

Open-Source in der Schule

Andreas Rittershofer

20. August 2006

1 Software in der Schule

Schule ohne Computer ist heute nur mehr schwer vorstellbar. Die eine Seite hiervon ist die Beschaffung der Hardware, die andere die der Software. Bei der Software wiederum ist zu unterscheiden zwischen dem Betriebssystem und der Anwendungssoftware. Sehr häufig bringen neu angeschaffte Computer als Betriebssystem eine aktuelle Version von Microsoft Windows mit, momentan ist dies Windows XP. Dies hat nicht nur Auswirkungen darauf, welches das typische Betriebssystem in der Schule ist, sondern auch oft darauf, welche Anwendungssoftware eingesetzt wird.

Dieser Text versucht zu zeigen, dass einerseits mit Open-Source-Software in der Schule ein großer Teil der Anforderungen abgedeckt werden kann und auch welche Vorteile für die Schule damit verbunden sind bis hin zu gesellschaftspolitischen Fragestellungen, die vor der Schule nicht halt machen.

2 Open Source in der Schule

Nähere Betrachtung der Open-Source-Szene zeigt eine enorme Vielfalt im Angebot und damit einher gehend auch, dass ein beachtlicher Teil der schulischen Anforderungen an die als notwendig angesehene Funktionalität durch Open-Source-Software abgedeckt werden kann. Doch warum sollte eine Schule dies tun? Die Vorteile sind u.a.:

- In Zeiten immer knapper werdender Kassen der Schulträger entlastet dies den stets chronisch klammen Schuletat merklich. Dabei geht es nicht nur um die erstmalige Beschaffung, sondern es sind ja auch immer wieder Updates erforderlich, deren Kosten nicht übersehen werden dürfen, auch weil die Updatebedingungen teilweise immer kundenunfreundlicher werden.
- Jeder Lehrer und jeder Schüler kann sich die in der Schule eingesetzte Open-Source-Software ganz legal und kostenfrei auch zu Hause auf dem privaten Rechner installieren und damit die gleiche Arbeitsumgebung einrichten wie in der Schule — in Zeiten zunehmenden Computereinsatzes in der Schule und weiter Verbreitung privater Computer immer wichtiger.

Hinzu kommt, dass den Eltern keine unnötigen Kosten aufgebürdet werden oder gar am Ende der Schulträger im Rahmen der Lernmittelfreiheit dafür aufkommen muss. Auch die gesamte Problematik des Urheberrechts mit Raubkopien ist kein Thema mehr, was immer unausgesprochen im Raume steht, wenn in der Schule kommerzielle Software eingesetzt wird — die Schüler wissen sich dann schon zu „helfen“; Einsatz kommerzieller Software in der Schule wird in der Realität immer viele Schüler dazu verleiten, eine Raubkopie dieser Software auch zu Hause einzusetzen. Hinzuweisen ist hier auch auf Shareware, die häufig dauerhaft eingesetzt wird, ohne sie bezahlt zu haben, was nicht zulässig ist. Selbstverständlich kann in der Schule nur Shareware eingesetzt werden, die auch bezahlt wurde, alles andere ist illegal und übt eine denkbar schlechte Vorbildfunktion auf die Schüler aus. Gefördert wird dies oft auch durch völlig falsche Vorstellungen vom Begriff „Shareware“ — die Bezahlung ist dort kein „Kann“, sondern ein „Muss“!

- Die Open-Source-Software steht sehr häufig für alle gängigen Betriebssysteme zur Verfügung, die Schule (gemeint sind hier Lehrer und Schüler, in der Schule wie zu Hause) sind frei in der Wahl ihres favorisierten Betriebssystems und können doch alle mit der gleichen Anwendungssoftware arbeiten.
- Open-Source und Softwarepatente passen schlecht zusammen, ja sind im Grunde Gegenpole. Dieser Aspekt wird häufig vernachlässigt. Der gesellschaftspolitische Aspekt der Softwarepatente mitsamt aller seiner Auswirkungen macht aber nicht vor der Schule Halt und sollte sogar dort thematisiert werden, z.B. im Gemeinschaftskundeunterricht. Die Konsequenz könnte sein, möglichst wenig bis keine Software einzusetzen von Herstellern, die Softwarepatente befürworten. Weitreichende Möglichkeiten der Patentierung von Software können eine große Gefahr für Open-Source-Projekte bis hin zu deren Tod bedeuten, als Folge nicht nur die Schule praktisch ausschließlich auf kommerzielle Software angewiesen wäre (bis hin zur Abhängigkeit von Monopolen) — alle oben genannten Vorteile wären praktisch auf immer verloren.

Diese offenkundigen Vorteile bedeuten nun aber nicht automatisch, dass allgemein dem endlich kommenden Einsatz von Open-Source-Software entgegengefiebert würde; es gibt durchaus auch Widerstände, die hier auch genannt werden sollen:

- Ein oft gehörtes Argument gegen den Einsatz von Open-Source-Software in der Schule ist: „In der Wirtschaft wird auch nur Software des Herstellers XYZ eingesetzt, da müssten sich die Schüler später nur mühsam umgewöhnen.“
Hier stellt sich sofort die Frage, welches Ziel die Schule denn erreichen will: Soll eine konkrete Produktschulung, z.B. mit der Textverarbeitung X, durchgeführt werden, oder sollen die Schüler in den Stand versetzt

werden, mit der Produktkategorie „Textverarbeitung“, unabhängig vom konkreten Hersteller, umgehen zu können? Es ist offensichtlich, dass letzteres das höherwertigere Ziel ist und dieses bevorzugt angestrebt werden sollte. (Schließlich lernt auch jeder Fahrschüler nicht „Fahren mit Auto des Modells A von Hersteller B“ sondern einfach nur „Auto fahren“ und kommt später prinzipiell mit jedem Auto zurecht.) Aus diesem Grund ist eine Fixierung auf ein bestimmtes Produkt eines bestimmten Herstellers ohnehin nicht sinnvoll und nicht erforderlich — es spricht daher nichts gegen den Einsatz eines kommerziellen Produktes aber genau so wenig gegen den Einsatz eines Open-Source-Produktes.

- Derzeit ist der überwiegende Teil des von Verlagen angebotenen Materials zur Vorbereitung und Durchführung des Unterrichts auf kommerzielle Software ausgerichtet und damit nur bedingt bis gar nicht verwendbar, wenn eben nicht diese kommerzielle Software in der Schule eingesetzt wird, sondern stattdessen vergleichbare Open-Source-Programme. Eine ausführliche Betrachtung dieses Sachverhaltes findet sich in Abschnitt 4.
- Recht häufig haben Lehrpersonen bereits Kenntnisse in der Bedienung kommerzieller Programme. Steigt die Schule nun auf äquivalente Open-Source-Produkte um, so ist ein gewisser Aufwand für die Einarbeitung in die neue Software erforderlich, die Widerstände dagegen angesichts ohnehin laufend zunehmender Belastungen sehr verständlich. Dies kann abgefangen werden durch entsprechende Angebote an Unterrichtsmaterialien (siehe Abschnitt 4) und durch geeignete Fortbildungen (siehe Abschnitt 5).
- Der überwiegende Teil von Fortbildungen im IT-Bereich für Lehrer orientiert sich derzeit nicht an Open-Source-Programmen oder ist im Idealfall unabhängig von einem bestimmten Programm, sondern basiert auf kommerziellen Produkten. Eine ausführliche Betrachtung dieses Sachverhaltes findet sich in Abschnitt 5.

Die Politik nimmt erkennbar immer stärker den Open-Source-Bereich war und damit auch die dort vorhandenen Vorteile gegenüber den bisherigen Lösungen, selbstverständlich aber auch die Problemfelder. Im Bereich der Wirtschaft gibt es bereits ganz beachtliche Entwicklungen bei der Förderung von Open-Source-Software, stellvertretend für alle seien hier beispielhaft genannt:

- Migrationsleitfaden des Bundesinnenministeriums: <http://www.kbst.bund.de/download/Migrationsleitfaden-Version-2-20050908-grob.pdf>
- bwcon-SIG Open Source (Baden-Württemberg Connected Special Interest Group Open Source): <http://www.bwcon.de/sixcms/detail.php?id=31290>
- bwcon:boss (Baden-Württemberg Open-Source-LayerS): <http://www.bwcon.de/boss/>

- Bericht des Rechnungshofes Bayern: <http://www.orh.bayern.de/Jahresbericht2001.pdf>

Aber nicht nur Bund und Länder interessieren sich zunehmend für Open-Source, sondern auch die Schulträger. Für letztere ist Open-Source u.a. eine Möglichkeit zur Kosteneinsparung im eigenen Verwaltungsbereich (siehe z.B. Schwäbisch-Hall und München), aber ebenso auch für die Schulen in ihrem Bereich. Schulen sind heute meist budgetiert und die Budgets werden eher kleiner als größer. Vor diesem Hintergrund müsste jede Schule ein großes Interesse daran haben, wo auch immer möglich Open-Source-Software einzusetzen, müssen aus dem chronisch knappen Budget auch viele viele andere dringend notwendige Dinge bezahlt werden.

Warum aber ist Open-Source-Software häufig schlicht nicht bekannt? Das hat sicherlich viele Ursachen, drei davon seien hier genannt:

- Durch Werbung wird kommerzielle Software beworben, aber keine Open-Source-Software. Kommerzielle Softwarehersteller müssen werben, um ihre Produkte bekannt zu machen und zu verkaufen, dafür steht ihnen auch ein entsprechender Werbeetat zur Verfügung. Open-Source-Software hat in der Regel keine Werbemittel zur Verfügung, wird nicht beworben und die Kenntnis von leistungsfähigen Open-Source-Programmen hat etwas von Insider-Wissen — der typische Lehrer jedenfalls kennt solche Software eher nicht.
- Die Verlage für Lehr- und Lernmittel konzentrierten sich bei ihren Unterrichtsmaterialien hauptsächlich auf kommerzielle Software: Wer diese Unterrichtsmaterialien erwirbt und gewinnbringend nutzen will, benötigt die entsprechende kommerzielle Software dazu. Des weiteren haben solche Unterrichtsmaterialien natürlich auch stark werbenden Charakter: Es wird damit die Botschaft „Wenn ein Verlag ein Unterrichtswerk basierend auf einer ganz bestimmten Software herausgibt, dann muss diese Software ja wohl gut sein.“ transportiert.
- Die Lehrerfortbildung im IT-Bereich konzentrierte sich meist auf kommerzielle Software, wer die auf einer solchen Fortbildung erworbenen Kenntnisse, Fähigkeiten und Fertigkeiten gewinnbringend nutzen will, benötigt in der Schule die entsprechende kommerzielle Software. Auch hier existiert ein stark werbender Charakter: Es wird die Botschaft „Wenn eine ganze Lehrerfortbildung auf einer ganz bestimmten Software basiert, dann muss diese Software offenbar gut sein.“ damit transportiert.

Diese und noch weitere, hier nicht genannte Gründe, sind natürlich nicht isoliert zu betrachten, sondern haben wechselseitige Abhängigkeiten voneinander.

3 Bildungspläne

Open-Source kann für die Schule sicherlich kein Selbstzweck sein, einfach nur damit Open-Source-Software um ihrer selbst willen eingesetzt wird. Die Bildungspläne geben die zu erreichenden Ziele vor und die Schule vor Ort wählt

dann aus, mit welcher Software sie diese Ziele zu erreichen gedenkt — genau so wie bei den Schulbüchern. Die Bildungspläne sollten daher immer so gehalten sein, dass keine konkrete einzusetzende Software oder Programmiersprache oder Entwicklungsumgebung fest vorgegeben wird, sondern nur das allgemeine mit einer bestimmten Softwarekategorie zu erreichende Ziel. Damit ist dann weder kommerzielle Software noch Open-Source-Software bevorzugt oder benachteiligt, die Schule vor Ort kann dann selbst an Hand von Kriterien wie in Abschnitt 2 genannt entscheiden, in welcher Situation sie welche Software einsetzen will.

Immer mehr Bundesländer gehen dazu über, die wichtigen Abschlussprüfungen der verschiedenen Schularten der besseren Vergleichbarkeit der Ergebnisse wegen zentral zu organisieren, am Gymnasium beispielsweise wäre dies das Zentralabitur. Dies hat aber auch zunehmend Auswirkungen auf die an den Schulen eingesetzte bzw. einsetzbare Software, derzeit hauptsächlich erst dort, wo unmittelbar informatische Inhalte abgeprüft werden, dies kann aber durchaus immer breiter werden. Zentral gestellte Prüfungsaufgaben bringen automatisch mit sich, dass aus Gründen der Chancengleichheit auch an allen Schulen dieselbe Software eingesetzt wird. Die Schulen sind in dieser Situation dann nicht mehr frei in der Entscheidung, welche Software sie einsetzen wollen, sondern sie müssen die im Hinblick auf die Durchführbarkeit der zentralen Prüfungen gemachten Vorgaben einhalten.

Bei der Erstellung von Bildungsplänen ist damit die Idealsituation, keine konkreten Angaben über die einzusetzende Software zu machen, nicht mehr aufrecht zu halten. Die Entscheidung über die zu verwendende Software sollte mit großer Umsicht getroffen werden, da sie sehr weitreichend ist und das für alle Schulen des ganzen Bundeslandes — eine große Verantwortung! Einige gute Argumente sprechen dafür, dass in den Bildungsplänen bevorzugt Open-Source-Software an Stelle kommerzieller Software genannt werden sollte — immer unter der Voraussetzung, dass mit beiden die zu erreichenden Ziele auch erreicht werden können:

- Den Schulen bzw. Schulträgern bleiben erhebliche Kosten erspart, die oft auch gar nicht notwendig sind.
- Schüler wie Lehrer können kostenfrei und legal auch zu Hause mit derselben Software arbeiten — dies wird immer wichtiger im Zeitalter von E-Learning, Online-Learning, usw..
- Sehr viel Open-Source-Software ist für alle derzeit gängigen Betriebssysteme verfügbar, Schule wie Schüler und Lehrer werden damit auch frei in der Wahl ihres Betriebssystems, bei der Entscheidung für ein Open-Source-Betriebssystem ist dies eine weitere größere Einsparung der knappen Mittel.

Was die Kostensituation angeht würde die Entscheidung für Open-Source-Software große Einsparungen landesweit bedeuten.

4 Unterrichtsmaterialien

Damit nicht jeder Lehrer immer wieder neu das Rad erfinden muss bieten viele spezialisierte Verlage fertige Unterrichtsmaterialien an, z.B. in Form von Stundenentwürfen, Kopiervorlagen, Arbeitsblättern, usw. Diese Materialien beziehen sich derzeit leider zum weit überwiegenden Teil auf kommerzielle Software; Lehrer, an deren Schule vor allem kommerzielle Software eingesetzt wird, können sich mit Hilfe dieser Materialien spürbar bei der Unterrichtsvorbereitung entlasten, Lehrer, an deren Schule so weit wie möglich mit OSS gearbeitet wird können davon kaum profitieren. So kann der Einsatz von OSS in der eigenen Schule durchaus als Strafe empfunden werden, da alle notwendigen Unterrichtsmaterialien selbst erstellt werden müssen.

Selbstverständlich handelt es sich hierbei ein Stück weit um das „Henne-Ei-Problem“: Die Verlage würden ja Materialien auf OSS basierend produzieren, wenn nur genügend Schulen auch OSS einsetzen - viele Schulen würden ja OSS einsetzen, wenn nur die Verlage ausreichend Unterrichtsmaterialien dafür im Angebot hätten. Ganz langsam ändert sich dies und die ersten Verlage bieten nun auch Materialien für die Informationstechnische Grundbildung und Informatik an, die sich schwerpunktmäßig auf OSS beziehen.

Das Vorhandensein solcher Materialien könnte sicherlich den einen oder anderen Lehrer dazu bewegen, seinen Unterricht mit Hilfe von OSS zu gestalten, da er nicht unnötig viel Zeit mit der Unterrichtsvorbereitung zubringen müsste. Je mehr Lehrer an einer Schule OSS einsetzen, desto eher ist eine vollständige Umstellung an dieser Schule zu erwarten, da es dann einfach keinen Sinn mehr macht, noch Lizenzkosten für kommerzielle Software zu bezahlen, wenn sie nur noch vereinzelt genutzt wird, des weiteren hat es auch Vorteile, wenn alle Lehrer derselben Fächer auch mit derselben Software arbeiten. Genau dies ist natürlich anfangs der Hemmschuh, wenn die ersten damit beginnen, auf OSS umzuschwenken — damit stören sie die Einheitlichkeit.

Verlage produzieren das, wovon sie annehmen, dass es gekauft wird; derzeit nehmen sie offenbar überwiegend an, es wird dann gekauft, wenn die marktgängige kommerzielle Software die Basis der Unterrichtsmaterialien ist. Wie groß die Nachfrage nach Alternativen ist — Unterrichtsmaterialien auf Basis von Open-Source-Software — ist vermutlich gar nicht bekannt bzw. der Bedarf wird gestillt durch kostenfreie Angebote im WWW, die gerade was Open-Source-Software angeht besonders zahlreich und umfangreich sind. Hier gibt es vermutlich ein Marktpotenzial, das die meisten Verlage noch gar nicht entdeckt haben. Hilfreich könnte sein, wenn möglichst viele Lehrkräfte, die Unterrichtsmaterialien auf Open-Source-Basis suchen, dies auch den Verlagen mitteilen — nur so können letztere mit der Zeit erkennen, wie groß die Nachfrage ist.

Einen starken Schub bei der Produktion von Materialien auf Basis von Open-Source-Software würde sicherlich auslösen, wenn in Bildungsplänen im Falle dort Vorgaben über die zu verwendende Software gemacht werden, dies bevorzugt Open-Source-Software wäre, siehe dazu auch Abschnitt 3. Mehrkosten sind hier nicht zu erkennen, aber auch keine Einsparungen: Die Unterrichtsmaterialien würden sich nach wie vor auf demselben Preisniveau bewegen, weder Schulen

noch Schüler noch Lehrer könnten Geld sparen.

5 Fortbildung

Fortbildungen im IT-Bereich basieren zum überwiegenden Teil auf bekannter, marktgängiger, beworbener kommerzieller Software. Die Interessenten an Lehrerfortbildungen wählen teilweise die Fortbildung ja gerade aus diesem Grund aus, weil die ihnen bekannte Software dort verwendet wird. Die Fortbildungsangebote nehmen insofern nur die Nachfrage auf und befriedigen sie. Meist wären die Interessenten aber gar nicht auf eine bestimmte Software fixiert, sondern sie haben eher Interesse an einer bestimmten Kategorie, z.B. „Bildbearbeitung“. Die Wahl einer bestimmten — eben meist kommerziellen — Software hat seinen Ursprung dann oft nur im Bekanntheitsgrad und dieser ist beim Nicht-Insider bei kommerzieller Software ungleich höher als bei Open-Source-Software, die deutlich unter ihrem systembedingten Werbedefizit leidet.

Ein weiterer Grund ist bei den Referenten zu sehen: Diese setzen in ihren Fortbildungen natürlich nur diejenige Software ein, die sie selber nutzen und gut kennen. Die Referenten aber sind meist Fachleute im Bereich der Pädagogik und Didaktik und nicht Insider im IT-Bereich. Auch ihnen ist daher vorwiegend die marktgängige, weil massiv beworbene kommerzielle Software bekannt und nicht die Open-Source-Software, die vergleichbares leisten würde.

Die Wahl der in Fortbildungen verwendeten Software hat einen starken Einfluss auf die Wahl der Software, die in den Schulen eingesetzt wird: Teilnehmer von Lehrerfortbildungen wollen selbstverständlich die auf der Fortbildung erhaltenen Fähigkeiten und Fertigkeiten auch im praktischen Alltag nutzen und benötigen dazu eben die auf der Fortbildung kennengelernte Software, „diese muss ja gut sein, sonst wäre sie nicht in einer Fortbildung verwendet worden.“ Dabei sind die meisten gar nicht fixiert auf bestimmte Produkte und Marken, es ist häufig unwichtig, welches konkrete Produkt es ist, Hauptsache ist es erfüllt die gestellten Anforderungen der entsprechenden Kategorie, z.B. „Bildbearbeitung“. Fand die Fortbildung auf Basis einer Open-Source-Software statt, dann ist es gar kein Problem, diese in der Schule einzusetzen, denn es entstehen ja keine Kosten, langwierige Etatberatungen und Beschaffungsverfahren entfallen — die Kosten für notwendige Updates dürfen dabei nicht unterschätzt werden. Fand die Fortbildung dagegen auf Basis einer kommerziellen Software statt, dann besteht natürlich der starke Wunsch, diese Software auch in der Schule einsetzen zu können: Es entstehen dann Lizenzkosten für viele Arbeitsplätze in der Schule, es entsteht ein unsichtbarer Druck auf die Kollegen, auch zu Hause aus Gründen der Einheitlichkeit auch mit dieser Software zu arbeiten, die sich diese dann selbst kaufen müssen und schlussendlich gilt dies eingeschränkt auch für die Schüler. Die Wahl der Software, die in einer Fortbildung verwendet wird, kann daher an vielen Schulen große Folgekosten nach sich ziehen.

Es wäre der Verbreitung von Open-Source-Software an Schulen sicherlich sehr zuträglich, wenn Fortbildungen vermehrt Open-Source-Software als Basis einsetzen würden. Dies kann geschehen in dem kurz und überblicksartig aus

möglichst vielen Bereichen für den Schulalltag geeignete Open-Source-Software vorgestellt wird. Die Erfahrung zeigt, dass die Teilnehmer den größten Teil dieser Software gar nicht kennen, aber sehr dankbar sind für solche Informationen und diese Software dann gewinnbringend für sich einsetzen können. (Beispiele sind Open Office als Office-Paket, Gimp als Bildbearbeitung, Dia als Diagrammsoftware, SnipSnap als kleines Wiki zum Mitnehmen, Eclipse als Entwicklungsumgebung, GanttProject als Projektplaner z.B. für den Stoffverteilungsplan, ...)

Einen sehr starken Schub Fortbildungen auf Basis von Open-Source-Software anzubieten würde sicherlich auslösen, wenn in Bildungsplänen, so dort überhaupt Vorgaben zur verwendenden Software gemacht werden, dies bevorzugt Open-Source wäre.

Fortbildungen sind teuer — an diesen Kosten ändert sich aber nichts, gleich welche Software dort eingesetzt wird. Einen sehr großen Kostenunterschied aber macht es bei den Folgekosten aus, wenn die Fortbildungsteilnehmer hinterher im Schulalltag mit der auf der Fortbildung kennengelernten Software weiterarbeiten wollen, wie oben erläutert.