

Raspberry Pi et développement : Java SE, first class citizen (épisode 2)

Poursuivons notre série consacrée au Raspberry Pi et le développement. Le **Raspberry Pi** a droit à un traitement de faveur de la part **d'Oracle**. La firme propose en général des versions embarquées de Java pour les machines ARM. Java Micro Edition, Java Micro Edition Embedded et Java Standard Edition Embedded sont assez largement accessibles sur plate-forme ARM, mais pas le Java Standard Edition si commun au monde x86.

Sauf sur Raspberry Pi. Oracle propose en effet une version du JDK adaptée aux machines Linux pourvues d'un processeur ARMv6 ou v7. Dans la pratique, ce sont toutefois les Raspberry Pi 1 et 2 qui sont directement visés par cette offre. On retrouve en effet mention du Raspberry Pi dans toute la documentation liée à cette solution. Le **JDK 8u33** est actuellement accessible [sur le site d'Oracle](#).

Version complète et standard de **Java SE 8**, cette mouture de Java propose tous les raffinements existants sur les machines traditionnelles, dont le support graphique et **JavaFX**. Notez que Java SE 7 est lui aussi accessible sous Raspbian, mais sans JavaFX cette fois-ci. Grâce à Java SE, le Raspberry Pi pourra réutiliser sans soucis le code Java conçu pour les PC classiques.

Pour le middleware c'est Red Hat

Oracle aime le Raspberry Pi. La firme a été jusqu'à dédier un cours Java Embedded à cette carte mère ARM (les ressources de cette formation [sont librement accessibles en ligne](#)).

Mais la firme boude encore le Raspberry Pi pour les applications business. Point de Java EE donc. Peu importe. Il y a un acteur qui croit dur comme fer au Raspberry Pi et autres machines ARM : **Red Hat**. Avec la prochaine version de son OS, Red Hat proposera un support de qualité professionnelle autour de cette architecture processeur. Depuis plusieurs années déjà, l'éditeur fait des démonstrations de sa *stack middleware* **JBoss** sur Raspberry Pi. Et la firme ne vise pas seulement le marché des serveurs ARM.

Le **Fog Computing** est en effet un domaine où les offres *middleware* ont une carte à jouer. Le Fog Computing est une approche qui consiste à déléguer une partie des traitements de données aux objets connectés. Ces derniers ne se bornent ainsi plus à capter des informations qu'ils font remonter de façon brute sur Internet : ils effectuent un traitement de base permettant de structurer ces données et de les rendre plus pertinentes, en enlevant par exemple les relevés inutiles.

Cisco a mis le Fog Computing au cœur de son offre IoT. Mais ce sont bien des solutions *middleware*, comme JBoss, qui devraient marquer des points dans ce secteur, **en facilitant le déplacement de la couche business vers l'objet connecté**.

Les alternatives

L'**OpenJDK** (1.7 ou 1.8) est une autre pile Java accessible sur le Raspberry Pi, et proposée d'ailleurs directement avec certains OS Linux. C'est ainsi cette solution qui est livrée en standard avec le système d'exploitation de Red Hat. Autre possibilité, **GCJ**, un des composants clés de GCC qui permet de créer des applications natives depuis du code Java.

Voir aussi nos quiz :

[Quiz Silicon.fr – Êtes-vous un expert des langages de programmation ?](#)

[Quiz Silicon.fr – Les outils de développement rapide \(RAD\)](#)

[Quiz Silicon.fr – 9 questions sur Java 9](#)

—

[Quiz Silicon.fr – L'informatique britannique, du BBC Micro au Raspberry Pi](#)

[Quiz Silicon.fr – Savez-vous tout sur le Raspberry Pi 2 ?](#)

[Quiz Silicon.fr – 10 questions sur le projet Debian](#)