

LES DIFFÉRENCES ENTRE LA VIRTUALISATION ET LA CONTENEURISATION

Introduction

Pour comprendre pourquoi la technologie Docker est devenue de nos jours si populaire que ça, il est important d'abord de comprendre quel est l'intérêt des conteneurs en le comparant à la virtualisation.

Commençons d'abord par expliquer le fonctionnement de la virtualisation.

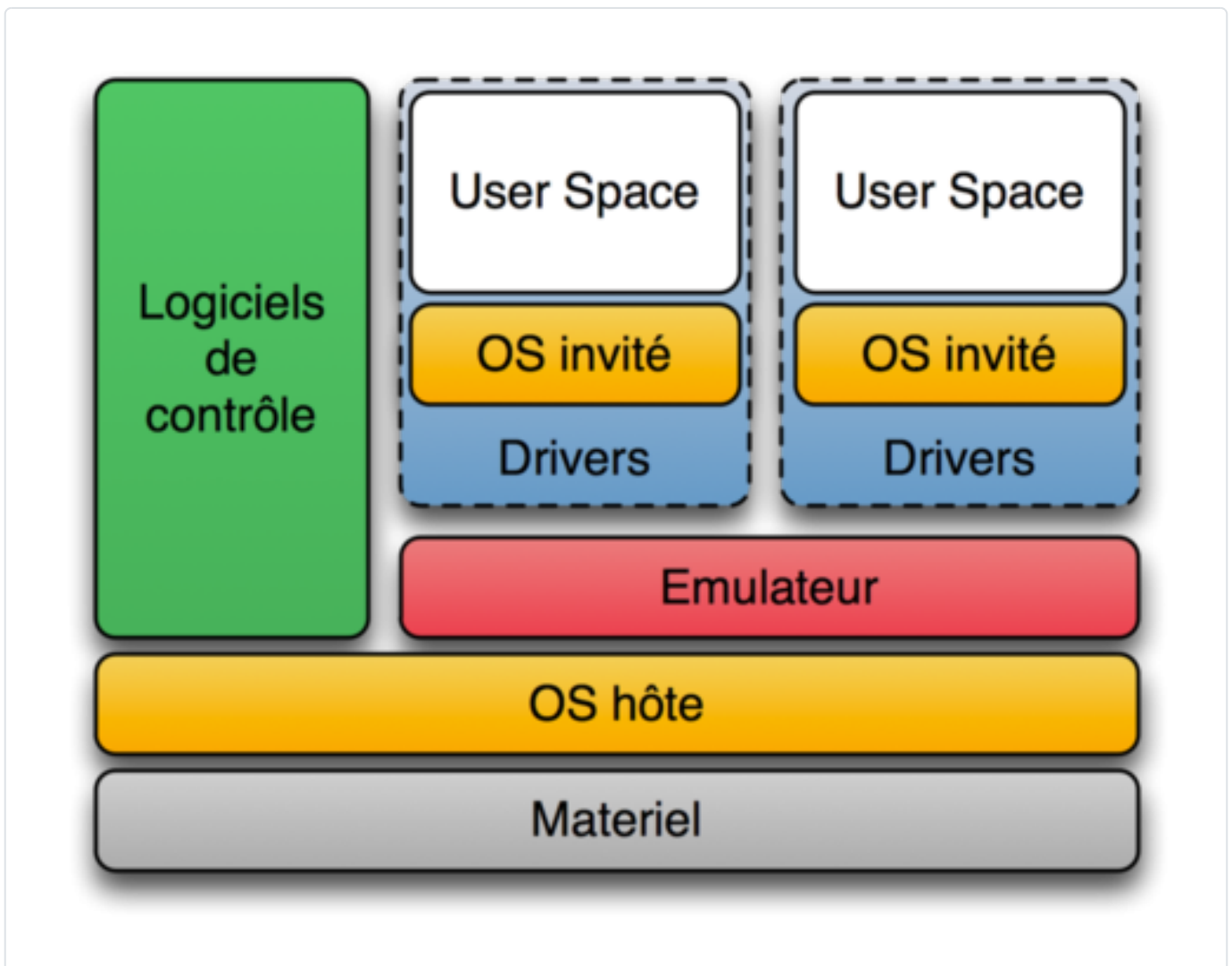
La virtualisation

Le fonctionnement de la virtualisation

Le fonctionnement de la virtualisation reste assez simple, c'est qu'au lieu d'avoir un serveur avec un système d'exploitation faisant tourner une ou plusieurs application(s), on préférera mutualiser plusieurs serveurs virtuels depuis un serveur physique grâce à un logiciel nommé l'**hyperviseur**. L'hyperviseur permet d'**émuler** intégralement les différentes **ressources matérielles** d'un serveur physique (tels que l'unité centrale, le CPU, la RAM, le disque dur, carte réseau etc ...), et permet à des machines virtuelles de les partager.

Ainsi ces machines virtuelles nommées aussi VM (Virtual Machine) bénéficieront de ressources matérielles selon leurs besoins (par exemple plus de puissance processeur et plus de mémoire vive mais avec moins d'espace disque). L'avantage

c'est qu'il est possible de modifier les ressources physiques de ces VMs en quelques clics. De plus elles **possèdent leur propre système d'exploitation** ainsi que leurs propres applications.



Les avantages de virtualisation

- Consacrer les **ressources adaptées** selon les applications qu'on souhaite mettre en place.
- Les machines virtuelles restent simples à manier. Il est possible par exemple de basculer une VM d'un lieu à l'autre voir même de sauvegarder et de dupliquer

une VM à volonté sans aucun impact visible pour les utilisateurs.

- La virtualisation **réduit les dépenses** en abaissant le besoin de systèmes matériels physiques. Elle permet ainsi de **réduire la quantité d'équipement** nécessaire et les coûts de maintenance d'alimentation et de refroidissement des composants.
- Les machines virtuelles apportent également une aisance à l'administration car un matériel virtuel n'est pas sujet aux défaillances. Les administrateurs profitent des environnements virtuels pour faciliter les sauvegardes, la **reprise après catastrophe**.

Les inconvénients de la virtualisation

- Le fait d'accéder aux ressources de façon virtuelle affaiblie les performances, cela est dû car on passe par une **couche d'abstraction matérielle** qui malheureusement doit faire des interprétations entre le matériel en place et celui simulé dans la machine virtuelle.
- Comme expliqué plus haut la virtualisation consiste à faire fonctionner sur un seul ordinateur physique plusieurs VMs avec différents systèmes d'exploitation, comme s'ils fonctionnaient sur des ordinateurs distincts. Mais malheureusement cette **couche d'OS** consomme à lui tout seul énormément de ressources alors qu'au final, ce qui nous intéresse c'est la ou les applications qui vont tourner dessus.

Conclusion

On se trouve alors avec une technologie très utile, malléable et économique pour les professionnels, mais malheureusement elle possède aussi son lot d'inconvénients, heureusement que d'autres personnes ont pensé à aller encore plus loin, et d'être encore plus efficace, et pour cela, la conteneurisation a été créée et par la suite la technologie Docker a permis de la populariser.

La conteneurisation vs virtualisation

Information

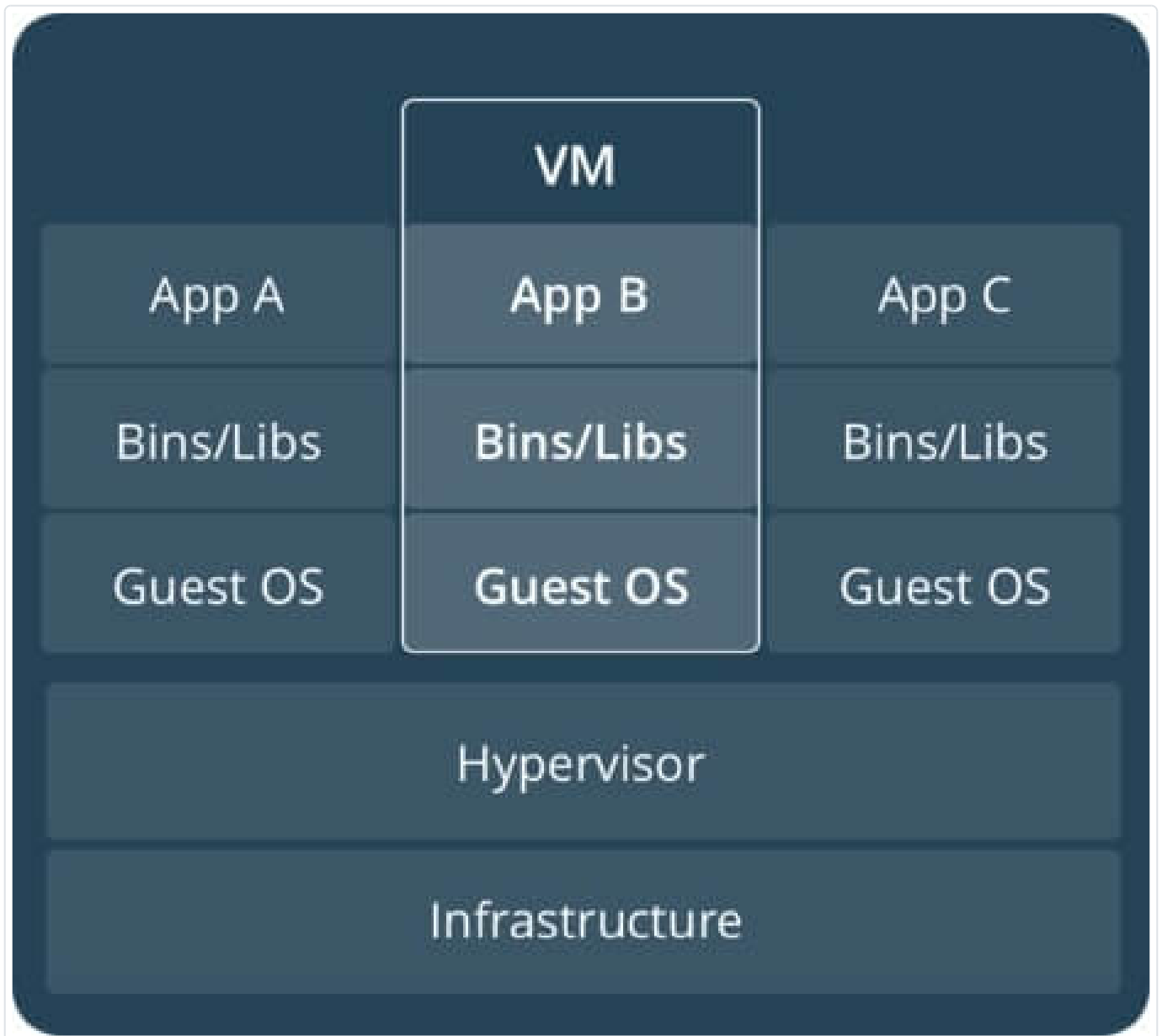
L'utilisation de conteneurs Linux pour déployer des applications s'appelle la **conteneurisation**.

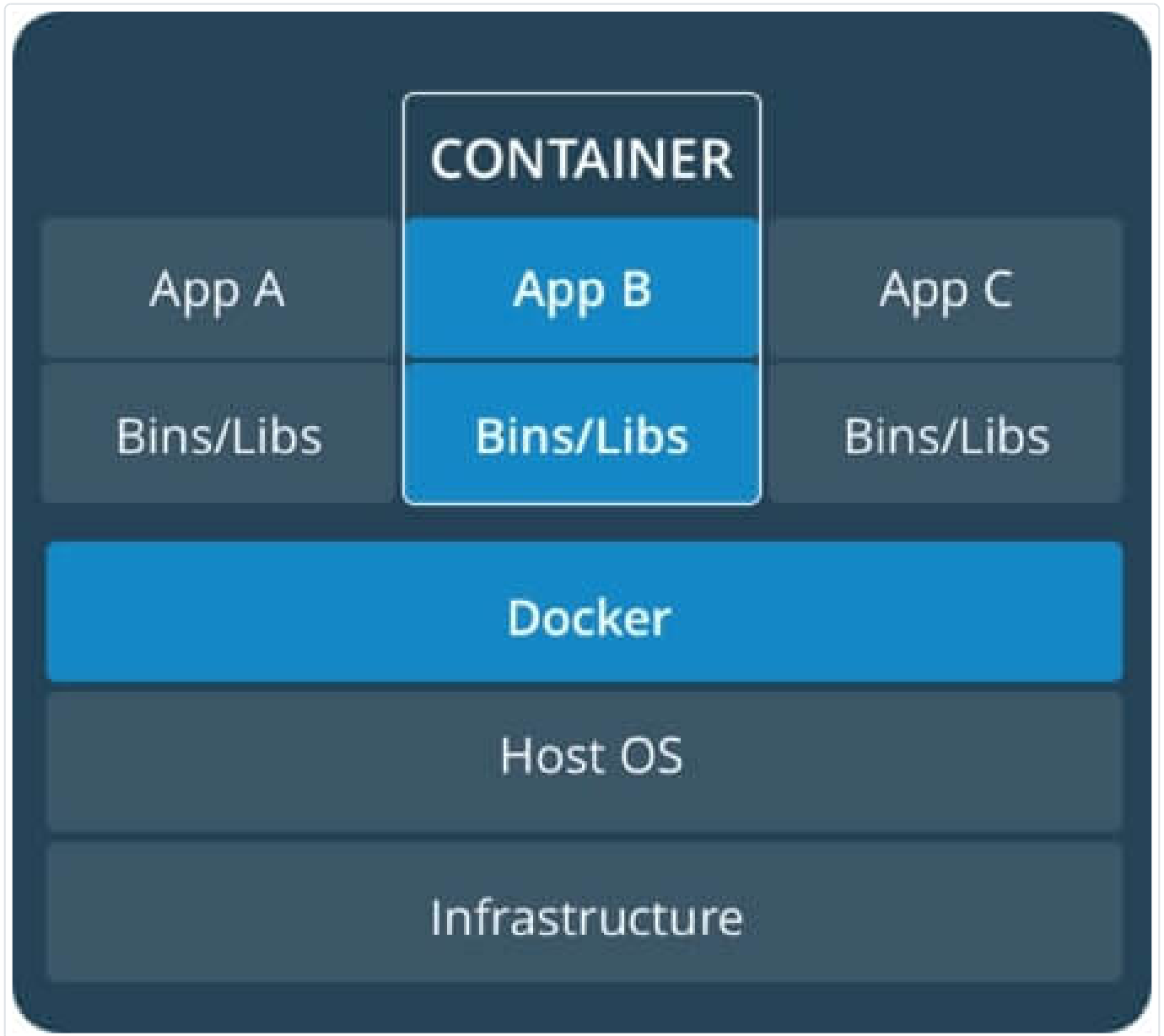
L'isolation

Dans le cas de la virtualisation l'isolation des VMs se fait au niveau matérielles (CPU/RAM/Disque) avec un accès virtuel aux ressources de l'hôte via un hyperviseur. De plus, généralement les ordinateurs virtuels fournissent un environnement avec plus de ressources que la plupart des applications n'en ont besoin.

Par contre dans le cas de la conteneurisation, l'isolation se fait au niveau du système d'exploitation. Un conteneur va s'exécuter sous Linux de manière native et va **partager le noyau** de la machine hôte avec d'autres conteneurs. ne prenant pas plus de mémoire que tout autre exécutable, ce qui le rend **léger**.

L'image ci-dessous illustre cette phase d'abstraction de l'OS.





Avantages de la conteneurisation par rapport à la virtualisation traditionnelle

- Comme vu plus haut les machines virtuelles intègrent elles-mêmes un OS pouvant aller jusqu'à des Giga-octets. Ce n'est pas le cas du conteneur. Le conteneur appelle directement l'OS pour réaliser ses **appels système** et exécuter ses applications. Il est beaucoup moins gourmand en ressources

- Le **déploiement** est un des points clés à prendre en compte de nos jours. On peut déplacer les conteneurs d'un environnement à l'autre très rapidement (en réalité c'est encore plus simple et rapide avec Docker, car il suffit juste de partager des fichiers de config qui sont en général très légers). On peut bien sur faire la même chose pour une machine virtuelle en la déplaçant entièrement de serveurs en serveurs mais n'oubliez pas qu'il existe cette couche d'OS qui rendra le déploiement beaucoup plus lent, sans oublier le processus d'émulation de vos ressources physiques, qui lui-même demandera un certain temps d'exécution et donc de la latence en plus.

Information

La virtualisation reste tout de même une technologie profitable qui a encore de beaux jours devant elle. Pour rappel nous avons besoin d'une machine avec un OS pour faire tourner nos conteneurs. Je vais me répéter mais pour moi ça reste une notion très importante, la conteneurisation permet d'optimiser l'utilisation de vos ressources tout en profitant des avantages qu'offre la virtualisation.

Les avantages de conteneurisation

La conteneurisation est de plus en plus populaire car les conteneurs sont :

- **Flexible**: même les applications les plus complexes peuvent être conteneurisées.
- **Léger**: les conteneurs exploitent et partagent le noyau hôte.
- **Interchangeable**: vous pouvez déployer des mises à jour à la volée

- **Portable:** vous pouvez créer localement, déployer sur le cloud et exécuter n'importe où votre application.
- **Évolutif:** vous pouvez augmenter et distribuer automatiquement les réplicas (les clones) de conteneur.
- **Empilable:** Vous pouvez empiler des services verticalement et à la volée.

Pourquoi Docker est si populaire ?

La conteneurisation est loin d'être une technologie récente. En réalité les conteneurs ne sont pas si nouveaux que ça, comme on pourrait le croire. Je peux en citer quelques technologies comme Chroot sur Unix (1982), Jail sur BSD (2000), conteneurs sur Solaris (2004), [LXC](#) (Linux conteneurs) sur Linux (2008). La célébrité de docker vient du fait qu'il a su permettre aux utilisateurs de gérer facilement leurs conteneurs avec une interface en ligne de commande simple.

Les conteneurs ne sont pas nouveaux, mais leur utilisation pour déployer facilement des applications l'est.

Hors de ce cours pour s'amuser un peu, on aura l'occasion sur d'autres articles de créer un conteneur Linux en utilisant des fonctionnalités nativement disponible sur Linux .