

M2 Génie Logiciel

Rapport TD5 : InputFormat

Exercice 1&2 : Point2DWritable & RandomPointInputFormat

Nous sommes finalement parvenu à relier les différents composants nécessaires pour adapter les entrées d'une application MapReduce à cet exercice, les nombreuses erreurs de compréhension ayant été corrigées au fil des tests exécutés lors de l'exercice 3.

Exercice 3 : Test du RandomPointInputFormat

Le seul point que nous ne nous expliquons pas dans notre rendu est que nous ne parvenons pas à adapter la taille de nos splits à l'argument fourni par l'utilisateur.

En effet, bien que cet argument soit récupéré dans notre

RandomPointInputFormat.getSplits, puis transmet à chaque split au sein de cette méthode via l'argument de leur constructeur, les **FakeInputSplit** prennent toujours la taille que nous avons mises dans le constructeur sans arguments. Nous ne comprenons pas à quel moment ce constructeur est appelé, et pourquoi les **FakeInputSplit** que nous avons instanciés sont inexploités (selon toute vraisemblance).

Exercice 4 : Calcul de PI

Nous nous sommes vu obligé de passer par la classe **BigDecimal** pour calculer PI, car le programme nous retournait sinon une valeur ronde et indépendante de nos calculs (0.0, 4.0 et 6.0). Le calcul n'a rien de compliqué :

1. lors de la fonction reduce, si le point appartient au cercle, alors on incrémente un compteur pour connaître le nombre de points dans le cercle ;
2. lors du cleanup, on fait le rapport du nombre de point dans le cercle sur le nombre total de point, et on multiplie le résultat par 4.

Pour des raisons d'efficacité, nous avons utilisé des Combiners.

Cependant, ceux-ci sont contraint d'avoir pour output le même ensemble <clé,valeur> que les Mappers, nous avons donc dû nous adapter et nos Combiners retournent des <LongWritable, Point2DWritable> qui sont concrètement :

- dans point2D.X : le nombre de points générés localement;
- dans point2D.Y : le nombre de points générés localement appartenant au cercle.

Déroulement dans les machines traitant le mapping

