

# 5 projets open source made in Twitter... en attendant l'algo ?

Publier l'algorithme de Twitter ? Elon Musk en a fait un de ses chevaux de bataille. L'idée n'est pas du goût de certains employés, qui l'ont fait savoir. Entre autres par l'intermédiaire du GitHub de l'entreprise. Y est apparu, la semaine dernière, un [dépôt public « the-algorithm »](#). Avec rien dedans. Ce que d'aucuns ont interprété comme un troll. Il n'est resté en place que quelques heures, collectant toutefois quelque 5000 étoiles.

En attendant une éventuelle publication d'algorithme(s), le GitHub est déjà bien fourni. Le [site vitrine](#) qui l'accompagne affiche une centaine de projets au compteur.

## Heron : le Spark de Twitter

Parmi eux, il y a [Heron](#). Il s'agit d'un moteur générique de traitement de flux distribué écrit en Java/C++/Python. Publié en 2016, il était passé deux ans plus tard sous l'aile de la Fondation Apache. Officiellement encore au stade d'incubation (dernière version : mai 2021), il s'architecture en conteneurs.

Chez Twitter, Heron a remplacé un autre moteur maison, lui aussi passé dans le domaine de l'open source : Apache Storm. Il est rétrocompatible avec lui (API et modèle de données). Mais y ajoute notamment l'isolation des tâches sous forme de processus, l'exploitation de ressources à la demande et une refonte architecturale qui améliore autant les débits que la latence.

Les entités de traitement (des graphes orientés acycliques, ici appelés topologies) sont écrites en Java ou en Python.

## Rezolus : pour une télémétrie plus fine

Sous licence Apache 2.0 comme Heron, [Rezolus](#) avait basculé en *open source* en 2019. Il s'agit d'un agent de télémétrie pour les systèmes Linux – avec une prise en charge limitée de macOS. Twitter l'a développé pour obtenir des mesures plus fines qu'avec son outillage précédent, qui échantillonnait à la minute.

Rezolus permet quant à lui de paramétrer la fréquence d'échantillonnage jusqu'à une précision de l'ordre de 200 ms. Ses sources : procfs, sysfs, le sous-système perf\_events et l'interface BPF. Il peut aussi fonctionner comme proxy pour collecter des métriques (voir [ici](#) celles prises en charge) au niveau des applications.

## Scalding : Cascading en Scala

À l'image de [Hive](#) et de [Pig](#), ce *framework* Hadoop simplifie l'écriture de fonctions *map* et *reduce*. Il exploite le langage Scala. Avec, sous le capot, la bibliothèque [Cascading](#) (implémentée dans

d'autres langages : Lingual pour SQL, Cascalog pour Clojure...).



Scalding est ouvert depuis 2013, sous licence Apache 2.0. Twitter l'utilise entre autres pour le ciblage publicitaire et la prédiction de clics. LinkedIn s'en sert sur sa partie *emailing*. Spotify, pour les recommandations musicales. Barclays, pour l'ETL.

## TIP (Twitter Image Pipeline) : le pendant iOS de Fresco

Le *framework* [TIP](#) (licence Apache 2.0 ; dernière *release* : octobre 2020) s'intègre dans les applications iOS. Comme [Fresco](#) (de Facebook) que Twitter utilise sur Android, il gère le rendu des images, à l'appui de trois niveaux de cache. Il gère notamment l'intégration de codecs, le redimensionnement, la reprise des téléchargements et le jpeg progressif.

## Twemoji : la collection Twitter

À travers [ce projet](#), qui compte près de 15 000 étoiles et 1600 *forks*, Twitter met à disposition sa propre collection d'emojis Unicode, en PNG et SVG (licence CC-BY 4.0). On les retrouve notamment sur Discord, Roblox et des claviers tiers pour Android. Divers projets se sont développés pour permettre d'implémenter Twemoji en Ruby, en .NET, au travers des classes CSS ou comme police par défaut sur Linux/OS X.

*Illustration principale* © GlebStock – Shutterstock