

Conteneurs : un format unique pour réconcilier Docker et CoreOS

Eviter l'éclatement de la technologie des conteneurs en de multiples chapelles incompatibles entre elles. Telle est la vocation de l'**Open Container Project**, un consortium réunissant une impressionnante liste de grands noms du secteur, dont Amazon Web Services, Cisco, EMC, Fujitsu, Google, HP, Huawei, IBM, Intel, Microsoft, Red Hat ou encore VMware. Bref, la quasi-totalité des entreprises qui comptent dans le secteur réunies, **sous l'égide de la Linux Foundation**, pour voir Docker, la start-up par qui la révolution des conteneurs est arrivée, se réconcilier avec CoreOS, éditeur de l'OS éponyme qui a lancé récemment une technologie concurrente (Rocket) bâtie sur un format différent.

[\[Lire notre dossier : Docker, déjà bon pour le service\]](#)

« La mission d'OCP consiste à permettre aux utilisateurs et aux entreprises de continuer à innover et à développer des solutions à base de conteneurs, avec l'assurance que leurs efforts de développement passés seront protégés et à l'abri de la fragmentation de l'industrie », écrit le nouveau consortium dans un communiqué. Pour ce faire, OCP pourra compter sur l'appui de **Docker**, qui a prévu de **donner le code de son format de conteneurs** et de son runtime ainsi que les spécifications associées, ainsi que sur celui de CoreOS. L'initiative Application Container Spec (appc), le format concurrent de Docker que promeut en premier lieu CoreOS, a également apporté son support à OCP. Le consortium entend développer un standard commun, qui devra être rétro-compatible avec Docker, indépendant de tout client et outil d'orchestration ainsi que portable sur de multiples plates-formes (Linux, Windows, Solaris, architectures Power et x86). Le brouillon de ce futur format est attendu sous trois mois.

« Se concentrer sur l'innovation qui compte »

Dans les faits, OCP apparaît plutôt comme **une initiative de Docker** pour éviter la fragmentation du marché. En plaçant la destinée de son format entre les mains d'un consortium ouvert, piloté par une institution respectée dans l'Open Source comme l'est la Linux Foundation, la start-up fondée par des anciens de l'Epitech espère ramener CoreOS et d'autres sur une voie de développement commune.

« Avec le Open Container Project, Docker s'assure que la fragmentation ne va pas détruire la promesse amenée par les conteneurs », explique d'ailleurs Jim Zemlin, le directeur exécutif de la Linux Foundation. La start-up continuera à maintenir le client Docker, les outils de sa plate-forme, ainsi que les fonctions d'orchestration qu'elle a construite au-dessus des technologies confiées au consortium. « Cet effort va permettre à l'écosystème dans son entier de se focaliser sur l'innovation dans les couches technologiques qui comptent, plutôt que de perdre du temps dans une guerre de formats de bas niveau », analyse Ben Golub, le Pdg de Docker.

[\[Lire notre dossier : Docker, déjà bon pour le service\]](#)

Côté appc, OCP précise dans une [page](#) de questions/réponses que les personnes à l'origine de cette initiative concurrente du format Docker rejoignent les experts techniques du consortium. « *Notre intention est de travailler sur un format commun compatible avec les formats de conteneurs existant ainsi que de préparer une spécification future combinant les meilleurs éléments de toutes les initiatives autour de la technologie de conteneurs* », poursuit le consortium. L'équipe pilotant l'évolution technique d'OCP regroupera ainsi les ingénieurs en charge du projet libcontainer de Docker (développeurs indépendants ainsi que spécialistes issus de Docker, Red Hat et Google) ainsi que deux membres de appc.

A ce jour, **Docker est le standard de facto** des déploiements en entreprises. Selon [une récente étude](#), 92 % des organisations ayant déployé des conteneurs ont eu recours, au moins pour une partie des déploiements, à la technologie de la start-up. Le Rocket de CoreOS est lui testé ou déployé par un peu plus de 20 % des utilisateurs de conteneurs.

A lire aussi

[Red Hat prend fait et cause pour Docker... enfin presque](#)

Crédit Photo : Grzegorz Petrykowski-Shutterstock