

## Freie Software in der Schule – sinnvoll oder nicht? von Sebastian Seitz

**S**pricht man über Schule und Software, so denken die meisten recht schnell an den Informatikunterricht. Das ist bei Freier Software nicht anders. Aber Freie Software ist deutlich mehr als nur Software. Die ihr zu Grunde liegende Philosophie hat einiges mit der Idee (und auch der Praxis) der Schule gemein.

### Warum Freie Software?

Der erste Absatz von §2 der Lernmittelverordnung [1] des Landes Berlin liest sich im ersten Moment wie ein Freifahrtschein für Freie Software in der Schule: „*Unterscheidet sich ein Lernmittel der Art, dem Inhalt und der didaktisch-methodischen Aufbereitung nach nicht wesentlich von einem anderen, so ist das preisgünstigste Lernmittel auszuwählen.*“ Eine Aussage, die zu denken gibt. Schließlich sind nicht nur Lehrbücher, Lineale oder Taschenrechner als Lernmittel definiert, sondern zusätzlich die Schulrechner und somit auch die verwendete Software.

Wenn es darum geht, in der Schule Freie Software einzusetzen, kommt es bei Lehrern oftmals zu Verwechslungen mit Freeware. Das mag daran liegen, dass das häufigste Argument für den Einsatz Freier Software deren kostenlose Verfügbarkeit ist. Und darin ähneln sich Freie Software und Freeware nun einmal wesentlich. Kostenlose Software ist für Schulen äußerst praktisch, denn in Zeiten von Sparmaßnahmen und Haushaltssperren seitens der Träger leiden besonders die

Bildungseinrichtungen. Zusätzlich zur Tatsache, dass sie meistens kostenlos verfügbar ist, bietet Freie Software aber noch viel mehr. Alle Möglichkeiten, Pflichten und Potenziale Freier Software werden durch ihre jeweilige Lizenz definiert – das ist nichts Neues. Im Allgemeinen spricht man dabei von den vier Freiheiten [2]: Nutzen, Teilen, Studieren, Verbessern. Aber welche davon sind eigentlich für die Schule relevant?

### Vier Freiheiten

Die freie Nutzung einer Software bezeichnet den Aspekt, dass die Software von jedem und für alles benutzt werden darf. Daneben ist Freie Software sehr oft kostenlos, was aber nicht zwingend so sein muss. Dennoch ist dieser Punkt essentiell für die Schule. Es fallen keine Kosten an, wenn es darum geht, die Software auf einer großen Anzahl an Computern zu installieren. Es ist auch unproblematisch, neue Software zu testen und diese bei Nichtgefallen wieder zu entfernen, da auch hier keine Kosten außer der Arbeitszeit des jeweiligen Lehrers anfallen. Die freie Nutzung ist daher der für die Schule entscheidende Faktor bei der Wahl Freier Software.

Beim freien Studieren sieht das anders aus. Während die freie Nutzung der Software allen Parteien der Schule zugute kommt, ist das freie Studieren auf den ersten Blick nur für Schüler und Lehrer des Informatikunterrichts relevant. Und auch bei diesen nur sehr eingeschränkt. An manchen Schulen existieren Computer-AGs, bei de-

nen sich diese Tatsache wesentlich stärker auf die Arbeit auswirkt. Allerdings werden in diesen kaum Änderungen an Programmen durchgeführt, die für die Schule selbst oder zumindest für den Unterricht relevant sind wie beispielsweise an einem Webbrowser oder einem Office-Paket. Die direkten Vorteile scheinen also beim freien Studieren eher gering. Die indirekten hingegen sind deutlich größer. Durch die Möglichkeit, den Programmcode zu studieren, wird einerseits die Möglichkeit gegeben, autodidaktische Fähigkeiten zu befriedigen. Andererseits transportiert diese Tatsache eine der Schule von Anfang an innewohnende Botschaft: Sei transparent. Wissen soll nicht nur wiedergegeben, sondern verstanden werden. Dafür muss es transparent, also durchschaubar sein. Nutzt eine Schule Freie Software, so kann sie diese Botschaft viel authentischer transportieren. Eine Schule, die in ihrem pädagogischen Profil stets auf Ideale wie Mitbestimmung, Transparenz und Gleichberechtigung verweist, aber durchweg IT-Strukturen nutzt, die diesen Idealen widersprechen, macht sich auf den zweiten Blick unglaublich – häufig jedoch ohne es zu wissen.

Dem freien Teilen kommt in der Schule eine besondere Stellung zu. Während man früher glaubte, dass alle Schüler mit den gleichen Voraussetzungen die Schule besuchen, weiß man mittlerweile, dass Homogenität in der Schule oder im Klassenverband eher Wunschdenken als Realität ist. Denn nicht jedes Elternhaus kann genug



Geld für eine bestimmte Schulsoftware bereitstellen. Hier kommt das freie Teilen zum Tragen. Die Schule kann die genutzten Programme direkt an die Schüler weitergeben. Geld spielt so für den Lernerfolg der Schüler keine Rolle. Zudem können diese zu Hause mit der gleichen Software weiterlernen. Freie Software kann auf diese Art und Weise soziale wie auch didaktische Verbesserungen bewirken.

Die vierte Freiheit, das Verbessern der Software, ist die Fortsetzung der Freiheit, die Software zu studieren. Auch hier erschließen sich die Vorteile nicht sofort, da sie vor allem indirekt wirken. Ein häufig angeführtes Argument folgt der Idee, dass durch den freien Zugriff aller Personen auf den Quellcode Sicherheitslücken schneller aufgespürt und beseitigt werden können. Davon würde auch die Schule profitieren. Gerade hier muss dem Datenschutz eine besondere Position eingeräumt werden, denn Schulen arbeiten mit vertraulichen Informationen, wie zum Beispiel Zensuren oder Informationen über eventuelle Lernbehinderungen.

## Nebeneffekte

Nahezu alle Bundesländer verfügen über schulische und außerschulische Medienkonzepte [3], die den Begriff der Medienkompetenz enthalten. Hierbei geht es nicht nur um technisches Verständnis, sondern auch um Fragestellungen zur Entstehung und Nutzung von Medien (siehe „Medienkompetente Ausbildung? Nur auf Basis Freier Software“, *freiesMagazin* 10/2010 [4]).

Freie Software bietet eine hervorragende Möglichkeit, den Entstehungsprozess einer Software zu erläutern und, je nach Vorwissen der Lerngruppe und Komplexität des Programms, aktiv zu unterstützen oder zu begleiten. Eine Einbettung des Themas kann auch im Zusammenhang mit den Fächern Philosophie oder Sozialwissenschaft stattfinden und ist nicht nur auf den Informatikunterricht beschränkt. Die ideologisch-geschichtlichen Grundlagen Freier Software bieten hierfür genügend Material und Diskussionsstoff. Dabei gibt es keine Beschränkungen auf bestimmte Schulformen. Die Medienkonzepte der Länder existieren für alle allgemeinbildenden Schulformen von Grund- bis Gesamtschule.

## Wer kümmert sich?

Bundesweit gibt es noch keine ausreichende Lösung für das Problem der Zuständigkeit der IT-Administration an Schulen. Oft kümmern sich engagierte Lehrer um das Schulnetzwerk. Hierfür werden sie „freigestellt“, meist können sie so zwei bis vier Stunden für die Pflege und Entwicklung der IT-Struktur nutzen. Diese Zeit ist vollkommen unzureichend, wenn man sich vor Augen hält, dass einige Schulen über mehr als 100 PCs verfügen. Ein professioneller Partner ist meist unumgänglich, durch den Einsatz Freier Software könnten die nötigen Mittel frei werden.

## Fazit

Freie Software bietet in der Schule vielfältige Optionen, das IT-Konzept zu erweitern und die Strukturen transparenter zu gestalten. Sie ist kosteneffizient, sicher und kann helfen, soziale Be-

nachteiligung als Entwicklungsfaktor von Schülern zu mindern. Sie ist damit weit mehr als eine Software-Alternative für besonders interessierte Lehrer und Schüler aus Informatikunterricht und Computer-AGs. Sie ist in der Lage, die humanistische Idee der Schule und ihres Bildungs- und Erziehungsauftrags in den Schulalltag zu integrieren und kann helfen, Schule effizienter, freier und gerechter zu gestalten.

## LINKS

- [1] <http://www.berlin.de/imperia/md/content/sen-bildung/rechtsvorschriften/lernmittelvo.pdf?start&ts=1300181404&file=lernmittelvo.pdf>
- [2] <http://fsfe.org/about/basics/freesoftware.de.html>
- [3] <http://www.bildungserver.de/zeigen.html?seite=2884>
- [4] <http://www.freiesmagazin.de/freiesMagazin-2010-10>

### Autoreninformation



**Sebastian Seitz** ([Webseite](#)) ist Diplom-Pädagoge und nutzt seit fünf Jahren Ubuntu. Anfang des Jahres wurde er Mitglied des Educational-Teams der FSFE. Freie Software und (Schul-)Bildung sind seine Lieblingsthemen.

*Diesen Artikel kommentieren*

