

# Linus Torvalds und die Anfänge von Linux

**Lukas Fischer**



**Interdisziplinäre Aspekte des digitalen Wandels**

**18.11.2014**

# Übersicht

- Über Linus Torvalds
- Betriebssysteme vor Linux (UNIX, GNU)
- Herausforderungen bei der Entwicklung von Linux

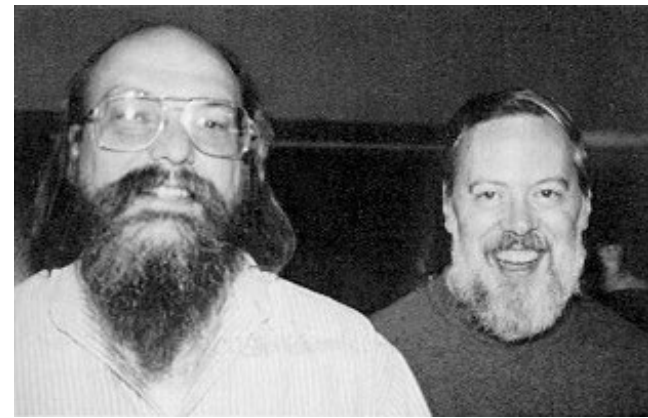
# Über Linus Torvalds

- Voller Name: Linus Benedict Torvalds
- Geboren am 28.12.1969 in Finnland
- 1988 bis 1996: Informatik-Student an der Universität Helsinki
- 1997 bis 2003: Software-Entwickler bei Transmeta in den USA
- Ab 2003: Linux-Entwickler bei den OSDL (heute: Linux Foundation)

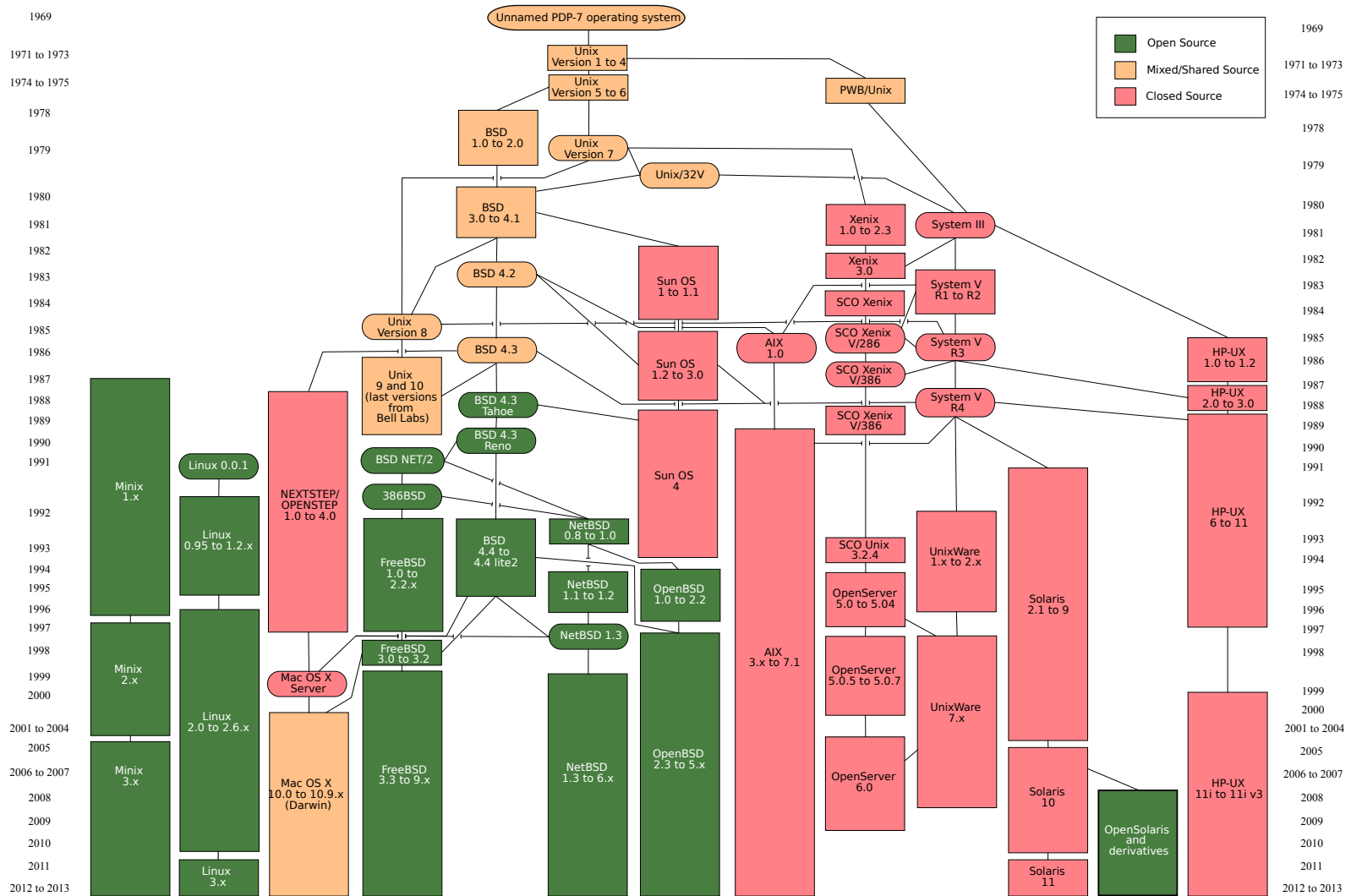


# Vorgeschichte von Linux

- 1970: Veröffentlichung von UNIX
  - Modulares Betriebssystem
  - Ein Programm erledigt genau eine Aufgabe, diese dafür sehr gut.
  - Programme werden durch Pipes verbunden, die Datenströme transportieren.
  - Daten werden in einfachen Textdateien gespeichert



# Übersicht aller unix-artigen Systeme



# GNU Project

- 1983 Gegründet
- von Richard Stallman
- Bedarf nach „freier Software“
  - In den 60ern und 70ern war es üblich den Quelltext von Programmen zu veröffentlichen
- Das freie Betriebssystem GNU soll es Benutzern ermöglichen, den Quelltext zu verstehen, zu verändern oder weiterzugeben.
- UNIX-Klon, weil:
  - „a system compatible with Unix would be convenient“



# GNU Project

- 1985: GNU Manifesto

„People who have studied the issue of intellectual property rights carefully (such as lawyers) say that there is no intrinsic right to intellectual property. The kinds of supposed intellectual property rights that the government recognizes were created by specific acts of legislation for specific purposes.

[...]

All intellectual property rights are just licenses granted by society because it was thought, rightly or wrongly, that society as a whole would benefit by granting them. But in any particular situation, we have to ask: are we really better off granting such license? What kind of act are we licensing a person to do?“

# GNU Project

- 1985: GNU Manifesto

„There is nothing wrong with wanting pay for work, or seeking to maximize one's income, as long as one does not use means that are destructive. But the means customary in the field of software today are based on destruction.

Extracting money from users of a program by restricting their use of it is destructive because the restrictions reduce the amount and the ways that the program can be used. This reduces the amount of wealth that humanity derives from the program. When there is a deliberate choice to restrict, the harmful consequences are deliberate destruction.“



# GNU Project

- 1985: GNU Manifesto

„The paradigm of competition is a race: by rewarding the winner, we encourage everyone to run faster. When capitalism really works this way, it does a good job; but its defenders are wrong in assuming it always works this way. If the runners forget why the reward is offered and become intent on winning, no matter how, they may find other strategies—such as, attacking other runners. If the runners get into a fist fight, they will all finish late.

Proprietary and secret software is the moral equivalent of runners in a fist fight.“

# GNU Project

- GNU bis 1992 fast fertiggestellt
  - Betriebssystem-Kern GNU Hurd fehlt noch
- Verwendete Lizenz: GNU GPL (General Public License)
  - Der Quelltext einer Software muss mitgeliefert werden
  - abgeleitete Software muss unter derselben Lizenz veröffentlicht werden (copyleft).

# Andere Opensource-Lizenzen

- GNU Affero GPL
  - Quelltext muss für den Nutzer einer Netzwerk-Anwendung verfügbar sein
- GNU GPLv3
  - Inkompatibel mit GPLv2
  - Schutz vor „Tivoisierung“ und Softwarepatenten
- GNU Lesser General Public License
  - Schwaches Copyleft, Verlinkung mit beliebiger Software erlaubt
- BSD-Lizenz
  - Kein Copyleft, uneingeschränkte Benutzung in proprietärer Software erlaubt

# Entstehung von Linux

- 1991 Kauf eines 386er IBM PCs
  - Kein zufriedenstellendes Betriebssystem vorhanden
  - Entwicklung einer Software, um die Hardware besser ausnutzen zu können
  - Angefangen als Hobby-Projekt

„Hello everybody out there [...]

I'm doing a (free) operating system (just a hobby, won't be big and professional like gnu) for 386(486) AT clones.“

(Torvalds, 1991)

# Entstehung von Linux

- Benutzung von GNU-Programmen zur Entwicklung (bash, gcc, ...)
  - Eigene Lizenz, die kommerzielle Nutzung ausschließt
  - 1992: Wechsel zur GNU GPL
  - Große Unterstützung durch Community
    - Keine Entwicklungsrichtung festgelegt, Torvalds nimmt alle Patches auf, die qualitativ gut sind und die Kompatibilität nicht zerstören
    - Linux-Distributionen entstehen (Software-Zusammenstellungen aus GNU und Linux, GNU/Linux genannt)
  - 1994: Linux 1.0 mit Netzwerkunterstützung
  - 1995: Linux 1.2 mit Portierungen auf andere Prozessor-Architekturen
  - 1996: Linux 2.0 mit Unterstützung für Multiprozessor-Systeme
  - 1998/1999: Investitionen in Linux durch IBM
    - Linux-Server von Dell und HP
    - kommerzielle Anwendungen für Linux
    - Microsoft sieht Linux als Konkurrenz
- „Linux is not in the public domain. 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

# Linux-Krise 2001 bis 2005

- 2001: Unzufriedenheit in der Community, da Torvalds es nicht mehr schafft alle Patches aufzunehmen
- 2002: Torvalds benutzt proprietäres Programm (Bitkeeper) zur Verwaltung des Quelltextes
  - mehr Kritik
  - 2005: Torvalds entwickelt git als Ersatz für Bitkeeper
- 2003: SCO klagt mit Unterstützung Microsofts gegen IBM wegen angeblicher Urheberrechtsverletzungen in Linux
  - Verunsicherung in der Community
  - Angst vor rechtlichen Problemen
  - 2007: SCO besitzt kein Copyright an UNIX
  - Bis heute wurde keine Urheberrechtsverletzung gefunden

# Rechtliche Probleme in Opensource

- Urheberrechtsverletzungen wegen vielen Entwicklern schwer auszuschließen
  - mögliche Lösung: Developer Certificate of Origin
- Durchsetzung von Markenrechten schwierig, da kein Geld für Anwälte vorhanden ist
  - Markenname „Linux“ wird vom Linux Mark Institute im Auftrag von Torvalds verwaltet
  - Für kleinere Projekte: Hilfe durch gemeinnützige Organisationen (z.B. Software Freedom Law Center)

# Sicherheit von Opensource-Software

- Linux-Entwicklung sehr schnell (ca. 200 commits pro Tag)
  - Entdecken von Sicherheitslücken schwierig
  - Elektronische Signaturen und Hashes für einzelne Patches
  - Code-Review durch mehrere **vertrauenswürdige** Entwickler
  - Network of trust