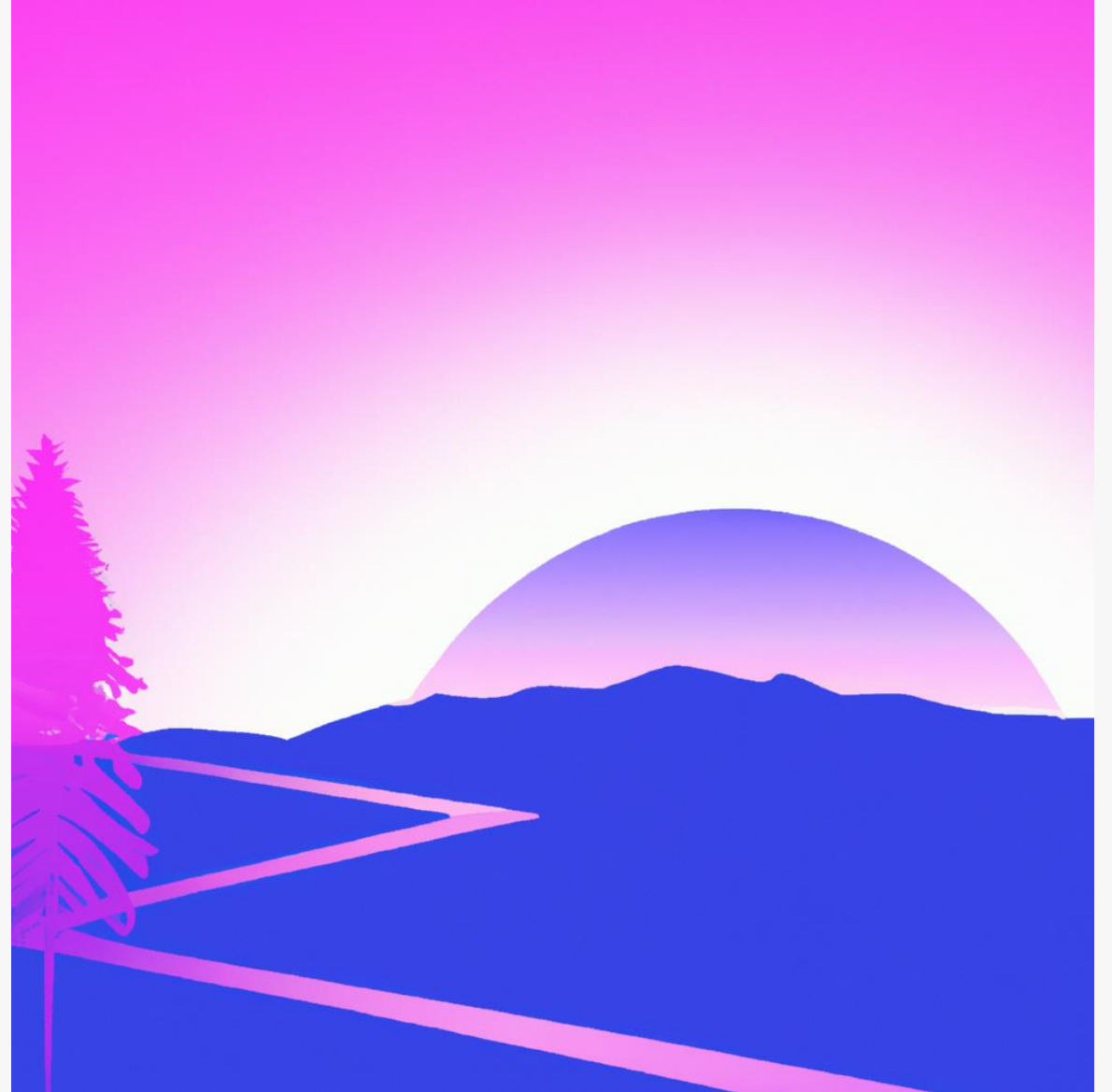


Spark Python

La JVM, environnement d'exécution pour applications Java

Crédit Agricole de Normandie

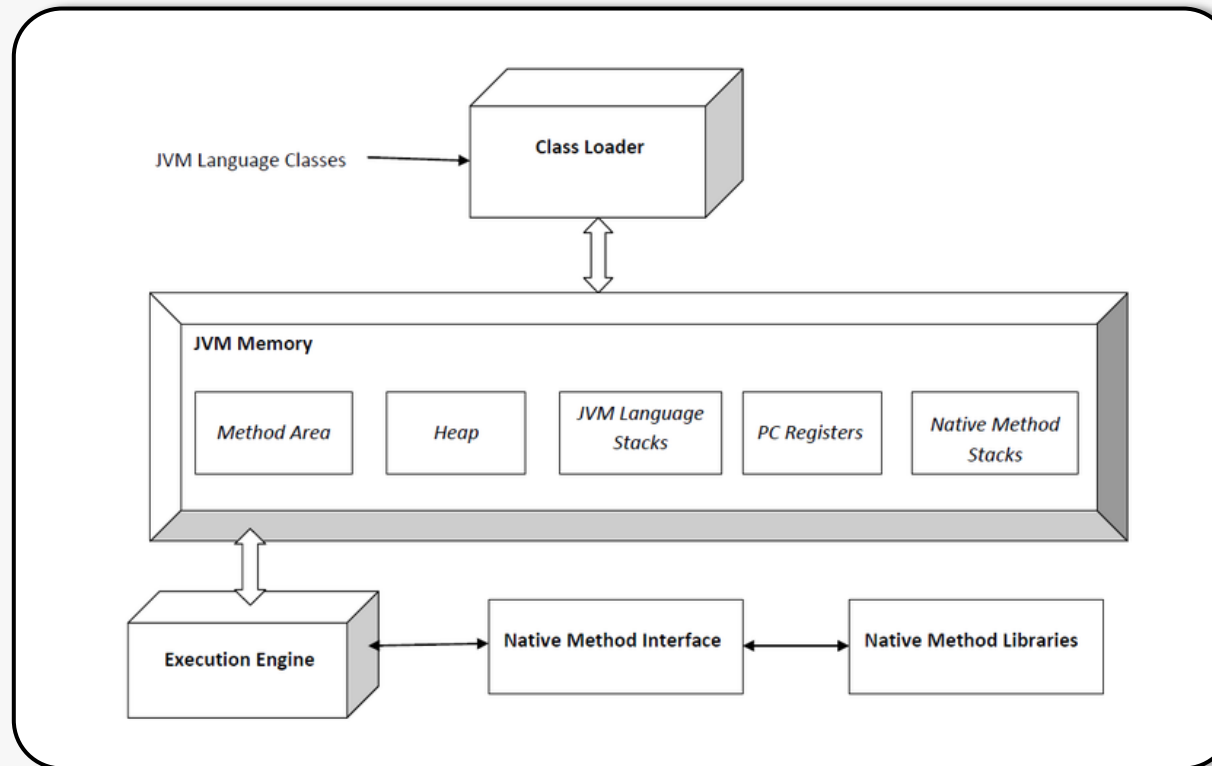
Complément – Mai 2025



JVM

En quelques mots

- Machine virtuelle qui permet d'exécuter des programmes Java.
- Elle est chargée d'interpréter le bytecode Java et de le traduire en instructions spécifiques au matériel sur lequel elle s'exécute.



Source : https://fr.wikipedia.org/wiki/Machine_virtuelle_Java

JVM

Interprétation du bytecode

- Lorsqu'un programme Java est compilé, il est converti en bytecode, un langage intermédiaire indépendant de la plateforme.
- La JVM prend ensuite ce bytecode et l'interprète ou le compile en code natif, en fonction de diverses optimisations et configurations.

JVM

Gestion de la mémoire

- La JVM gère automatiquement la mémoire (allocation, libération, garbage collection).
- Cela simplifie grandement le développement d'applications Java en éliminant la nécessité pour les programmeurs de gérer manuellement la mémoire.

JVM

Pourquoi la JVM ?

- **L'élément essentiel : la portabilité.**
- La JVM permet de rendre les programmes Java portables, car le code source Java est compilé en bytecode Java **indépendant de la plateforme**, qui peut ensuite être exécuté sur n'importe quelle plateforme compatible JVM.



La JVM assure l'indépendance du matériel et du système d'exploitation lors de l'exécution des applications Java. Une application Java ne s'exécute pas directement dans le système d'exploitation mais dans une machine virtuelle qui s'exécute dans le système d'exploitation et propose une couche d'abstraction entre l'application Java et ce système.

JVM

La réalité derrière le concept

En réalité, c'est un appareil informatique « fictif » car il est simulé par un logiciel spécifique à chaque plateforme ou couple (machine/système d'exploitation) pour permettre vraiment aux applications Java compilées en bytecode de produire les mêmes résultats quelle que soit la plate-forme, tant que celle-ci est pourvue de la machine virtuelle Java adéquate.

JVM

Et en plus

La JVM peut interpréter le bytecode Java à la volée ou le compiler en code machine natif pour une exécution plus rapide. Les JVM modernes utilisent généralement une combinaison d'interprétation et de compilation JIT pour optimiser les performances.

JVM

Une idée d'un autre aspect ?

(Se rappeler les avantages de la conteneurisation vus avec l'aparté sur Docker.)

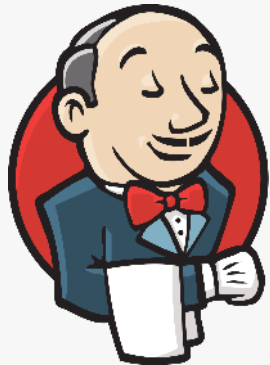
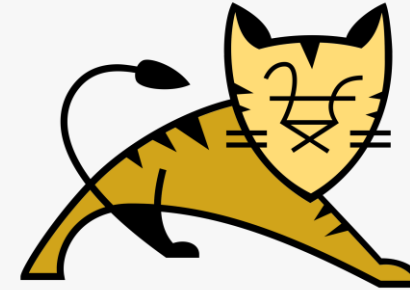
JVM

Une idée d'un autre aspect ?

- La JVM offre un environnement sécurisé pour l'exécution de programmes Java en isolant les programmes en cours d'exécution les uns des autres et en imposant des restrictions strictes sur les opérations qu'ils peuvent effectuer.
- Par exemple, la JVM utilise un modèle de sécurité basé sur des sandbox pour limiter l'accès aux ressources sensibles du système d'exploitation, telles que le système de fichiers, le réseau et les interfaces système.
- Cela aide à prévenir les attaques malveillantes et à garantir l'intégrité et la confidentialité des données pendant l'exécution des programmes Java.

JVM

Ecosystème



JVM

Ressource utile

<https://jmdoudoux.developpez.com/cours/developpons/java/chap-jvm.php#jvm-5>

Fin du complément