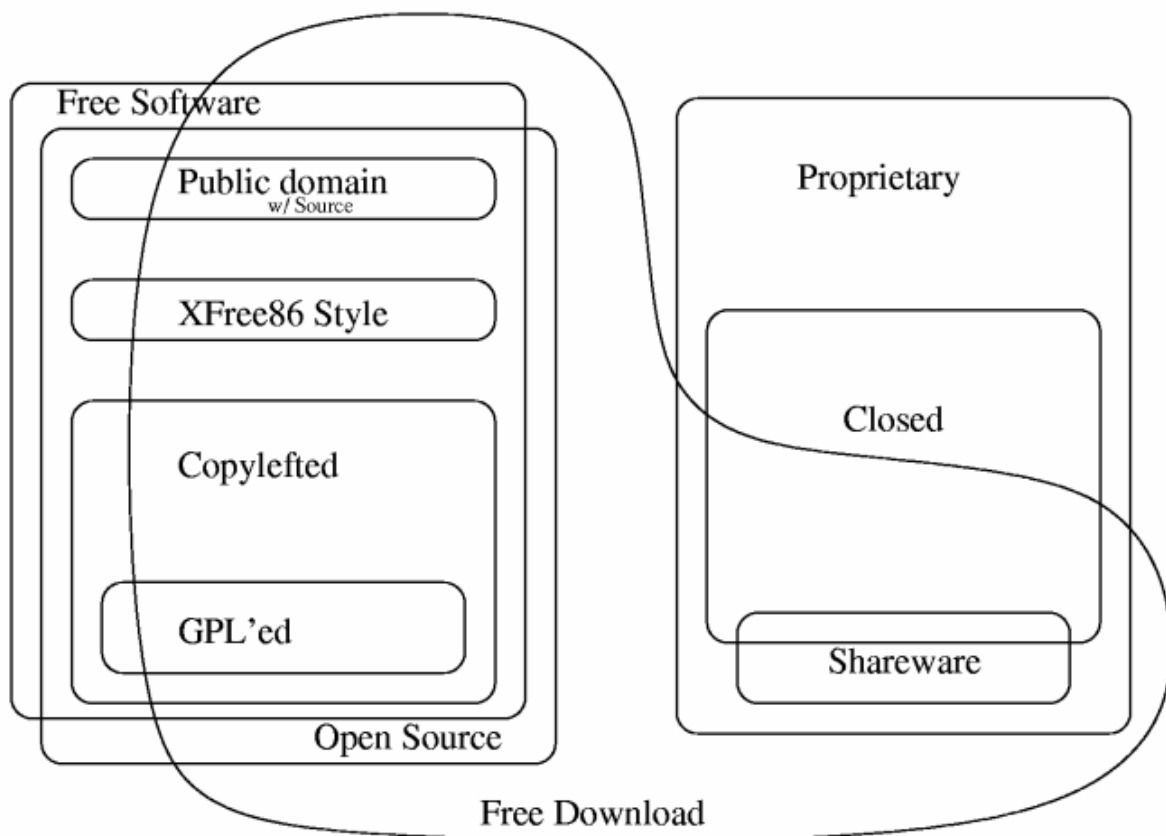


# Die „Open Source“-Bewegung

## 1. Was ist Open Source? (Das GNU-Projekt)

GNU:

- Abkürzung für: **G**NU's **N**ot **U**nix
- Das erste mal 1989 von Richard Stallman erwähnt
- Betriebssystem, welches vollkommen kostenfrei zu beziehen ist



### ***Freie Software (Open Source):***

- Jeder kann sie benutzen, kopieren und verbreiten (verändert oder unverändert, gratis und gegen ein Entgelt)
- Quellcode muss mit beigelegt werden

### Public Domain Software:

- urheberrechtsschutzfrei (juristisch)
  - ⇒ kein Copyleft
  - ⇒ veränderte Versionen sind nicht frei

### Software unter Copyleft:

- keine weiteren Einschränkung der Verbreitungsbedingungen möglich, wenn die Software verändert wird
  - ⇒ Status der freien Software wird vererbt und behält für immer diesen Status
- GPL'ed (General Public Licence)  
Software:

- Bestimmte Vertriebsbedingungen

### ***Proprietäre Software:***

- nicht frei
- Weiterverbreitung oder Änderung verboten (oder Erlaubnis vom Urheber)
  - ⇒ Kein Quellcode für die Öffentlichkeit

### Freeware:

- Weiterverbreitung erlaubt
- Keine Erlaubnis zur Veränderung der Software

## Shareware:

- Weiterverbreitung erlaubt unter der Voraussetzung, dass bei längerer Nutzung der Software ein Lizenzentgelt bezahlt werden muss
- Kopien verboten (auch für nichtkommerzielle Zwecke)

## Kommerzielle Software (Closed Software):

- Software mit dem Ziel bei der Entwicklung Profit zu erwirtschaften
- Kopien sowie Verbreitung sind vom Urheber her verboten



Richard Stallman

## 2. Free Software Foundation

- ⇒ ist eine gemeinnützige Organisation, die 1985 von Richard Stallman mit dem Zweck gegründet wurde, Freie Software zu fördern und für diese Arbeit Kapital zusammen zu tragen.
- ⇒ Executive Director ist derzeit Peter Brown
- ⇒ Bis 1990 wurden die Finanzmittel der FSF im Wesentlichen dazu verwendet, Programmierer für die Entwicklung freier Software anzustellen.
- ⇒ Seit viele Unternehmen und Privatpersonen begonnen haben, selbständig freie Software zu schreiben, konzentriert sich die Arbeit der FSF zunehmend auf rechtliche und strukturelle Belange der Freier-Software-Gemeinschaft.

Version 3 der General Public Licence:

- ⇒ Entwurf, die Ausarbeitung und die Kommunikation rund um die Schaffung der dritten Version der **GNU GPL**.
- ⇒ Das Ziel ist dabei, in Zusammenarbeit mit Entwicklern, aber auch Nutzern und Organisationen die nächste Version der verbreitetsten freie-Software-Lizenz zu schaffen sowie es unmöglich zu machen GPL-Code in Kopierschutzprogrammen zu verwenden.

### 3. Open Source vs Closed Software

- open source nicht nur mehr Begriff für Studenten, Universitäten und “Freaks” → Unternehmen und Ämter steigen auch um
- besonders im Bereich des Betriebssystems stehen sich Linux (Linus Torvalds) und Windows (Bill Gates) gegenüber
- „Warum sollten professionelle Programmierer eine Leistung vollbringen, für die sie kein Geld erhalten?“ – Bill Gates
  - Open source-Programmierer „leben“ in einer Welt ohne Hierarchien, kaum Gesetzen und keiner Polizei
  - Gesellschaftlicher Status oder Kleidermoden sind fremd
  - Wichtig sind nur geistige und handwerkliche Fähigkeiten sowie die Zusammenarbeit an einer gemeinsamen Sache
  - Arbeitsteilung nicht nach einem Geldsystem sondern Tauschprinzip
  - Freier Gedankenaustausch
    - ⇒ So entstanden der PC, das Internet, die ersten Betriebssysteme, Webbrowser und Computerspiele durch solche Programmierer
  - man kann an opensource durch Support verdienen (z.B.: Suse)
- Nicht nur Vorteile:
  - Da keine Kontrolle, der Vielzahl an Programmen möglich ist, geht die Objektivität dieser verloren
    - ⇒ z.B.: Bei Wikipedia (Enzyklopädie, wo jeder einen Artikel zu einem Stichwort schreiben kann) kam es durch „schwarze Schafe“ dazu, dass die Glaubwürdigkeit dieser Onlinebibliothek ins Wanken kam
  - Programmierer, welche freie Software entwickeln verdienen daran kein Geld
    - ⇒ Ist „nur“ ein Hobby für viele

## Closed Software:

- ⇒ Programmierer überschreiben ihre Urheberrechte an die Firma
- ⇒ weniger Leute können das Programm benutzen, da es sich nicht jeder leisten kann
- ⇒ kein Benutzer kann es adaptieren oder verbessern
- ⇒ andere Entwickler können nichts von dem Programm lernen oder neue Arbeit darauf aufbauen
- ⇒ erzwungene Updates durch Standardisierung der Software
- ⇒ Universitäten entwickeln Programme nur noch bis zu einen Punkt, welchen sie selbst als „beendet“ bezeichnen und geben dieses Programm an Firmen weiter die dieses nutzbar für jedermann machen
  - Linzenzen und Gesetze werden ignoriert

## Vorteile:

- ⇒ schafft Arbeitsplätze (Microsoft beschäftigt 76000 Menschen in Deutschland)
- ⇒ wäre Software komplett open source dann gäbe es weniger Programmierer (kein Geldverdienst mehr durch programmieren)
- ⇒ Updates werden nach Zeitplan erarbeitet und kommen somit zu einem definierten Zeitpunkt heraus
- ⇒ Bei Programmen welche für Sicherheit zuständig sind (Firewalls, Anti-Virenprogramme, Spam-Filter,...) ist es von Vorteil, dass der Sourcecode nicht öffentlich ist, da Sicherheitsschranken so schwerer überwindbar sind

#### 4. Quellen:

- ⇒ Juli Zeh: Es werde Linux. Die Zeit, 30. 3. 2006, S. 39-40.
- ⇒ Richard Stallman: Why software should be free. <http://www.gnu.org/philosophy/shouldbefree.html> (31. 3. 2006).
- ⇒ Kategorien freier und unfreier Software. <http://www.gnu.org/philosophy/categories.de.html> (31. 3. 2006).
- ⇒ Bradley M. Kuhn: The GNU GPL and the American Dream. <http://www.gnu.org/philosophy/gpl-american-dream.html> (31. 3. 2006).

#### *Ergänzendes Material:*

- ⇒ Gunhild Lütge: Programmierer aller Länder vereinigt Euch! Die Zeit, 23. 10. 2003, S. 17-18.
- ⇒ Stefan Krempl: Ja zum Monopol. c't, 2004, Heft 1, S. 45.
- ⇒ Bernd Graff: Unleserlicher Mist. Süddeutsche Zeitung, 7. 12. 2005, S. 13.
- ⇒ Runderneuerung für freie Softwarelizenz. Computer Zeitung, 12. 12. 2005, S. 6.
- ⇒ Opensource-Demokratie versus Klarheit. Computer Zeitung, 12. 12. 2005, S. 6.
- ⇒ Opensource-Lizenz zeigt Patenten die Stirn. Computer Zeitung, 23. 1. 2006, S. 4.
- ⇒ Opensource-Lizenzen sind nicht frei verfügbar. Computer Zeitung, 6. 2. 2006, S. 12.
- ⇒ "Softwarepatente verhindern bessere Methoden". Computer Zeitung, 6. 3. 2006, S. 12.