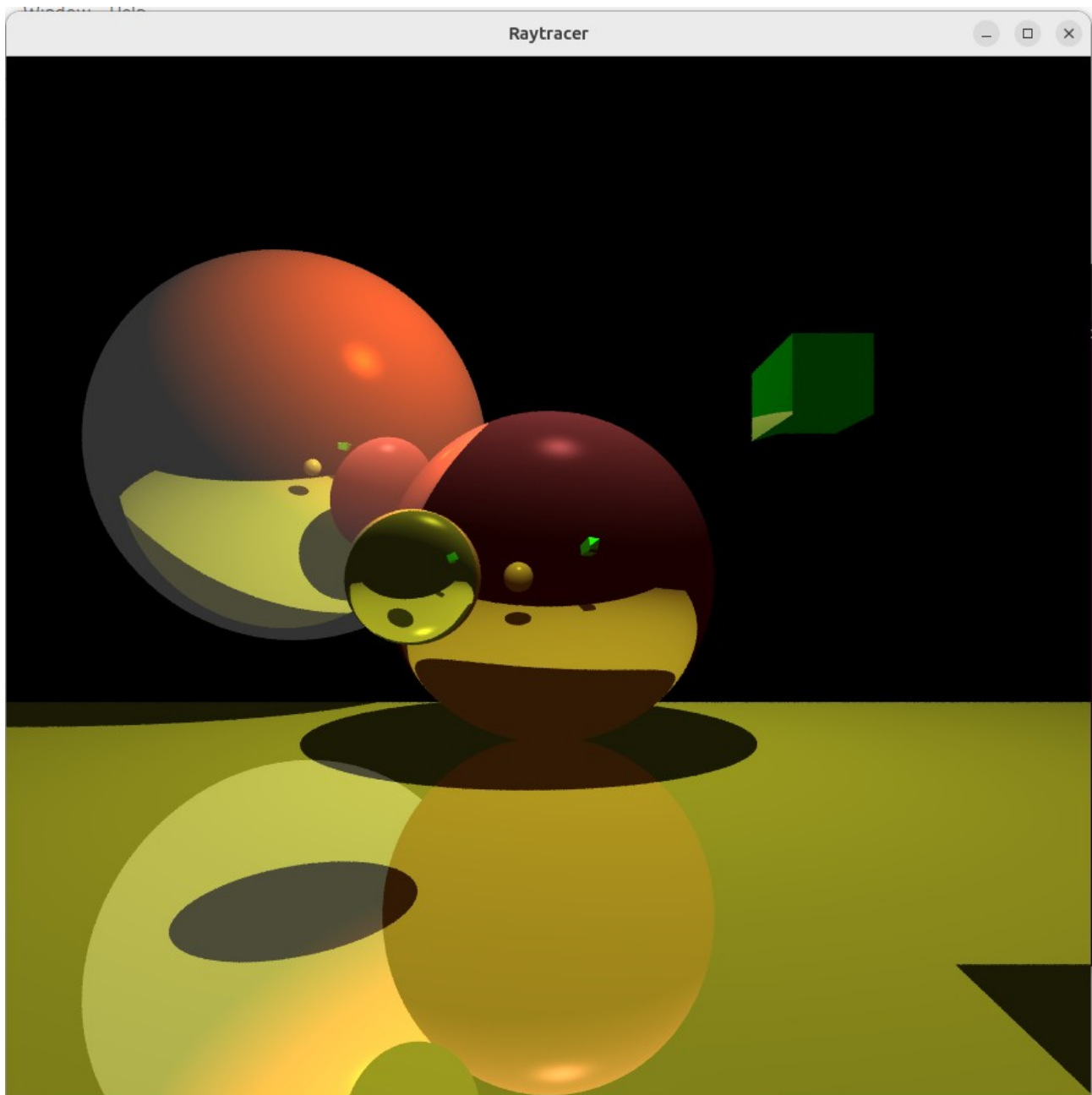
Tester le programme en modifiant ses paramètres (sur la ligne de commande).

Image du terminal :



```
Terminal
scheffe28u@iutnc-125-14:~/Documents/qdev/projet/projet_SETI/service$ java LancerRaytracer simple.txt 894 894
Calcul de l'image :
- Coordonnées : 0,0
- Taille 894x894
Image calculée en :3801 ms
```

Image du résultat de l'exécution :



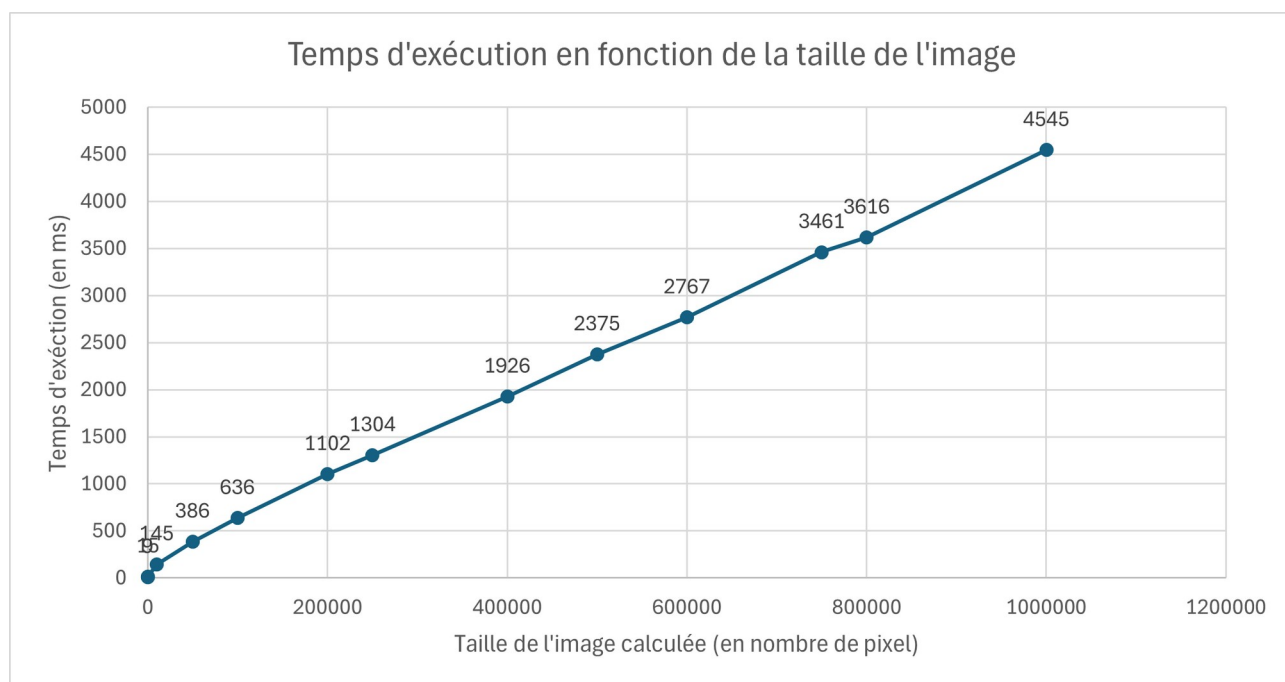
Observer le temps de d'exécution en fonction de la taille de l'image calculée. Vous pouvez faire une courbe (temps de calcul et tailles d'image).

Voici les données que nous avons tester :

Nombre de Pixels	Temps en ms	Largeur x Hauteur
1	9	1 (Image de 1x1)
100	15	10 (Image de 10x10)
10000	145	100

50000	286	223
100000	636	316
200000	1102	447
250000	1304	500
400000	1926	632
500000	2375	707
600000	2767	774
750000	3461	866
800000	3616	894
1000000	4545	1000

Image de la courbe :



En ne modifiant **que** le fichier *LancerRaytracer.java*, reproduire l'image suivante :

Voici les modifications apporté au fichier LancerRaytracer :

```
int x0 = 0, y0 = 0;
    int l = largeur, h = hauteur;

    // Chronométrage du temps de calcul
    Instant debut = Instant.now();
    System.out.println("Calcul de l'image :\n - Coordonnées : "+x0+", "+y0
```

```
+"\n - Taille "+ largeur + "x" + hauteur);
```

```
//Boucle
```

```
Image image = scene.compute(x0, y0, l/2, h/2); // Le l/2 et h/2 permet de créer une image de la longueur et hauteur divisé par 2. x0 et y0 est le point d'origine de l'image.
```

```
Image image2 = scene.compute(l/2, h/2, l/2, h/2); // Deuxième image qui débute a l/2 et h/2 avec la meme taille que la premiere image.
```

```
//Fin de boucle
```

```
//Caculer le temps
```

```
Instant fin = Instant.now();
```

```
long duree = Duration.between(debut, fin).toMillis();
```

```
System.out.println("Image calculée en :"+duree+" ms");
```

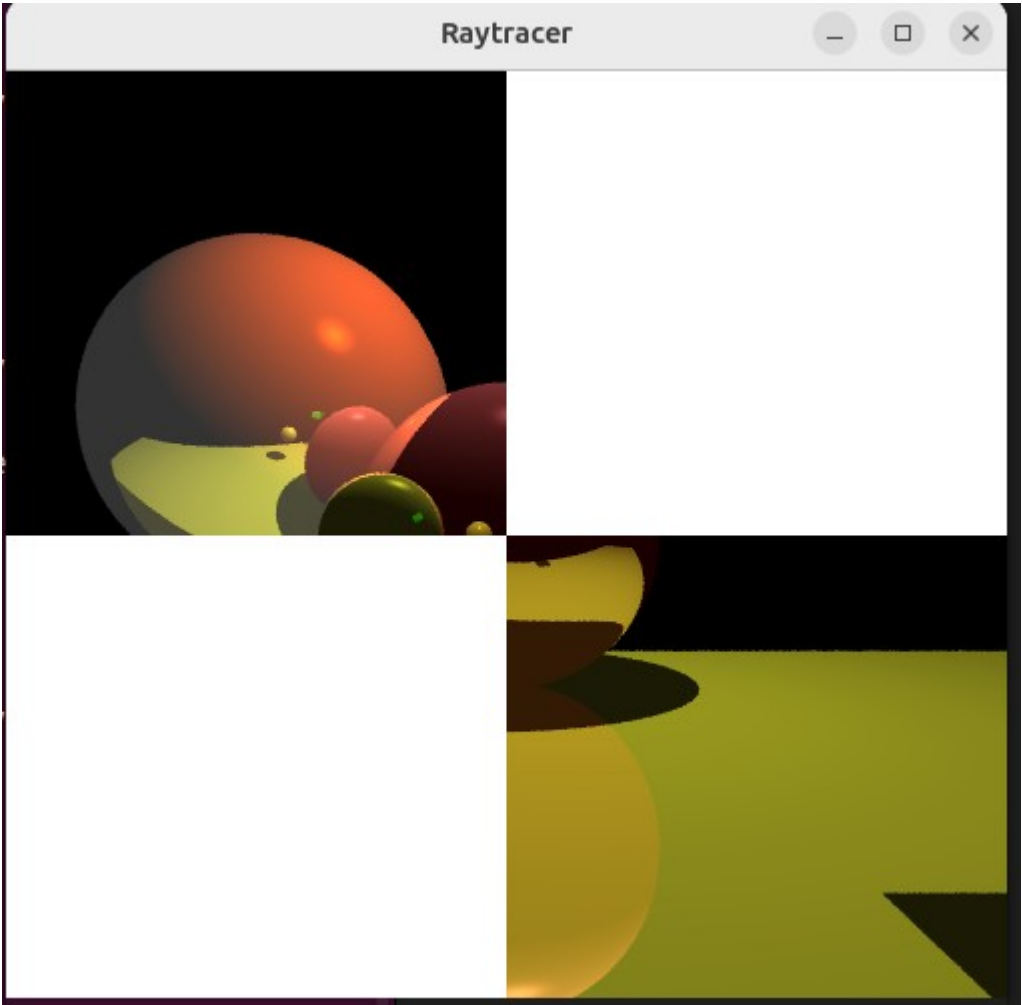
```
// Affichage de l'image calculée
```

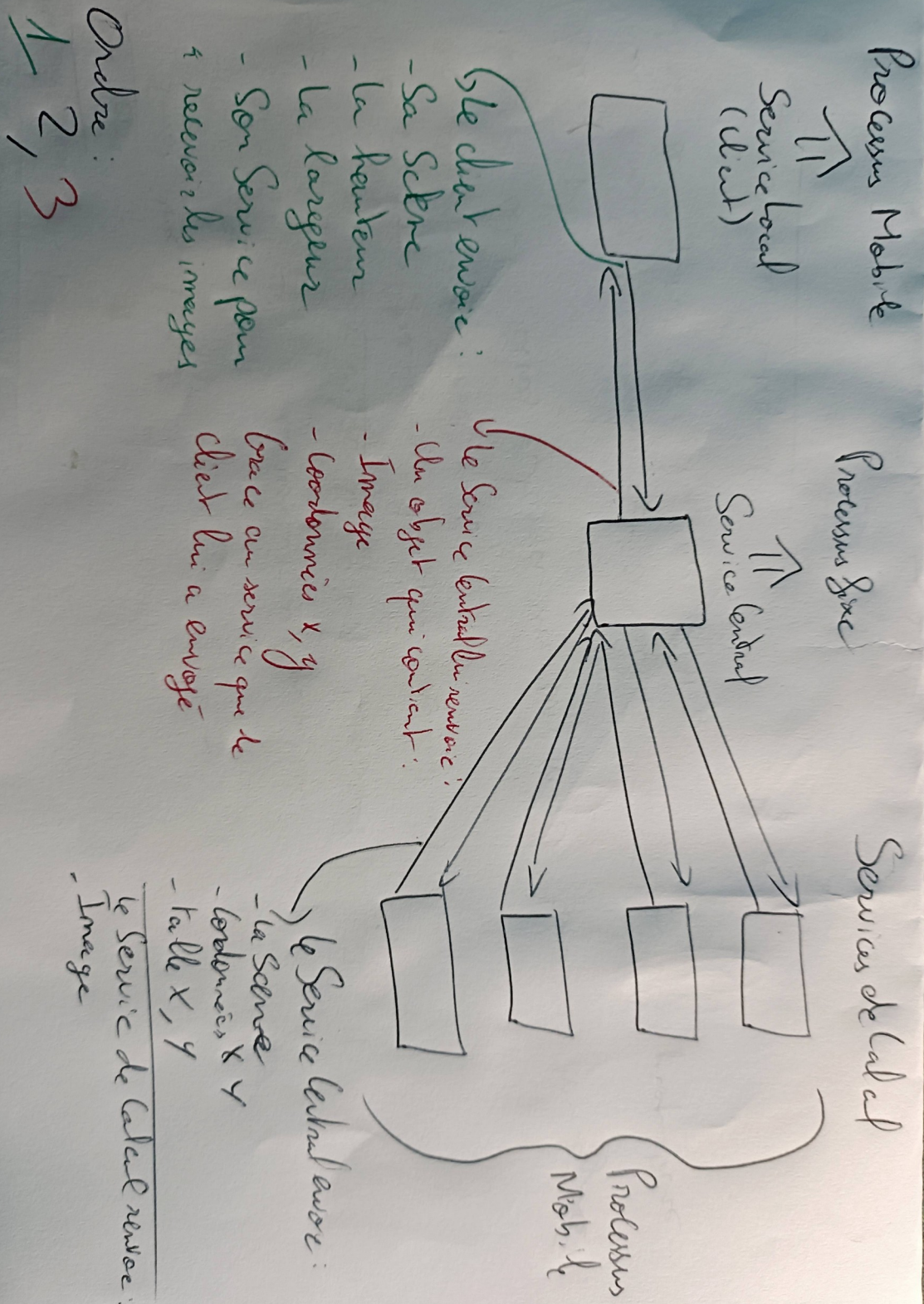
```
disp.setImage(image, x0, y0);
```

```
disp.setImage(image2, l/2, h/2); // Ajoute l'image à la fenetre
```

Par exemple si on lance le programme avec une taille de scene de 300 300. La premiere image couvrira la zone de 0 0 à 150 150 et la deuxième image de 150 150 à 300 300. Donc le coin supérieur gauche et inférieur droit seront remplis.

Image du résultat :





Si on veut que les calculs se fassent en parallèle il faut créer des threads pour pouvoir réaliser plusieurs calculs en simultané.

Lien vers la vidéo :

<https://www.youtube.com/watch?v=xtDQa9Wk4mg>