

# Scylla réécrit Cassandra en C++ et décuple ses performances

La base de données ScyllaDB fait tomber deux idées fausses très couramment acceptées dans le monde IT. La première est que le code Java, exécuté par un compilateur *just-in-time*, est presque aussi rapide que du code natif. La seconde est que sur des logiciels répartis sur un vaste ensemble de machines, les performances du *software* ne sont guère importantes, l'architecture matérielle étant un facteur limitant plus lourd.

ScyllaDB élimine ces deux préjugés. Réécriture en C++ de la base de données NoSQL Cassandra – programmée en Java –, ce projet propose des performances **jusqu'à dix fois plus rapides** que l'original, [rapporte Slashdot](#). L'éditeur Scylla, à l'origine de ce projet, affirme que chaque nœud de l'architecture serveur sera capable de gérer jusqu'à **1 million de transactions par seconde**.

Preuve s'il en est que la performance brute de la JVM ne fait pas tout. Sa gourmandise en ressources mémoire et les variations de vitesse au fil du temps ont un impact fort sur les performances réelles des applications Java.

## 99 % des requêtes en moins de 1 ms

Mais plus que tout, c'est en termes de consistance du taux de latence que ScyllaDB affiche sa supériorité. Dans 99 % des cas, le taux de latence **ne dépassera pas les 1 ms**.

Cassandra ne peut arriver à un tel résultat, le ramasse-miettes de Java pouvant impacter ses performances à tout moment, et ce de façon **quasi impossible à prédire**. La prédictibilité du temps nécessaire à la réalisation d'une tâche n'a jamais été le point fort de Java. Un défaut confirmé ici.

Notez qu'au cœur de ScyllaDB se trouve l'utilisation du **framework C++ Seastar**, qui permet de faciliter la mise au point d'applications serveur à hautes performances. L'équipe de Scylla a participé à la mise au point de cette offre, mais également à d'autres projets d'envergure, comme KVM et OSv.

### À lire aussi, notre dossier programmation Raspberry Pi :

[Raspberry Pi et développement : Apprendre à programmer \(épisode 1\)](#)

[Raspberry Pi et développement : Java SE, first class citizen \(épisode 2\)](#)

[Raspberry Pi et développement : C/C++ à toutes les sauces \(épisode 3\)](#)

[Raspberry Pi et développement : du calcul au Big Data \(épisode 4\)](#)

[Raspberry Pi et développement : .NET, avec ou sans Windows \(épisode 5\)](#)

[Raspberry Pi et développement : un RAD nommé Xojo \(épisode 6\)](#)

**Crédit photo : © Martin Bech – Fotolia.com**