

# Wieso Linux? für (Windows) Entwickler

Ein praktischer Leitfaden in 7 Phasen

von Peter Hrenka

**Tübix 2024**



## Phase 1: Zweifel

- Die wichtigste Phase
- Erkenntnis, das Windows nicht alles ist
  - Egal, was die Kollegen, Chief Architects oder die Kinder sagen
- Neugier auf was Anderes
- Ein Neue Welt

- Du bist doch Entwickler?
  - Du möchtest verstehen was passiert?
  - Du musst verstehen, warum etwas passiert!
- OpenSource Systeme wie Linux sind der einzige Weg, die Software wirklich zu verstehen
- Build-Zeiten sind unter Linux oft dramatisch kürzer als unter Windows



## Phase 2: To boldly go where no windows user...

- Jetzt mal nicht so schnell
  - erstmal WSL

### Windows Subsystem for Linux

- Ja. Windows liefert jetzt Linux aus
  - Irgendwie fair, Linux hat ja auch schon lange "WINE" gehabt
- Installieren mit `wsl --install`
  - Dauert ein Weilchen





## **Vorteile von WSL**

- Einfach zu installieren
- Verschiedene Distributionen verfügbar
- Mehrere Distributionen gleichzeitig installierbar
- Deutlich schneller als vollständige Virtualisierung
- Gute Integration mit Windows-Dateisystem

## **Nachteile von WSL**

- Grundsystem ist immer noch Windows
- Keine echten Linux-Treiber z.B. Grafik /GPU



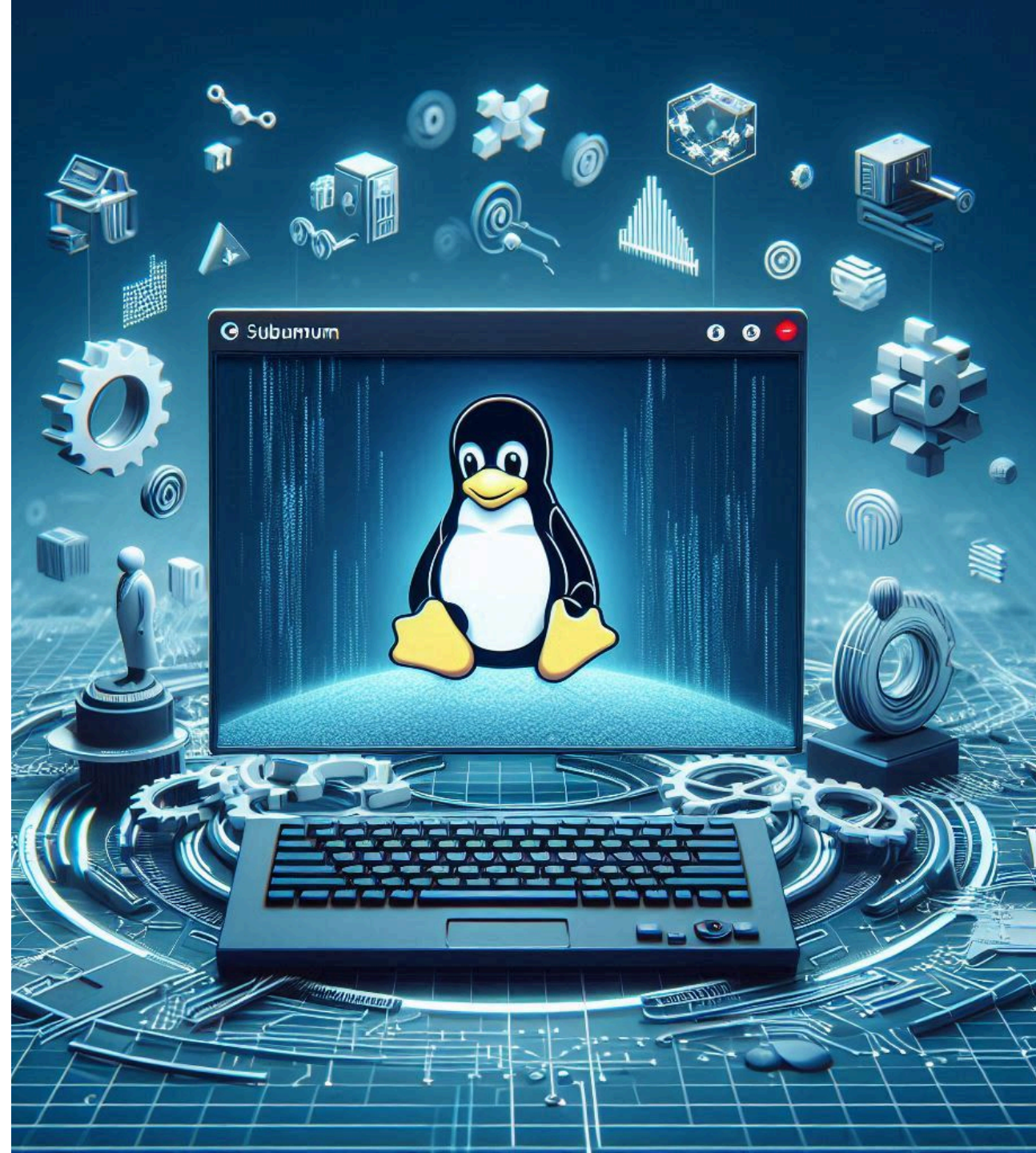
## Phase 3: Desorientierung

- Was mache ich jetzt mit dem komischen Text-Fenster?
  - So eine Eingabeaufforderung gibt es ja auch schon in Windows
    - Gab es ja auch schon unter DOS



## Tipps für Phase 3

- Ordentliches Terminalprogramm besorgen
  - [Windows Terminal](#) von Microsoft
  - Oder [MobaXTerm](#)





# Vom 0 zum Softwareentwickler in 60 Sekunden mit Linux

- `nano prog.cpp` eingeben

```
#include <iostream>
int main() {
    std::cout << "Hello Tübix!"<<std::endl;
    return 0;
}
```

- `<STR-O>` Speichern
- `<STR-X>` Editor verlassen
- Entwicklungsumgebung einrichten
  - `apt install g++ make`<sup>1</sup>
  - Leichte Abwandlungen notwendig für andere Programmiersprachen
- Übersetzen, Linken und Ausführen: `make prog && ./prog`

<sup>1</sup> je nach Distribution ist das subtil anders, vgl. Internet

## Phase 4: Hunger auf mehr

- Ganz nett hier, aber waren sie schon mal in Visual Studio Code?
  - Eigentlich ganz nettes Tool [code.visualstudio.com](https://code.visualstudio.com)
  - Gibt es auch ohne Telemetrie [vscodium.com](https://vscodium.com)
  - Kann auch mit als Windows-App mit WSL arbeiten
    - WSL ist dann "remote" (!)
- Mit X-Server (MobaXTerm) kann man VS Code auch im WSL ausführen
  - Sieht nicht wirklich anders aus
  - Vortrag zu kurz um Unterschiede zu erklären

# Phase 5: Stützräder abschrauben

- Wozu braucht man dieses Windows eigentlich gleich nochmal?
  - Als Entwickler oft gar nicht, also...

## "Richtiges" Linux installieren

- Linux vom USB-Stick booten
  - [Knoppix](#): Braucht keinen Plattenplatz
- Oder ganz viele Linuxe booten mit [Ventoy](#)
- Langfristig System aufsetzen
  - Am besten erstmal eigener Rechner
  - Wenigstens eine eigene Platte (d.h. SSD)
  - Vorsicht: Probleme mit Secure Boot
    - Bei Problemen abschalten





## Tipps für Phase 5

- Am besten mal jemanden fragen, der sowas schon mal gemacht hat
  - Auf dem Tübix sollten genug zu finden sein
- Im Internet steht gibt es viele Hilfen zur Installation (Zweitrechner!!)
  - Es gibt auch fertige Rechner mit vorinstalliertem Linux zu kaufen
  - Linux läuft im Gegensatz auch auf alten PCs

## Phase 6: Rekursion



- Haben Sie schon mal von `docker` gehört?
  - Quasi Linux im Linux, leichtgewichtig
  - `podman` ist das Gleiche, aber besser
- Leichtgewichtige Virtualisierung / Containerisierung, super praktisch für
  - Reproduzierbare Umgebungen
  - Isolierte (Web)Services
  - Dienste in die "Cloud" bringen
  - DevContainer mit VS Code
- "Works on my machine"
  - → "Let's ship my machine"

## Phase 7: Das ist alles erst der Anfang

- Eigentlich funktioniert mittlerweile alles unter Linux
  - Die meisten Applikation sind schon Web-Applikationen
    - oder werden es bald sein
  - Das geht alles besser unter Linux
- KI / AI / LLM / Buzzword
  - Ja, klar. Läuft alles auf Software Stacks die unter Linux laufen...





## Tipps für Phase 7

- Shell-Handhabung ist eine Superpower!
  - Empfehlung: Vorträge von Harald König
- Wenn das Schulungsbudget noch nicht ganz erschöpft ist:
  - Linux-Tage, OpenSource Konferenzen besuchen
- Wenn kein Budget:
  - Videos von Konferenzen anschauen
  - Linux User Groups kontaktieren