



**Christophe PICHAUD**  
Architecte Microsoft chez 'Modern Applications by Devoteam'  
christophepichaud@hotmail.com  
www.windowsscjp.com

# Windows Subsystem for Linux 2 (WSL2)

## Faire tourner des binaires Linux 64 bits sur Windows, c'est possible.

Avant, il y avait Nutcracker dans les années 1990. Il fallait recompiler. Avec Windows NT 3.51, on avait un subsystem POSIX. Ensuite, il y a eu Interix. Et enfin WSL et maintenant WSL2. Du côté des solutions open-source, il y a toujours eu Cygwin et MinGW (Minimal GNU Windows). Le meilleur est certainement MinGW et sa version 64 bits. Utiliser les application Unix/Linux sous Windows est un vieux rêve.



WSL2 a été annoncé officiellement le 6 mai dernier et disponible dans Windows 10 Juin 2019. L'équipe en charge de cette fonction mettait en avant deux améliorations importantes :

- Les performances globales du système de fichiers ;
- La compatibilité avec tous les appels systèmes.

A cela, se rajoute, comme nous le verrons dans ce court article : une nouvelle architecture, un meilleur support des binaires Linux, une expérience utilisateur identique à la version précédente.

On dispose aussi de nouvelles commandes :

- `wsl -set-version <Distro> <Version>` : pour convertir une distribution utilisant WSL1 ou WSL2. Distro est le nom de la distribution, Version est la version de WSL.
- `wsl -set-default-version <Version>` : change la version par défaut pour les nouvelles distributions que l'on déploie.
- `wsl -shutdown` : on termine sans délai les processus et les distributions ainsi que la VM WSL2.
- `wsl -list -quiet` : liste le nom des distributions. L'option `-verbose` à la place de `-quiet` affiche l'ensemble des informations des distributions.

A noter que WSL2 utilise un disque VHD pour stocker les fichiers. WSL1 n'est pas dépréciée. Pour le moment l'accès matériel depuis WSL2 est limité. Microsoft prévoit, dans le futur, de supporter les ports séries, l'USB et le GPU. **1**

### Installation

L'installation de WSL se fait par l'activation de fonctionnalités Windows (Windows Features On/Off). Ensuite, on installe une distribution au travers le Windows Store. Exemple, Ubuntu 18.04 LTS. La distribution fait 220 MB. Le seul prérequis est un Windows 10 version 16237.0 ou plus. La suite est classifiée comme Developer Tools / Utilities. Quand on parle de distribution Linux, on parle en fait de l'environnement ligne de commandes.

Vous devez disposer d'un Windows build 18917 minimum, l'activation du programme Windows Insider et l'option Fast doit être activée. En plus, vous devez activer la fonction Windows : Virtual Machine Platform.

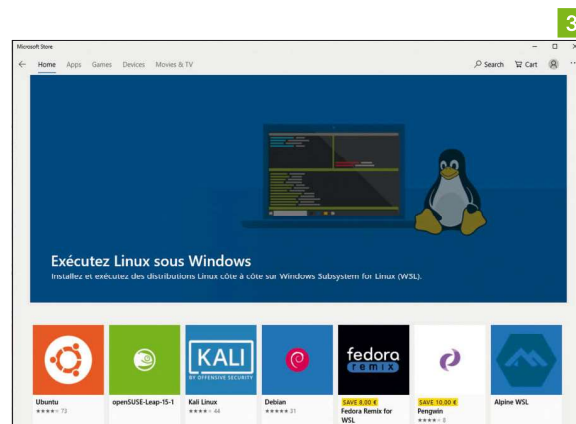
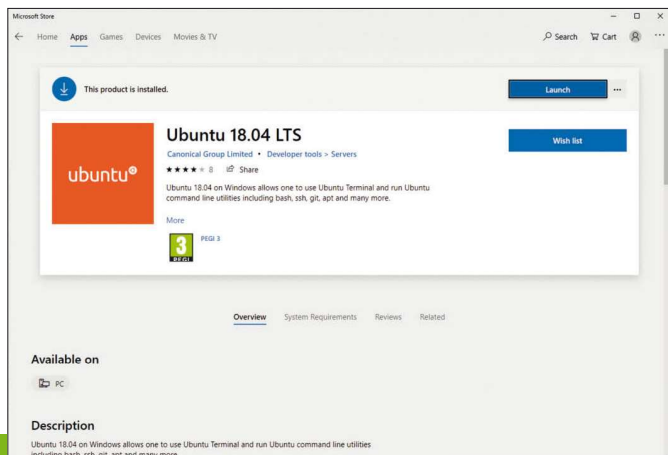
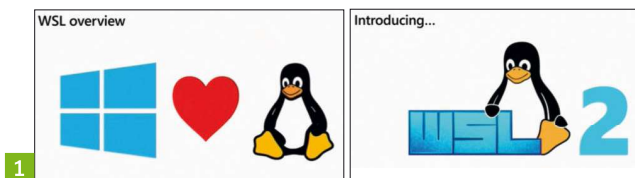
Depuis la console Powershell (en mode Administrateur) :

```
Enable-WindowsOptionalFeature -Online -FeatureName VirtualMachinePlatform
```

**2** Il existe plusieurs distributions comme Ubuntu, openSUSE, Kali Linux, Debian, Fedora, Alpine. **3**

Par exemple, Debian ne requiert que 80 MB et Windows 10 version 16215.0.

Ensuite, on se retrouve à avoir l'application bash... On est réelle-



ment dans le bash Ubuntu : ce n'est pas une émulation. Je suis sous Linux ! Je décide donc d'installer gcc, g++ et cmake via :

- sudo apt update
- sudo apt install gcc
- sudo apt install g++
- sudo apt install cmake

Depuis le bash, je tape « gcc -v » et voici le résultat. **4**

## Essayons GCC

Je décide de lancer VS Code et d'entrer un petit programme « Hello Ubuntu ! » :

- Ouverture du dossier \\wsl\$\Ubuntu\home\christophep
- Création du fichier hello.cpp

```
#include <iostream>
#include <string>

int main()
{
    std::cout << "Hello Ubuntu!" << std::endl;
    return 0;
}
```

```
christophep@DESKTOP-7VJ0H39:~$ g++ hello.cpp -std=c++17
christophep@DESKTOP-7VJ0H39:~$ ls
a.out hello.cpp
christophep@DESKTOP-7VJ0H39:~$ ./a.out
Hello Ubuntu!
christophep@DESKTOP-7VJ0H39:~$
```

## WSL2 c'est quoi ?

C'est la possibilité d'avoir un kernel Linux directement inclus dans Windows. De plus, il est possible d'avoir plusieurs distributions : Kali, Debian, Ubuntu, etc. WSL2 est plus performant que WSL1, comme nous avons pu le constater. Par exemple :

- git clone : 2.5 x plus rapide
- npm install : 4.7 x plus rapide
- cmake : 3.1 x plus rapide

La plupart du temps, cela vient des améliorations du système de fichiers. L'autre considération est la compatibilité des appels système. Surtout pour les tips & tricks de debugging. WSL1 avait des limitations.

« wsl -l » donne la liste des distributions. **5**

## Architecture de WSL

WSL est un Windows driver et un Linux Kernel en *translation layer* au-dessus du Windows NT Kernel. **6**

## Au cœur de WSL v1

Les translations sont lentes par exemple sur les opérations de fichiers car les systèmes sont différents... Parfois, la translation n'est pas possible. Exemple : renommer un fichier ouvert sous Windows donne ERROR\_ACCESS\_DENIED alors que sous Linux, ça donne SUCCESS.

Dans WSL2, la donne change : **7**

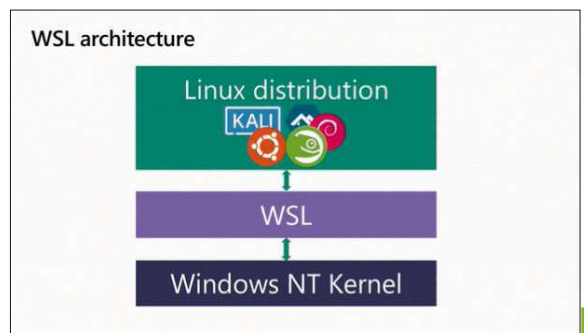
Il y a un Kernel Linux construit à partir des sources Linux qui tourne avec la virtualisation Hyper-V.

Ce n'est pas de la VM traditionnelle. C'est de la virtualisation légère ; Native x64 bits Linux support !

Développée initialement pour Windows Server, c'est la technologie Hyper-V des containers isolés qui est utilisée. Il peut y avoir de nombreux containers sur le même host (la même machine). Le temps de boot est très rapide et la création de nouveau container est une activité anodine. Les applications Windows comme Windows Defender Guard (WDAG) et Windows Sandbox sont des applications clients.

C'est de la virtualisation légère, comparée au monde des VM. **8**

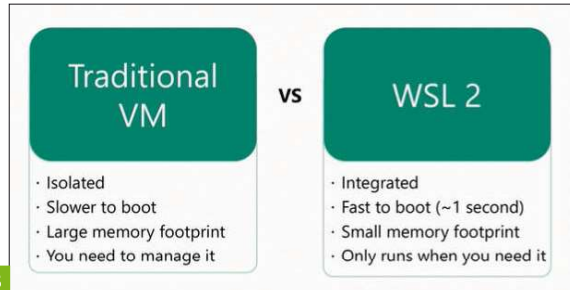
Le diagramme montre la comparaison entre les deux mondes.



## Architecture WSL2 9

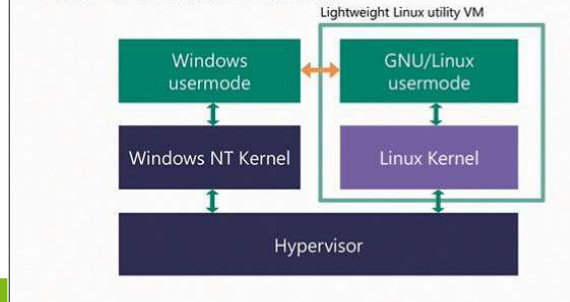
Il y a donc un vrai Kernel Linux et ça, ça change tout ! Car, il ne s'agit plus d'émuler à travers une couche de translation mais bien d'être compatible. Le kernel Linux est issu des sources de kernel.org. 10

Il est possible de lancer des processus Windows depuis le bash Linux. 11



8

## WSL 2 architecture overview



9

## Accès aux fichiers

L'explorateur de fichiers Windows peut accéder au filesystem Linux.

Il faut spécifier `\\wsl2\<distro>` 12 13

Dans ce schéma on y voit la coexistence du Noyau Linux au sein d'un environnement Windows sous forme de VM légère avec la technologie Hyper-V. On y voit le protocole d'échanges entre Windows et Linux pour les fichiers.

## Pour quels usages ?

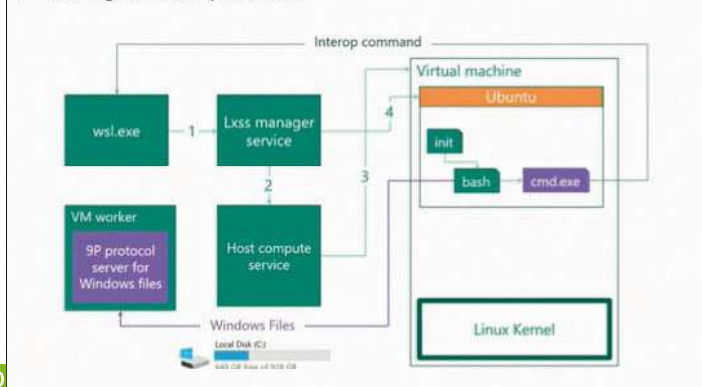
Soyons franc, il s'agit d'un mode console donc c'est orienté Serveur. On pourra y faire tourner NGINX, Apache mais quid des application X fenêtrées ? il existe dans le Windows Store une application X Server pour Windows 10. Est-ce fiable, est-ce compatible WSL2, je ne peux pas répondre à cette question.

Le truc c'est que chaque distribution fournit son environnement de commandes qui est relativement commun. Pour le développeur qui veut faire du bash avec toute la série des outils comme sed, awk, grep, tail, vi & co, c'est parfait. Pour celui qui veut builder avec le dernier GCC c'est parfait aussi. Il faut noter aussi que la consommation mémoire est relativement faible.

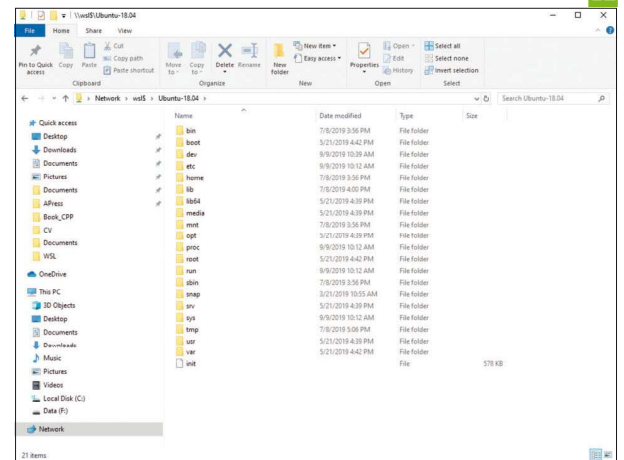
## Conclusion

WSL2 est la solution pour faire du Linux sous Windows. C'est du vrai Linux avec un vrai Kernel Linux. Plus besoin d'avoir deux portables en réseau, tout est sous Windows 10, c'est génial ! •

## Launching Windows processes

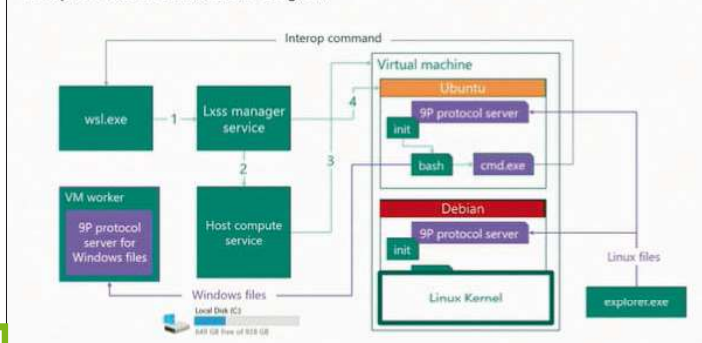


10



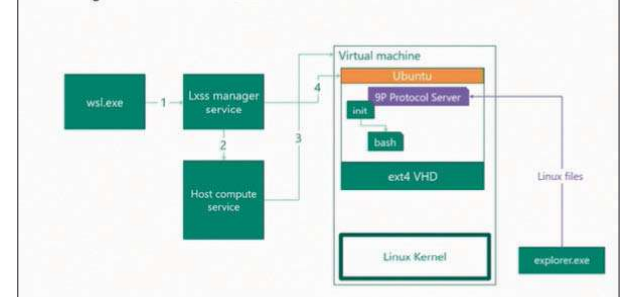
12

## Complete WSL 2 architecture diagram



11

## Accessing Linux files with WSL 2



13