

# [Le Cloud signe la fin des éditeurs de middleware \(tribune\)](#)

Le début des années 2000 a été l'âge d'or des éditeurs de middleware qui commercialisaient des serveurs d'applications, des EAI, des ESB, des bases de données, des solutions d'Identity & Access Management, etc. Ces solutions sophistiquées avaient un coût de licence, de support et d'intégration conséquent, pour le plus grand bonheur de leurs éditeurs. Le Cloud marque la fin de ce modèle.

## **Anatomie des plateformes Cloud**

Les acteurs du Cloud et du Web bâtissent leur plateforme en grande partie en développement spécifique, selon les principes du "Build vs. Buy". Ils ont eu recours à des middlewares Open Source (Linux, Apache, MySQL, etc.) customisés par leurs ingénieurs.

Cette approche "Fork Open Source" est intéressante pour eux pour les raisons suivantes :

- La plateforme est totalement adaptée à leurs besoins et spécificités ;
- La performance et la capacité à monter en charge sont assurées par des architectures distribuées, reposant sur les solutions comme NoSQL ou Hadoop ;
- Elle permet aux opérateurs Cloud la maîtrise de leur roadmap technique, car leur plateforme n'est dépendante ni de la vision d'un éditeur, ni de projets Open Source à la pérennité aléatoire ;
- Ne pas utiliser de middleware d'éditeur (Oracle, IBM, Microsoft, etc.) permet d'éviter des coûts de licence rédhibitoires dans une architecture de grande envergure.

Certains opérateurs Cloud maîtrisent en fait leur informatique "du sol au plafond" : Google conçoit ainsi lui-même ses serveurs et ses datacenters.

Ainsi les middlewares d'éditeurs sont exclus des plateformes Cloud. Une exception confirme la règle : Salesforce utilise les bases Oracle, car Marc Benioff est un ancien de la maison.

## **Tendances du middleware en entreprise**

De leur côté, les entreprises connaissent aussi une montée en puissance des middlewares Open Source (Linux, Tomcat, PostgreSQL...), car les Unix propriétaires, les serveurs d'applications comme ceux de WebSphere ou Oracle, sont de plus en plus considérés comme trop 'lourds' : ils sont vus comme complexes à mettre en œuvre, surdimensionnés dans bien des cas.

Et les entreprises ont retenu la leçon de l'époque SOA : les éditeurs les avaient alors poussés à acheter de coûteux ESB, un investissement superflu pour le déploiement de quelques Web Services.

En parallèle, l'usage des Cloud de type IaaS/PaaS se développe dans les grandes entreprises, clientes historiques des éditeurs de middleware. Ces plateformes Cloud leur proposent des outils clefs en main, sans étape de déploiement. Elles les débarrassent de la complexité, des Capex et d'une bonne part des coûts d'intégration.

## L'avenir bouché des éditeurs de middleware

Ainsi les éditeurs traditionnels de middlewares se trouvent pris entre deux feux :

- des opérateurs Cloud qui proposent des middlewares Open Source gratuits, issus de leur R&D
- des opérateurs Cloud qui proposent des middlewares as a Service, "out of the box", plus agiles et moins coûteux.

De fait, IBM, Microsoft, Oracle et autres ne proposent plus d'innovation dans le monde middleware, ils sont devenus suiveurs des acteurs Web/Cloud, qui ont, par exemple, créé Hadoop et les solutions NoSQL. Ils se mettent à faire du packaging d'offres Open Source comme Docker ou Hadoop, selon le modèle de RedHat. Et ils investissent désormais leur R&D ailleurs. Leur activité middleware est vouée à disparaître à moyen terme.

De son côté, VMware innove encore dans son domaine, mais l'éditeur est aussi menacé, à terme, par OpenStack d'un côté, et par les acteurs du Cloud de l'autre.



*Guillaume Plouin, architecte Cloud, est l'auteur de "Cloud Computing, Sécurité, stratégie d'entreprise et panorama du marché" et "Tout sur le Cloud Personnel" chez Dunod.*

**A lire aussi, ses précédentes tribunes :**

[Le Cloud privé est-il une escroquerie ?](#)

[Ma vie professionnelle à 100 % dans le Cloud](#)

[Le faux départ du Cloud souverain](#)

[Grâce aux conteneurs, Google Cloud va rattraper Amazon Web Services](#)

**crédit photo © Jiunn- shutterstock**