

Silicon Valley Tour – Containers et Big Data restent des technologies incontournables

En dehors des aspects matériels de notre périple en Silicon Valley, nous avons aussi rencontré CoreOS un concurrent de Docker et Jut pour parler de sa plateforme Big Data hybride.

Un système d'information à la Google pour tous avec CoreOS

Les lecteurs de Silicon.fr suivent depuis longtemps [les évolutions de CoreOS](#), que la rédaction a très vite identifié comme un utilisateur puis un concurrent des technologies de container de Docker (les petits Français installés aux États-Unis).

Créée en mai 2013 par Alex Polvi, son CEO, et Brandon Philips dans le rôle de directeur technique, CoreOS a déjà levé 20 millions de dollars en quatre tours de table auprès d'Accel Partners, Fuel Capital, Kleiner Perkins Caufield & Byers, Sequioa Capital, Andreessen Horowitz, Y Combinator et aussi Google Ventures. La société emploie aujourd'hui 40 personnes venant de Twitter, Google, Suse ou encore Amazon ou Rackspace

« Les géants d'Internet comme Google ou Amazon ou le Nasdaq ont conçu des systèmes d'exploitation très similaires pour répondre à leurs propres besoins, » explique Alex Polvi. « Avec le projet open source CoreOS, nous mettons l'infrastructure de Google à la portée de tous. »

Pour résumer, CoreOS est un système d'exploitation super-allégé basé sur un noyau Linux permettant de démarrer très vite en consommant un minimum de ressources serveur. Il est destiné à des environnements en cluster favorisant l'évolutivité, l'automatisation des tâches, mais aussi le déploiement d'applications et la sécurité. Pour l'aspect applicatif, CoreOS utilise la technologie des containers permettant d'embarquer tout le nécessaire à une application (environnement, librairies, etc.) afin qu'elle puisse être déployée simplement dans divers environnements.

Plusieurs containers peuvent être placés dans une instance CoreOS. Et trois éléments interagissent pour assurer le fonctionnement en cluster de manière centralisée. Sur chaque serveur (et pas dans le container), etcd permet de partager des données en mode haute disponibilité (distribué et résilient) entre les instances CoreOS. Écriture et lecture des données sont réalisées via une API Rest. Le module fleet contrôle au niveau cluster les instances de CoreOS pour que tous les serveurs soient considérés comme un cluster unique. Pour cela, il interagit avec systemd, le cœur des instances CoreOS (le daemon permettant de démarrer ou arrêter les services d'une instance).

Les premières versions de CoreOS et etcd ont été publiées en août 2013, mais il a fallu attendre juillet 2014 pour une première mouture stable de CoreOS. La société en profite alors pour proposer aussi une version commerciale intégrant Managed Linux. En août 2014, CoreOS rachète Quay.io et lance son second produit commercial Enterprise Registry depuis son nouveau bureau de New-York.

Après avoir embarqué dans sa solution les containers de Docker, CoreOS annonce [son propre container Rocket \(rkt\)](#) en décembre 2014 afin de revenir à la simplicité du container face à Docker

devenant une plateforme. « Le principe du container consiste à disposer d'un élément standard qui peut être transporté partout de la même manière, et sans se poser de question sur la compatibilité. Ce qui implique aussi de pouvoir choisir le camion ou le train sur lequel est transporté le container!» rappelle Alex Polvi.

En avril 2015, CoreOS lance sa solution commerciale baptisée [Tectonic](#), tout en conservant évidemment les produits open source CoreOS, etcd ou encore son container rkt. Tectonic associe CoreOS et la solution de gestion de containers de Google Kubernetes. Au passage, la société reçoit un financement de 12 millions de dollars de Google Ventures.

« Nous assurons le support commercial sur toute la pile logicielle, et proposons des outils d'administration et de supervision d'entreprise,» souligne Alex Polvi. «Les projets open source CoreOS continuent et sont animés par une communauté très active, avec des meetup à travers le monde, sans oublier les CoreOS Fest où se déroulent de très nombreux échanges et partages de bonnes pratiques.»

La rédaction de silicon.fr suit cette tendance des containers et les évolutions des deux organisations visant à standardiser les formats de containers : L' [Open container project](#) et l'Application Container Spec (appc), promu par CoreOS qui a apporté son support à OCP.

La plateforme Jut remplace l'analytique au cœur de l'entreprise

Après avoir créé Riverbed (optimization Wan) entrée au Nasdaq en 2006, et FastForward Networks (streaming multicast en temps réel) rachetée par Inktomi en 2000 (rachetée par Yahoo en 2003), Steve McCanne lance JUT en 2013. Année au cours de laquelle Jut lève 23 millions de dollars (Accel, Lightspeed, Wing), et s'est installée dans de beaux locaux en plein centre de San Francisco. Qu'est-ce que Jut? Une plateforme centrale pour capturer et analyser toutes données de l'entreprise.

« Toutes les entreprises deviennent des "software companies". Donc, elles deviennent aussi des "data companies". C'est pourquoi la santé d'une organisation peut souvent s'évaluer d'après l'analyse de son système d'information logiciel,» affirme Steve McCanne, fondateur et dirigeant de Jut. «Toutefois, une compréhension holistique de la santé de "son logiciel" s'avère complexe à cause des multiples silos dont est composé son système d'information.»

Pour pouvoir analyser ses données et prendre des décisions, l'entreprise choisit souvent divers logiciels pour chaque besoin et génère des silos rendant les corrélations difficiles ou impossibles. Ou alors, elle cherche à construire sa propre plateforme analytique (Hadoop, Spark, Cassandra et des développements de bas niveau avec Java, Scala...), et alors elle est coincée par ses développements et le moindre changement nécessite de nombreux développements.

« Nous avons conçu JUT comme une plateforme de données holistique proposant une vue unique pour les données,» explique le CEO. «Une solution qui offre des possibilités infinies d'analyses pour tous dans l'organisation afin de répondre à toutes les questions.»

Présentée comme une plateforme analytique des flux de données (dataflow analytics platform), Jut fournit un traitement In-Memory en temps réel avec manipulation des données et des axes de données. La solution combine donc In-memory et disque de données sur la même plateforme.

Le moteur Jut Data Engine utilise de connecteurs ou des API pour récupérer tout flux de données sous toutes les formes (historiques, mesures, événements, streaming temps réel...). Le moteur

associe plusieurs technologies comme la base de données distribuée NoSQL Cassandra ou la technologie ElasticSearch.

Pour simplifier le développement et donc l'accès à ces données, le langage Juttle de type scripting aux airs de SQL évolué permet de réaliser des requêtes complexes en quelques instructions simples. Simplifiant ainsi le développement, la compréhension et la maintenance des applications.

Le tout est accompagné d'une technologie utilisant des projets open source Big Data optimisés pour chaque type de données quelles soient "historiques" ou en temps réel.

« Nous sommes une sorte de mélange de Google DataFlow et de Tableau, » résume le Steve McCanne.

Autre particularité, l'architecture volontairement mixte de la solution. Ainsi, le moteur s'installe sur site près des données de l'entreprise, tandis que les applications s'exécutent sur le cloud de Jut.

« Ainsi, l'entreprise dispose toujours de la dernière mise à jour de l'application, sans avoir à installer ou déployer quoi que ce soit, » assure l'éditeur.

Lors de la démonstration, la solution semble plutôt destinée aux devops et développeurs, qui bien entendu doivent interagir avec les métiers pour répondre à leurs besoins. Effectivement, deux ou trois lignes très brèves donnent rapidement des résultats spectaculaires avec de nombreuses visualisations possible.

Très simplement, le développeur peut insérer une instruction dans son code pour modifier l'interface, proposer une liste déroulante de paramètres ou de visualisation. Et le résultat s'affiche instantanément. Il suffit ensuite de ne présenter à l'utilisateur final que l'écran sans le script et tout semble à sa portée.

La politique tarifaire ne semble pas encore totalement arrêtée, mais la solution semble opérationnelle. Elle est d'ailleurs en open beta sur <https://app.jut.io> (plus d'informations [ici](#)).

A lire aussi :

[Silicon Valley Tour : Un stockage toujours plus logiciel et objet](#)

[Silicon Valley Tour : 100% ou hybride, le flash devient incontournable](#)

[Silicon Valley Tour : Le Software Defined Network toujours hyperactif](#)