

# CROSS-PLATFORM (almost) EVERYTHING USING WSL IN WINDOWS

JULY 2020 TECH TALK

MIKE NELSON – SA @ FA MS SOLUTIONS



# POINTS

- What the WSL is
- How it works
- What can I use it for?
- Demos

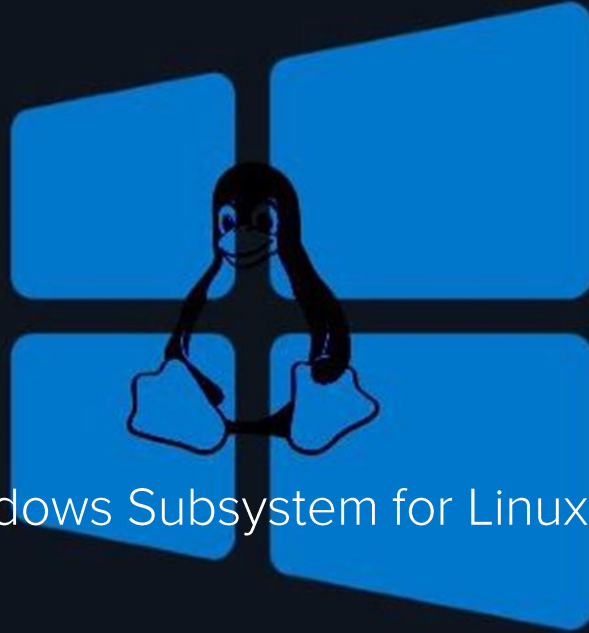
“Free Software licenses are the devil’s work.” – Microsoft PR (2001)

“Linux is a cancer that attaches itself in an intellectual property sense to everything it touches. That’s the way that the license works.” – Steve Ballmer (2001)

“Novell pays us some money for the right to tell customers that anybody who uses SUSE Linux is appropriately covered,” Ballmer said. This “is important to us, because [otherwise] we believe every Linux customer basically has an undisclosed balance-sheet liability.” – Steve Ballmer (2006)

“I may make jokes about Microsoft at times, but at the same time, I think the **Microsoft hatred is a disease.**” – Linus Torvalds

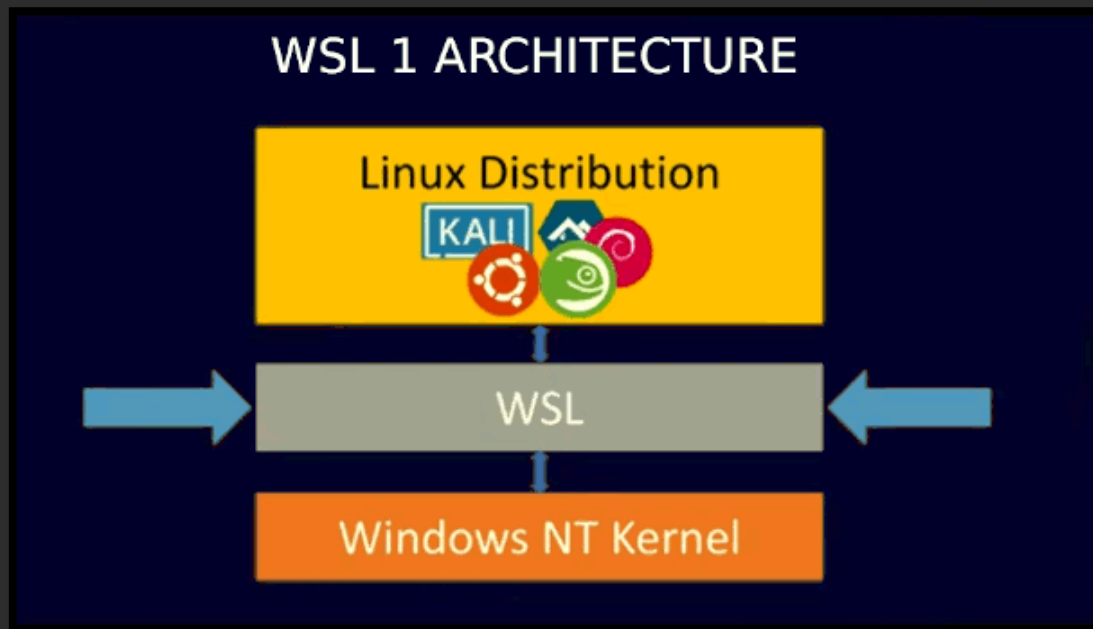




## Windows Subsystem for Linux v2

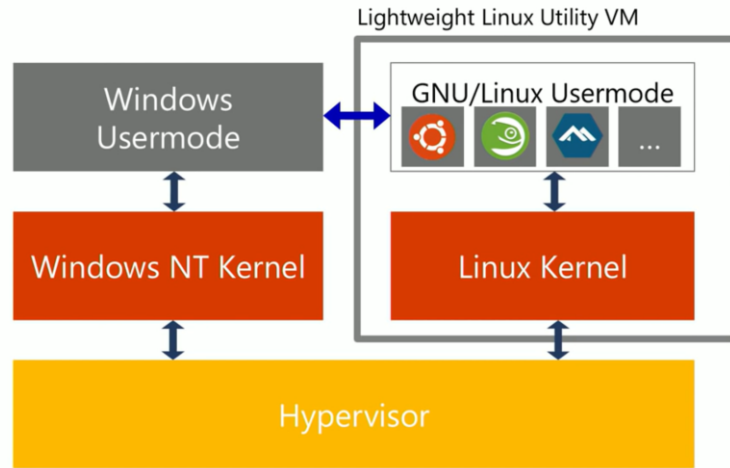
# WSL v1

August 2016 Ubuntu only  
Late 2017 Store images  
W10 1607+ & Server 2019  
Linux-compatible kernel with GNU  
Could not scale  
Not like Cygwin  
No systemd – no binaries



# WSL v2 (SO FAR)

## WSL 2 Architecture Overview



No longer emulating calls

Type1 “Lightweight” Hypervisor running single VM with ext4 VHD

One kernel running multiple distros

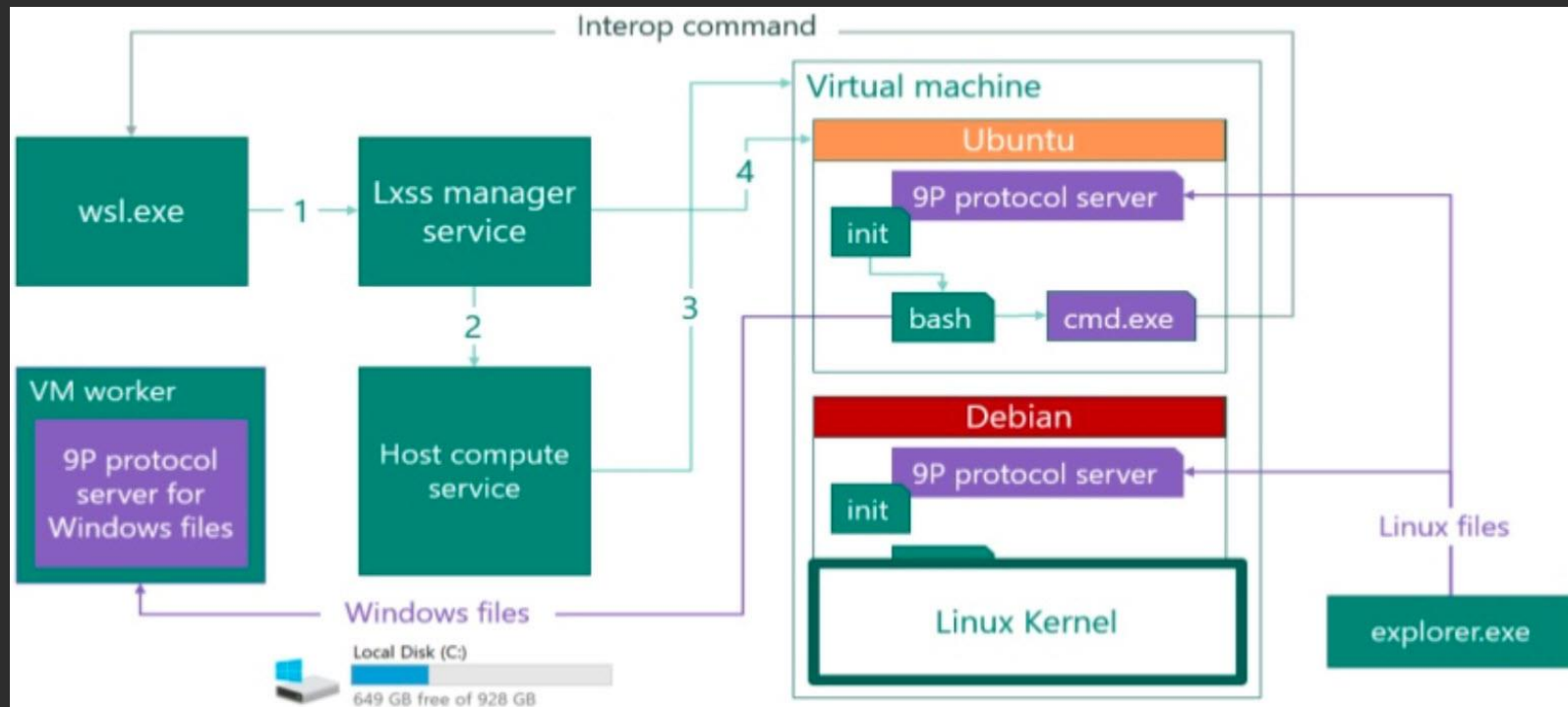
Independent IP’s

New Docker backend

Host filesystem accessible via 9P protocol (everything is a file)

IFS Network Redirector for Linux file access

# V2 OVERALL ARCHITECTURE



# WSL VERSION COMPARE

Feature	WSL 1	WSL 2
Integration between Windows and Linux	✓	✓
Fast boot times	✓	✓
Small resource foot print	✓	✓
Runs with current versions of VMWare and VirtualBox	✓	✓
Managed VM	✗	✓
Full Linux Kernel	✗	✓
Full system call compatibility	✗	✓
Performance across OS file systems	✓	✗



# WSL FEATURE AVAILABILITY

	1607	1703	1709	1803, 1809	1903, 1909	2004
lxc.exe	✓	✓	✓			
bash.exe/wslconfig.exe	✓	✓	✓*1	✓*1	✓*1	✓*1
<distro>.exe			✓	✓	✓	✓
wsl.exe			✓	✓	✓*2	✓*2
WSL Interoperability		✓	✓	✓	✓	✓
DrvFS Mounting			✓	✓	✓	✓
wsl.conf				✓	✓*2	✓*2
\$WSL_DISTRO_NAME					✓	✓
\$WSLENV				✓	✓	✓
\\wsl\$\\					✓	✓
WSL2						✓
/mnt/wsl/						✓

\*1: Kept for backward compatibility

\*2: feature of the function extended in these builds

# WHY?

Raised on \*nix

Work on appliances

Docker Containers & Kubernetes (K8s / K3s)

Cross-Develop

Flexible automation & administration

VMs optional

It's fun to play with?

# DEVELOPING IN WSL

Visual Studio Code - Remote – WSL

Atom

Sublime Text 3

Any \*nix Editor

Not using an editor/IDE that supports WSL?

LF/CR

Blanks

Case

Perms

32bit apps need `\Windows\SysNative\bash.exe`

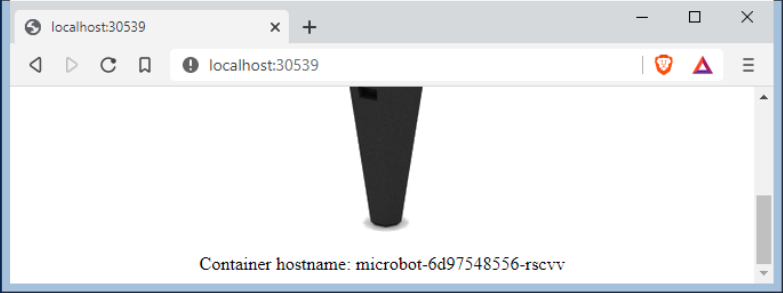
If your project requires cross-compilation using both Windows and Linux tools on the same files, use WSL v1 for now.

# DOCKER DESKTOP INSTALL WITH v2



# KUBERNETES

```
mk8s on WSLNOMICRON - Virtual Machine Connection
File Action Media View Help
mk8s@mk8smaster: ~
mk8s@mk8smaster:~$ microk8s.kubectl create deployment microbot --image=dontrebootme/microbot:v1
deployment.apps/microbot created
mk8s@mk8smaster:~$ microk8s.kubectl get pods -o wide
NAME READY STATUS RESTARTS AGE IP NODE NOMINATED NODE READINESS GATES
microbot-6d97548556-rscvv 1/1 Running 0 5s 10.1.56.108 mk8snode1 <none> <none>
mk8s@mk8smaster:~$ microk8s.kubectl scale deployment microbot --replicas=5
deployment.apps/microbot scaled
mk8s@mk8smaster:~$ microk8s.kubectl get pods -o wide
NAME READY STATUS RESTARTS AGE IP NODE NOMINATED NODE READINESS GATES
microbot-6d97548556-69jsl 1/1 Running 0 6s 10.1.56.109 mk8snode1 <none> <none>
microbot-6d97548556-6qfg4 1/1 Running 0 6s 10.1.56.143 mk8smaster <none> <none>
microbot-6d97548556-gmc9n 1/1 Running 0 6s 10.1.56.142 mk8smaster <none> <none>
microbot-6d97548556-h7cnc 1/1 Running 0 6s 10.1.56.104 mk8snode2 <none> <none>
microbot-6d97548556-rscvv 1/1 Running 0 21s 10.1.56.108 mk8snode1 <none> <none>
mk8s@mk8smaster:~$ microk8s.kubectl delete pod/microbot-6d97548556-6qfg4
pod "microbot-6d97548556-6qfg4" deleted
mk8s@mk8smaster:~$ microk8s.kubectl get pods -o wide
NAME READY STATUS RESTARTS AGE IP NODE NOMINATED NODE READINESS GATES
microbot-6d97548556-69jsl 1/1 Running 0 72s 10.1.56.109 mk8snode1 <none> <none>
microbot-6d97548556-gmc9n 1/1 Running 0 72s 10.1.56.142 mk8smaster <none> <none>
microbot-6d97548556-gn89j 1/1 Running 0 47s 10.1.56.105 mk8snode2 <none> <none>
microbot-6d97548556-h7cnc 1/1 Running 0 72s 10.1.56.104 mk8snode2 <none> <none>
microbot-6d97548556-rscvv 1/1 Running 0 87s 10.1.56.108 mk8snode1 <none> <none>
mk8s@mk8smaster:~$ microk8s.kubectl expose deployment microbot --type=LoadBalancer --port 80 --name=microbot-service
service/microbot-service exposed
mk8s@mk8smaster:~$ microk8s.kubectl get services
NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE
kubernetes ClusterIP 10.152.183.1 <none> 443/TCP 18h
microbot-service LoadBalancer 10.152.183.181 80.0.0.101 80:30539/TCP 11s
mk8s@mk8smaster:~$
```



## Windows

- Dev tools
- User utilities and scripts
- Other user customizations...

### Docker Desktop

Deploys and update  
the WSL integration  
package

Bootstrapper

Control execution of  
Linux Daemons  
(start/stop/config...)

Lifecycle  
controller

Exposes the daemon to  
the Windows Docker  
CLI, handle bind mounts

Docker Proxy

CLI &  
plugins

DockerD  
(windows)

## WSL VM

### User Distribution

- Dev tools
- User utilities and scripts
- Other user customizations...

Manage  
execution of  
Linux Daemons

### Docker Integration Package

Lifecycle  
server

DockerD

Kubernetes

Socket  
Forwarder

CLI &  
plugins

...

Forward  
connections to  
Linux Daemons

# THINGS TO KNOW & PLAY WITH

- permissions, paths “\ or /” & CaSe
- wslpath
- .wslconfig
- /etc/wsl.conf
- WSLENV (17063+)
- Hold off on GUI
- Server install is different
- LxssManager service
- wsl command line
- hcsdiag
- Custom Distros/MS kernel available
- wslutilities (Patrick Wu)



# RESOURCES

MS Docs - <https://docs.microsoft.com/en-us/windows/wsl/>

Github Repo/Issues - <https://github.com/microsoft/WSL>

Ubuntu WSL - <https://ubuntu.com/wsl>

Brian Ketelsen - <https://wsl.dev/posts/>

Hayden Barnes - <https://github.com/sirredbeard/Awesome-WSL>

Patrick Wu - <https://patrickwu.space/#>





# DEMOS

