Windows Subsystem for Linux

Marius Rusu[1], Julia Sommer[2]

Abstract:

Keywords: LNI Guidelines; LaTeXVorlage

1 Microsoft's reasons for WSL

2 WSL - What is it?

2.1 General Concept

"Windows Subsystem for Linux is a collection of [user mode and kernel mode] components that enable native Linux ELF64 binaries to run on Windows."[3] User mode applications are low privileged and depend on system calls to operate on kernel mode. Only in kernel mode low level operations directly handled by the operating system can be executed. In WSL we have the bash.exe running in user mode and initiating the Linux Instance. Further this instance submits, if necessary, native Linux system calls to be executed on a Linux Kernel. However, by virtualizing a Linux Kernel interface those system calls can be executed directly on the Windows Kernel. This virtualization is done by the LXCore/LXSS running in kernel mode.[4]

2.2 Basic Architecture

Windows Subsystem for Linux is primarily comprised of:

1. LX Session Manager

[1] Ludwig-Maximilian-Universität München, Fakultät für Informatik, Oettingenstraße 67, 80538 München, Deutschland rusu.marius97@gmail.com
[2] Technische Universität München, Fakultät für Informatik, Boltzmanstraße 3, 85748 Garching, Deutschland sommerjulia99@gmail.com
[3]
[4] https://blogs.msdn.microsoft.com/wsl/2016/04/22/windows-subsystem-for-linux-overview/, 18.04.2018 19:32 Uhr

2. LXCore/LXSS
3. Pico processes[5]

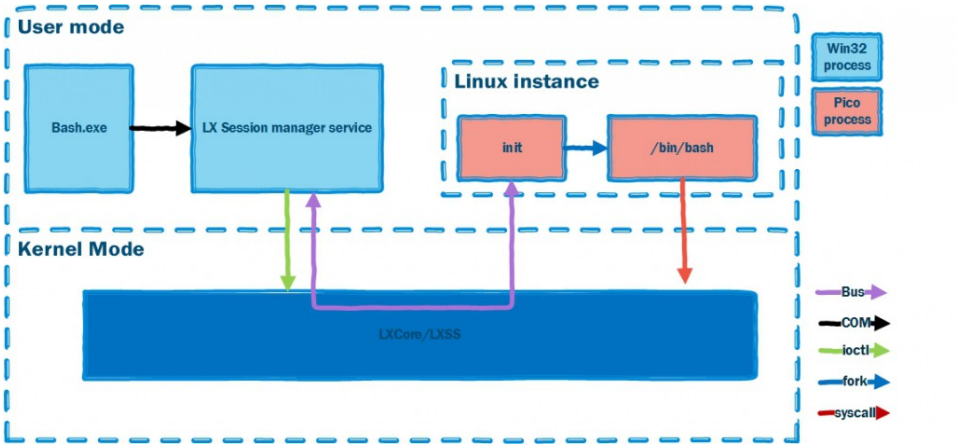

Abb. 1: Components of WSL

As depicted in the image above, the user initiates the Windows Subsystem for Linux by launching the bash.exe on Windows. This application then calls the LXCore/LXSS (green arrow) which is a "[...] driver behaving like a Linux Kernel and working in coordination with the Windows Kernel. [The Driver] would then spin up a native Linux process [being] /bin/bash (purple arrow)."[6] All other Linux processes run under /bin/bash in the so called Linux Instance that "[...] you can think of as [...] a container or a virtualized operating system environment."[7]

„By wrapping unmodified Linux binaries into pico processes we enable Linux system calls to be directed into the Windows kernel.“[8] A pico process itself is an empty process, as far as the Windows kernel is concerned and therefore cannot be handled by the Windows kernel but instead is redirected to the LXCore/LXSS (red arrow).[9] It is safe to say, that „pico processes and drivers [LXCore/LXSS] provide the foundation for the Windows Subsystem for Linux [...]. “[10]

5

6 https://blogs.msdn.microsoft.com/wsl/2016/05/23/pico-process-overview/

7 https://blogs.msdn.microsoft.com/wsl/2016/05/23/pico-process-overview/

8

9 https://blogs.msdn.microsoft.com/wsl/2016/05/23/pico-process-overview/

10

2.3 Implementation of its components

„The pico process concept originated in MSR [Microsoft Research] as part of the Drawbridge project. A goal of this project was to implement a lightweight way to run an application in an isolated environment, with the application's OS dependencies decoupled from the underlying host OS„[11] This was achieved, not by a Virtual machine, which would be too resource consuming but by „run[ning] the target application and OS entirely within the user-mode address space of a single process on the host OS. „[12] „The Drawbridge pico process is a lightweight, secure isolation container. It is built from an OS process address space, but with all traditional OS services removed. [...] All ABI [Application binary interface] calls are serviced by the security monitor, which plays a role similar to the hypervisor or VM monitor in traditional hardware VM designs.„[13]

In case of Windows subsystem for Linux this is exactly the point where pico processes are redirected to the LXCore/LXSS. „The drivers do not contain code from the Linux kernel but are instead a clean room implementation of Linux-compatible kernel interfaces. [..] Where possible, lxcore.sys translates the Linux syscall to the equivalent Windows NT call which in turn does the heavy lifting. Where there is no reasonable mapping the Windows kernel mode driver must service the request directly.„[14] Since the Windows Kernel was originally designed to support multiple operating systems, Microsoft had to „[dust] [...] of old functionality, [enhance] it for performance and correctness and it was able to go right away.„[15] Those changes in the Windows kernel enable it to execute even foreign operations like fork(). Windows applications however do not have access to those specific Linux system calls. [16]

„The File system support in WSL was designed to meet two goals:

1. Provide an environment that supports the full fidelity of Linux file systems
2. Allow interoperability with drives and files in Windows

[17]

„VolFs is a file system that provides full support for Linux file system features, including Linux permissions that can be modified through operations such as chmod and chroot [etc].„[18] Due to the fact that VolFs file system is not able to interoperate with Windows,

[11] https://blogs.msdn.microsoft.com/wsl/2016/05/23/pico-process-overview/
[12] https://blogs.msdn.microsoft.com/wsl/2016/05/23/pico-process-overview/
[13] https://www.microsoft.com/en-us/research/project/drawbridge/?from=http%3A%2F%2Fresearch.microsoft.com%2Fen-us%2Fproject s%2Fdrawbridge%2F
[14]
[15] https://blogs.msdn.microsoft.com/wsl/2016/05/23/pico-process-overview/
[16]
[17] https://blogs.msdn.microsoft.com/wsl/2016/04/22/windows-subsystem-for-linux-overview/
[18] https://blogs.msdn.microsoft.com/wsl/2016/04/22/windows-subsystem-for-linux-overview/

a second file system named DriveFs is implemented. „All fixed Windows volumes are mounted under /mnt [...] where users can access all Windows files.„[19] To make this possible the DriveFs file system meets Windows requirements such as legal file names and Windows security but looses some of the Linux features.

3 Alternatives to Windows Subsystem for Linux

3.1 Classical approach to virtualization

3.2 Virtualization via Virtual Machine

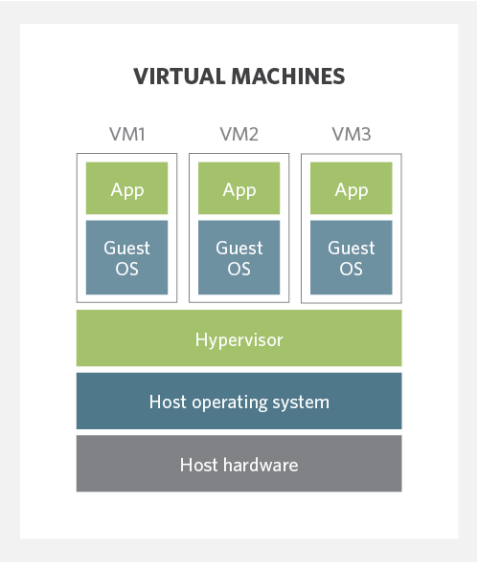

Abb. 2: Virtual Machine Overview

„The virtual machine concept allows the same computer to be shared as if it were several. IBM defined the virtual machine as a fully protected and isolated copy of the underlying physical machine's hardware. „[20] Therefore it is possible to run different applications on different operating systems at the same time on the same hardware as depicted in the image above. „The VMM [Virtual Machine Monitor] is the software component that hosts guest virtual machines.„[21] According to Prof. Dr. Kranzmüller and Dr. Danciu the Virtual Machine Monitor is firstly responsible for coordinating the access of the guest operation systems on the host's hardware and secondly for handling traps. Traps are created when a

[19] https://blogs.msdn.microsoft.com/wsl/2016/04/22/windows-subsystem-for-linux-overview/
[20] R. Rose, „Survey of System Virtualization Techniques „https://ir.library.oregonstate.edu/concern/parent/t148fh24b/file$_s ets$/0r967383v
[21] https://ir.library.oregonstate.edu/concern/parent/t148fh24b/file$_s ets$/0r967383v

guest operating system is trying to execute a privileged operation. In that case, the Virtual Machine Monitor emulates its function in order to be executed on the host operating system. [22]

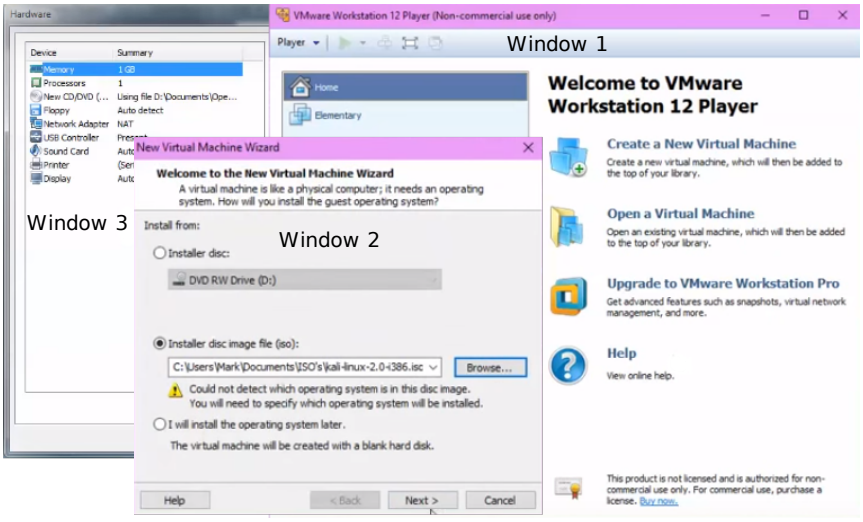

Abb. 3: VMware Player's graphic interface

A common tool for creating a virtual machine is VMware Player by VMware Inc. This application hides all the work of the Virtual Machine Monitor and comes with a graphic interface that can be seen above (Window 1). For creating a new virtual machine an image file of the guest operating system is needed, which can be either on a disc or downloaded from the internet (Window 2). After this, the user can decide on how many ressources, e.g. memory space and CPU cores the new virtual machine can use (Window 3). Finally, VMware Player launches the guest operating system in a new window as a fully functional and isolated operating system.

3.3 Virtualization via Container

Rather than virtualizing the hardware, containers use the host operating system and share its kernel. According to V. Badola containers are stripped down Virtual Machines running just enough software to deploy an application. [23] Instead of a Virtual Machine Monitor there is a container engine running on top on the host operating system which can be seen in the image above. „A container engine is a managed environment for deploying containerized applications. The container engine allocates cores and memory to containers [and] enforces spatial isolation and security [...] „[24]

[22] Skript!

[23] https://cloudacademy.com/blog/container-virtualization/

[24] https://insights.sei.cmu.edu/sei$_b log$/2017/09/$virtualization-via-containers.html$

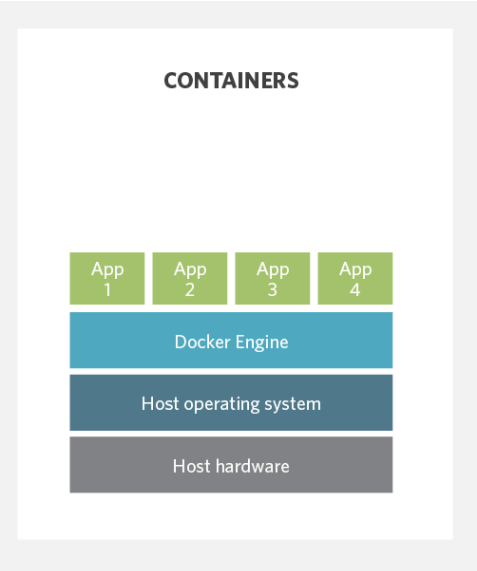

Abb. 4: Container Overview

One of the most used container engines is the open source Docker„„[...] which developed a method to give containers better portability [...]. With Docker containers, there are no guest OS environment variables or library dependencies to manage. „[25] In order to run a container in Docker, a so called Dockerfile is needed. „Dockerfile instructions provide the Docker Engine with the steps needed to create a container image.„[26] Dockerfiles contain firstly the image of the system environment of the container, secondly the application to run and thirdly the port for communication between host and container. Docker Hub provides users with a collection of images e.g. Ubuntu for download. All those steps are done in the console and a text editor for the Dockerfile, since there is no graphic interface.

3.4 Comparison to Windows Subsystem for Linux

3.5 Literaturverzeichnis

Der letzte Abschnitt zeigt ein beispielhaftes Literaturverzeichnis für Bücher mit einem Autor [Ez10] und zwei AutorInnen [AB00], einem Beitrag in Proceedings mit drei AutorInnen [ABC01], einem Beitrag in einem LNI Band mit mehr als drei AutorInnen [Az09], zwei

[25] https://searchservervirtualization.techtarget.com/definition/container-based-virtualization-operating-system-level-virtualization

[26] https://docs.microsoft.com/en-us/virtualization/windowscontainers/manage-docker/manage-windows-dockerfile

Bücher mit den jeweils selben vier AutorInnen im selben Erscheinungsjahr [Wa14a] und [Wa14b], ein Journal [Gl09], eine Website [GI14] bzw. anderweitige Literatur ohne konkrete AutorInnenschaft [An14]. Es wird biblatex verwendet, da es UTF8 sauber unterstützt und im Gegensatz zu lni.bst keine Fehler beim bibtexen auftreten.

Referenzen sollten nicht direkt als Subjekt eingebunden werden, sondern immer nur durch Authorenanganben: Beispiel: Abel; Bibel [AB00] geben ein Beispiel, aber auch Azubi et al. [Az09]. Hinweis: Großes C bei `Citet`, wenn es am Satzanfang steht. Dies ist analog zu `Cref`.

Formatierung und Abkürzungen werden für die Referenzen `book`, `inbook`, `proceedings`, `inproceedings`, `article`, `online` und `misc` automatisch vorgenommen. Mögliche Felder für Referenzen können der Beispieldatei `lni-paper-example-de.bib` entnommen werden. Andere Referenzen sowie Felder müssen allenfalls nachträglich angepasst werden.

3.6 Abbildungen

Abb. 5 zeigt eine Abbildung.

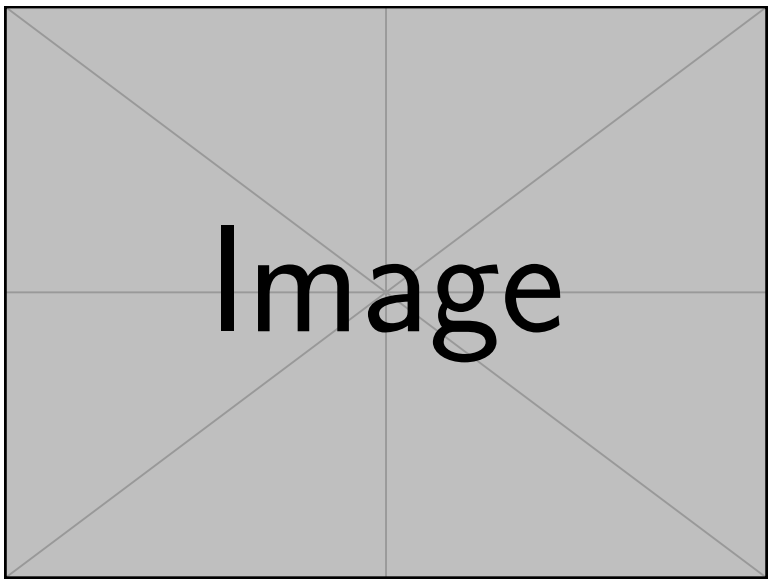

Abb. 5: Demographik

3.7 Tabellen

Tab. 1 zeigt eine Tabelle.

Überschriftsebenen	Beispiel	Schriftgröße und -art
Titel (linksbündig)	Der Titel ...	14 pt, Fett
Überschrift 1	1 Einleitung	12 pt, Fett
Überschrift 2	2.1 Titel	10 pt, Fett

Tab. 1: Die Überschriftsarten

3.8 Programmcode

Die LNI-Formatvorlage verlangt die Einrückung von Listings vom linken Rand. In der `lni`-Dokumentenklasse ist dies für die `verbatim`-Umgebung realisiert.

```
public class Hello {
    public static void main (String[] args) {
        System.out.println("Hello World!");
    }
}
```

Alternativ kann auch die `lstlisting`-Umgebung verwendet werden.

List. 1 zeigt uns ein Beispiel, das mit Hilfe der `lstlisting`-Umgebung realisiert ist.

List. 1: Beschreibung

```
public class Hello {
    public static void main (String[] args) {
        System.out.println("Hello World!");
    }
}
```

3.9 Formeln und Gleichungen

Die korrekte Einrückung und Nummerierung für Formeln ist bei den Umgebungen `equation` und `eqnarray` gewährleistet.

$$1 = 4 - 3 \tag{1}$$

und

$$2 = 7 - 5 \tag{2}$$

$$3 = 2 - 1 \tag{3}$$

Literatur

[AB00] Abel, K.; Bibel, U.: Formatierungsrichtlinien für Tagungsbände. Format-Verlag, Bonn, 2000.

[ABC01] Abraham, N.; Bibel, U.; Corleone, P.: Formatting Contributions for Proceedings. In (Glück, H. I., Hrsg.): Proc. 7th Int. Conf. on Formatting of Workshop-Proceedings. Noah & Sons, San Francisco, S. 46–53, 2001.

[An14] Anteil an Frauen in der Informatik, Statistics Worldwide, 2014.

[Az09] Azubi, L. et al.: Die Fußnote in LNI-Bänden. In (Glück, H. I., Hrsg.): Formatierung 2009. LNI 999, Format-Verlag, Bonn, S. 135–162, 2009.

[Ez10] Ezgarani, O.: The Magic Format – Your Way to Pretty Books. Noah & Sons, 2010.

[GI14] Gesellschaft für Informatik e. V., 2014, URL: `http://www.gi-ev.de`, Stand: 24. 12. 2014.

[Gl09] Glück, H. I.: Formatierung leicht gemacht. Formatierungsjournal 11/09, S. 23–27, 2009.

[Wa14a] Wasser, K.; Feuer, H.; Erde, R.; Licht, H.: Essenzen der Informatik. Verlag Formvoll, 2014.

[Wa14b] Wasser, K.; Feuer, H.; Erde, R.; Licht, H.: Ganz neue Essenzen der Informatik im selben Jahr. Format-Verlag, 2014.