

Werkzeugkasten Freie Software



Materialien für den Unterricht



**Medien
in die
Schule**

Impressum

Titel

Werkzeugkasten Freie Software
im Projekt »Medien in die Schule«
– Materialien für den Unterricht –

Herausgeber

FSM

Freiwillige Selbstkontrolle
Multimedia-Diensteanbieter e.V.
Beuthstraße 6
10117 Berlin
030 / 24 04 84 30
<http://fsm.de>

fsf FREIWILLIGE
SELBSTKONTROLLE
FERNSEHEN

Freiwillige Selbstkontrolle
Fernsehen e.V.
Am Karlsbad 11
10785 Berlin
030 / 23 08 36 20
<http://fsf.de>

Google

Google Germany GmbH
Unter den Linden 14
10117 Berlin
<http://google.de>

Unterstützer



**1. Auflage –
Dezember 2016**

 **creative
commons**



Textsatz und Produktion: racken GmbH <http://racken.de>

Illustrationen: Tamim Sibai

Bildnachweise: Foto Volker Grassmuck: Berit Schuck, Open Content License

Vervielfältigung und Verbreitung ist unter Angabe der Quelle (Titel,
Herausgeberschaft sowie Auflage) erlaubt. Weitere Informationen:
<http://creativecommons.org/licenses/by-sa/4.0/deed.de>

Es wird darauf hingewiesen, dass trotz sorgfältiger Bearbeitung und Prüfung alle
Angaben ohne Gewähr erfolgen. Eine Haftung der Herausgeber ist ausgeschlossen.

<http://medien-in-die-schule.de>

Inhaltsverzeichnis

1 Einleitung 09

- 09 Zum Verständnis: Hardware, Software und Co.
- 10 Das Betriebssystem
- 11 Software im Netzwerk
- 11 Der Quellcode
- 12 Installation von Software
- 13 Hinweise

2 Was ist Freie Software? 15

- 16 Andere Begriffe für Freie Software
- 17 Software-Modelle im Vergleich
- 17 Rechte und Rechtliches
- 18 Freie Software
- 18 Proprietäre Software

3 Philosophie 21

- 21 Wissenschaft
- 22 Hippies und Hacker
- 23 Software wird zur Ware
- 23 Von UNIX zu LINUX
- 25 Wissensallmende
- 26 Warum machen die das?

4 Freie Software in der Bildung 29

- 29 Wirtschaftlichkeit und Soziales
- 30 Rechtssicherheit
- 30 Bezugsquellen
- 31 Abspaltungen
- 31 Unabhängigkeit
- 32 Übereinstimmende Ideale
- 32 Flexibilität
- 33 Freie Software als Werkstoff

35	Präsentation und Textverarbeitung
35	Das Office-Programmpaket LIBREOFFICE
36	LIBREOFFICE WRITER
37	LIBREOFFICE IMPRESS
37	LIBREOFFICE CALC
38	LATEX

38	Bildbearbeitung und Grafik
39	BLENDER
40	DIA
40	GIMP
41	INKSCAPE
41	KRITA
42	LIBRECAD
42	LIBREOFFICE DRAW
43	PINTA
43	RAWTHERAPEE
44	SYNFIG
45	SCRIBUS

45	Audio und Video
47	AUDACITY
48	AVIDEMUX
49	KODI
49	MEDIATHEKVIEW
50	SHOTCUT
50	VLC

51	Internet und Kommunikation
51	BLUEFISH
51	CHROMIUM/CHROME
52	FIREFOX (alle Fächer)
52	THUNDERBIRD

53	Unterrichtsunterstützung und Fächerspezifisches
53	EXEARNING (alle Fächer)
54	FREE MIND (alle Fächer)
54	GCOMPRIS (Grundschule)
55	OPEN-SANKORÉ (alle Fächer)
55	PAUKER (alle Fächer)
56	ENERGIE TYCOON (Simulationsspiel Politik, Wirtschaft etc.)
56	GEOGEBRA (Mathematik)
57	JMOL (Chemie)
57	MARBLE (Geografie)
58	MUSE SCORE (Musik)
58	SCRATCH 2.0 OFFLINE EDITOR (Informatik)
59	Unterrichtsorganisation und Klassenmanagement
59	GRADEMAN (für Lehrkräfte)
59	ITALC

6 **Werkzeugporträts** 61

61	GIMP
63	INKSCAPE
66	SCRIBUS
69	FIREFOX
71	LIBREOFFICE WRITER
74	AUDACITY
76	EXEARNING

7 **Glossar** 80

Autoren



VOLKER GRASSMUCK

ist Mediensoziologe, freier Autor und Aktivist.

Er hat über die Wissensordnung digitaler Medien, Urheberrecht und Wissensallmende geforscht. Neben seiner Publikation **FREIE SOFTWARE ZWISCHEN PRIVAT- UND GEMEIN-EIGENTUM** hat er die Konferenzserie **WIZARDS-OF-OS.ORG** und das Information-sportal zum Urheberrecht **IRIGHTS.INFO** geleitet.

Er bloggt unter <http://vgrass.de>.



DANIEL ROHDE-KAGE

ist Lehrer für Technik, Informatik und Biologie an einer niedersächsischen Oberschule. Er arbeitete drei Jahre in der Medienberatung des Niedersächsischen Landesinstituts für schulische Qualitätsentwicklung mit den Arbeitsschwerpunkten Open-Source-Software, Creative-Commons-Medien, freie Betriebssysteme und Open Educational Resources. Privat ist er engagierter **LINUX**-Nutzer seit **OPENSUSE 9.0** und Mitarbeiter informatik verschiedenen freien Projekten.



BJÖRN SCHIESSLE

hat Informatik studiert und im Bereich der verteilten künstlichen Intelligenz und der Robotik geforscht. Gesellschaftliche und politische Auswirkungen von Software haben ihn immer interessiert. Seit 2015 ist er als stellvertretender Deutschland-Koordinator der **FSFE** aktiv. Auch hauptberuflich entwickelt er ausschließlich Freie Software. Weitere Informationen findet man auf seiner Homepage

<http://schiessle.org>.



STEFAN SCHOBER

unterrichtet seit 25 Jahren Informatik und Chemie, zeitweise auch in der Erwachsenenbildung. Seine Interessenschwerpunkte liegen im Bereich Softwareprojektmanagement sowie der Schnittstelle von Soft- und Hardware. Aktuell ist er Teil der Schulleitung der Friedensburg-Oberschule, Berlin.



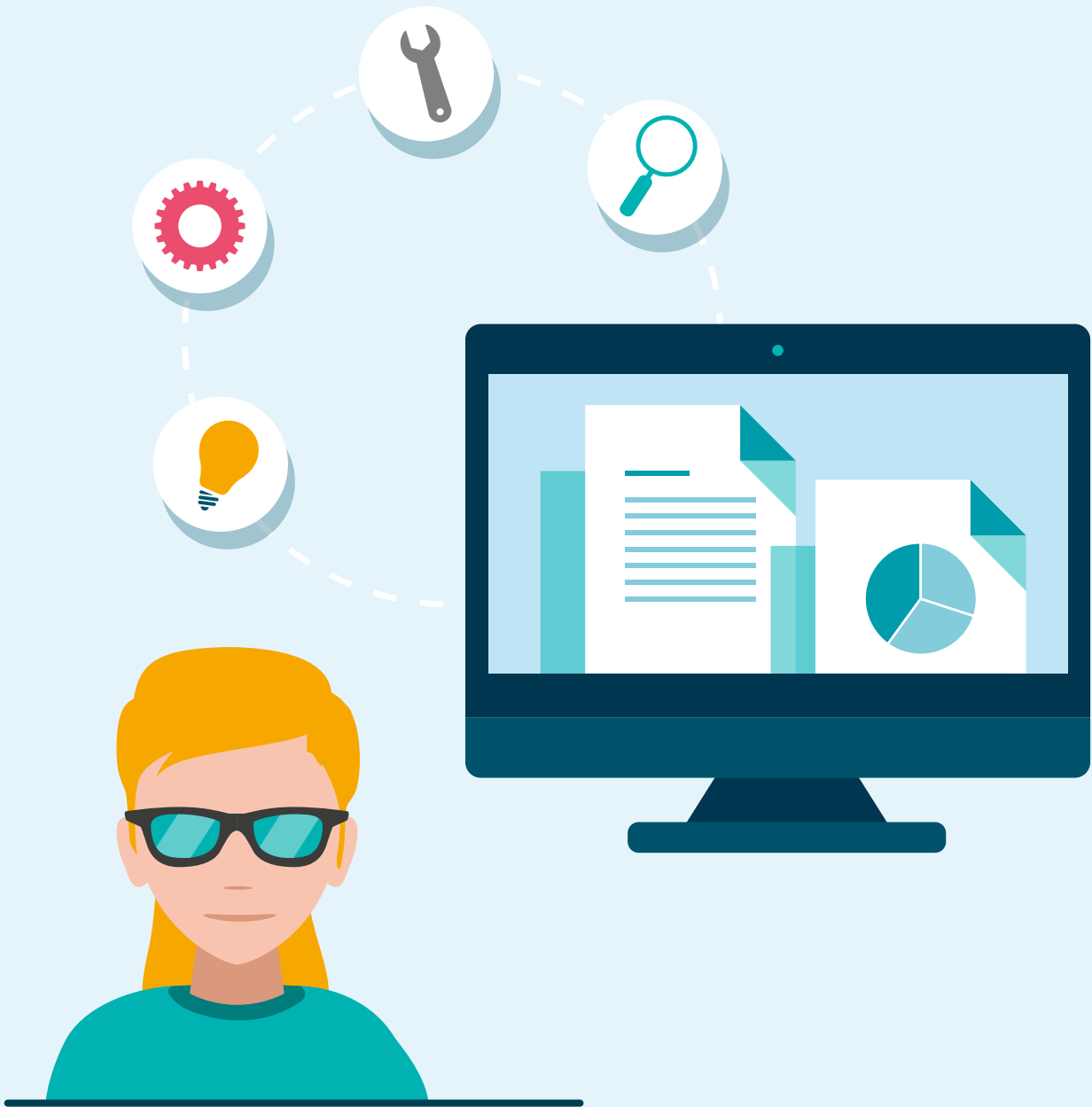
SEBASTIAN SEITZ

ist Diplom-Pädagoge und arbeitet als Projektmanager für die Technologiestiftung Berlin. Er interessiert sich besonders für die Zusammenhänge von Open Source und Bildung und nutzt seit zehn Jahren LINUX. Er bloggt unter <http://s-seitz.de> und ist auf Twitter als [@bastiseitz](https://twitter.com/bastiseitz) zu finden.



WOLF-DIETER ZIMMERMANN

pensionierter Leiter des Studienseminar Neuss, Autor verschiedener Bücher u.a. zum Thema Lehrerausbildung, nebenberuflich Kabarettist. Fellow der FSFE. In Mülheim Ruhr aktiv beim Thema Nachhaltigkeit im Alltag (Träger des Klimaschutzpreises 2016).



EINLEITUNG

Dieser Werkzeugkasten ebnet den Einstieg in das Thema »Freie Software« in der Schule und stellt einige für die schulische Arbeit geeignete Werkzeuge vor, sodass Lehrkräfte in kurzer Zeit entscheiden können, ob und wie diese ihnen bei ihrer Arbeit helfen können. Die Auswahl erhebt natürlich keinen Anspruch auf Vollständigkeit, präsentiert jedoch eine Übersicht über Software zu allen schulisch relevanten Bereichen und porträtiert detailliert und praxisbezogen sieben wichtige Werkzeuge samt pädagogischer Einschätzung.

Die Projektwoche steht an und die Schüler*innen haben eine Idee, für die Software nötig ist. Da das Thema für manche Lehrkraft Neuland ist, stellt sich die Frage, welches Programm eingesetzt werden soll. Förderlich wäre es, wenn jede Schülerin und jeder Schüler die Software – auch zu Hause und auf den unterschiedlichen Systemen – dauerhaft verwenden kann. Dabei möchte man rechtliche Probleme vermeiden. Viel Geld zum Kauf von Software ist nicht da, dafür aber das Unbehagen bei dem Gedanken, die Schüler*innen für ein kostenpflichtiges Produkt einer kommerziellen Firma zu schulen.

In dieser Situation spricht vieles für den Einsatz von Freier Software. Ein weiterer Vorteil: Freie Software kann auch als Philosophie verstanden werden; als praktisches Beispiel dafür, dass aus vielen individuellen Beiträgen ein großer Nutzen für die Allgemeinheit entstehen kann. Dies ist ein Aspekt, der ebenso im Unterricht angesprochen werden kann.

Zum Verständnis: Hardware, Software und Co.

Man stelle sich vor, man wolle Tango tanzen lernen. Dann studiert man die zugehörige Bewegungsfolge (das »Programm«) ein, die die Beine (die »Hardware«) später nachvollziehen: Wiegeschritt – Promenade – Linksdrehung etc. Sehr ähnlich ist die Situation bei einem Computer. Man muss ihm beibringen,

1 Ausführliche Informationen zum Making im schulischen Kontext inkl. detaillierter Beschreibungen von Tools und Projekten finden sich im Werkzeugkasten »DIY und Making – Gestalten mit Technik, Elektronik und PC« unter <http://medien-in-die-schule.de/werkzeugkasten/werkzeugkasten-diy-und-making>

wie er das jeweilige Problem (»Tango tanzen«) systematisch lösen soll: Wenn dies, dann tue das etc. Diese Anleitung bezeichnet man als »Software«. Das Gerät »Computer« kann im Gegensatz zur Software angefasst werden und wird daher als »Hardware« bezeichnet (von ursprünglich *hardware* = Eisenwaren). Das Anpassen von Hardware nennt man übrigens Hacking und die gerade populäre Maker-Bewegung¹ hat dazu das Motto »If you can't hack it, you don't own it«.

Der Weg des Do-it-Yourself ist prinzipiell gangbar, aber sehr mühselig: Man müsste den Computer selber programmieren. Viel einfacher ist es, ihn »zur Schule zu schicken«. Dann übernehmen Programmierer*innen die Aufgabe und schreiben eine Anleitung – das Programm. Da Programme ohne Aufwand vervielfältigt werden können, muss die Anleitung nur einmal geschrieben werden und die Kosten verteilen sich fair auf alle Programmbenutzer*innen. Einige Programmierer*innen schreiben ihre Programme in der Freizeit und verzichten auf manche ihrer Rechte. Diese Programme darf jeder benutzen – einfach so. Das ist ideal für den Unterricht: Die Schüler*innen können die Software überall verwenden; Die Lizenzbedingungen lassen dies explizit zu.

Jedes Programm muss also sowohl zum Chip als auch zum Betriebssystem passen. So kann z.B. eine tolle Anwendung für einen Apple-Computer nicht auf einem Windows-Computer starten.

Das Betriebssystem

Verschiedene Computertypen »sprechen« unterschiedliche »Muttersprachen«. Entscheidet sich ein Hersteller für einen bestimmten CPU-Typ (das Gehirn des Computers), so kann dieser Computer nur Programme in der Maschinensprache (= Muttersprache) dieses speziellen Typs verstehen. Ein Computer mit einem anderen CPU-Typ kann dieses Programm nicht ausführen.

Zudem geben die Hersteller ihren Computern meist eine spezielle Software mit, welche sich selbstständig um die technischen Abläufe wie Suchen, Löschen, Drucken etc. kümmert. Diese Programme heißen Betriebssysteme (engl. *operating system*, kurz OS). Bekannt sind LINUX, WINDOWS, ANDROID und **OS X**².

Bei Windows-Systemen gibt es von jeder Version zwei leicht unterschiedliche Typen: 32-Bit und 64-Bit. Der Unterschied liegt darin, dass 64-Bit-Programme mehr Daten speichern können. Neuere Programme gibt es immer öfter nur noch als 64-Bit-Version, die nicht in einem 32-Bit-Betriebssystem funktionieren. Umgekehrt laufen aber ältere 32-Bit-Programme in 64-Bit-Betriebssystemen;

2 Mit der Version 10.12 wurde OS X in »MAC OS« umbenannt

sie sind abwärts-kompatibel. Jedes Programm muss also sowohl zum Chip als auch zum Betriebssystem passen. So kann z.B. eine tolle Anwendung für einen Apple-Computer nicht auf einem Windows-Computer starten.

Software im Netzwerk

Wichtig ist, ob ein Programm alle Daten lokal – also in dem Rechner, an dem man arbeitet – speichert. Oft werden nämlich Programmteile oder die Hilfetexte auf einem Server gespeichert. Werden diese Daten benötigt, so muss sich das Programm erst mit dem Server verständigen. Ist jedoch die Verbindung gestört oder der Server »offline«, so können die Daten nicht abgerufen werden. Dann kann das Programm nicht genutzt werden. Findet diese Kommunikation über das Internet statt, so muss der Computer bei jedem Programmstart online gehen und überträgt dabei Informationen.

Bei ► *Client*-Programmen (oft auch als Webanwendungen bezeichnet) wird das Programm zumeist nicht am Ort der Nutzer*innen ausgeführt, sondern läuft auf einem Server. Die Eingabedaten werden dorthin übertragen und das Ergebnis zurückgeliefert. Ein Beispiel sind Suchmaschinen: Man gibt lokal seine Suchanfrage ein, diese wird über das Internet zu einem Server übertragen, der sie ausgewertet und eine Trefferliste liefert. Dabei werden Daten an Dritte übertragen und auf fremden Computern verarbeitet und ggf. gespeichert. Wenn Daten den eigenen Computer nicht verlassen sollen, dürfen nur lokale, also auf dem eigenen Rechner installierte Programme verwendet werden.

Der Quellcode

Computer verstehen nur einfache Befehle wie »addiere« oder »vergleiche« – die Maschinensprache. Deshalb ist es eine anspruchsvolle Aufgabe, Computern alle Eventualitäten genauestens zu erklären. Dieses Problem löst man mit Programmiersprachen: Sie sind leichter zu lernen und die Anweisungen wesentlich komplexer als Maschinensprachen. Die Befehlsfolgen (»Tue erst dies, dann tue das« etc.) werden als Quellcode (engl. *source code*) bezeichnet. Dieser wird vom Computer in die gewünschte Maschinensprache übersetzt. So muss ein Programm nur einmal programmiert werden und wird dann für jedes Betriebssystem und jeden Chip in die zugehörige Maschinensprache übersetzt. Nur Dateien, die Maschinensprache enthalten, werden umgangssprachlich als Programm oder App (engl. *application*) bezeichnet, denn nur sie können (auf dem passenden Chip und dem richtigen Betriebssystem) ausgeführt werden.

Der Quellcode selbst kann nicht ohne weiteres vom Computer verstanden werden – wohl aber von anderen Programmierer*innen. Da im Code viel

Know-how steckt, wollen viele Entwickler*innen nicht, dass er eingesehen werden kann – sie halten ihn als Betriebsgeheimnis unter Verschluss. Einige Programmierer*innen stellen ihren Code hingegen frei zur Verfügung. Dann können andere daraus lernen, Codeteile in eigene Projekte übernehmen, Fehler verbessern und weitere Funktionen ergänzen. So wird der Quellcode unabhängig von seinen Ersteller*innen und kann quasi »ewig leben« und weiterentwickelt werden.



Installation von Software

Neue Programme müssen vor der ersten Benutzung installiert werden. Das bedeutet, dass sich das Programm und das Betriebssystem ein wenig aneinander anpassen. Dabei werden bestimmte Daten auf dem Computer gespeichert – bei Windows oft in einer Datenbank namens *registry*. Dort steht z.B., welche Dateitypen von dem Programm geöffnet werden.

Ganz anders bei »portabler Software«: Hier muss keine Anpassung vor dem ersten Start ablaufen. Das Programm kann ohne Installation direkt, z.B. von einem USB-Stick, gestartet werden. Das ist sehr praktisch, denn so kann man wichtige Programme immer dabei haben.

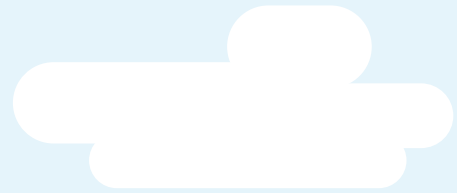
Hinweise

Vor der Erstverwendung eines Programms sollten ein paar Punkte bedacht werden: Woher beziehe ich die Dateien? Beim Herunterladen von unbekanntem Webseiten besteht die Gefahr, dass sie Malware enthalten. Da jeder den Quellcode Freier Software verändern kann, könnten auch unerwünschte Funktionen hinzugefügt worden sein. Deshalb sollte man Programme nach Möglichkeit nur von offiziellen Projektseiten oder renommierten Portalen laden. Gelegentlich bitten Programmierer*innen um eine Spende, lassen Werbung anzeigen oder verlangen einen Betrag für Programm.

Bevor das Programm auf Schulrechnern installiert wird, sollten einige Fragen geklärt werden: Wie lange dauert die Installation? Müssen Daten entpackt oder in einem besonderen Ordner gespeichert werden? Sollen und können die Schüler*innen dies selber tun? Lässt der Schulfilter den Download überhaupt zu? Welche Rechte werden vom Betriebssystem für die Installation benötigt? Ist ein Passwort nötig? Wird zusätzliche Software benötigt (z.B. ► *JAVA*)? Hierzu sollte im Vorfeld mit dem Administrator oder der Administratorin der Schule gesprochen werden. Eventuell gibt es dazu auch schulweite Vereinbarungen.

Mit der Klasse sollten Absprachen getroffen werden, wer wann was genau machen soll und welche Dinge zu unterlassen sind. Da die Schüler*innen unterschiedlich schnell arbeiten, müssen zusätzliche Aufgaben und ergänzende Hilfestellungen vorgesehen werden, um Frust zu vermeiden.

Nicht zuletzt ist der Datenschutz zu beachten. Welche Daten überträgt das Programm? Für personenbezogenen Daten wie Namen, Anschriften, Geburtsdaten und Noten gelten je nach Bundesland unterschiedliche Datenschutzbestimmungen, die bei der Speicherung und Verarbeitung berücksichtigt werden müssen. Hinweise bekommen Lehrkräfte bei den Schulverwaltungsbehörden und den Datenschutzbeauftragt*innen der Länder.



WAS IST FREIE SOFTWARE?

Der Begriff Freie Software (engl. *free software*) wurde 1986 von der FREE SOFTWARE FOUNDATION (FSF) geprägt. Freie Software bietet für Anwender*innen wie für Entwickler*innen eine Reihe technischer und praktischer Vorteile. Im Kern geht es bei Freier Software um die Freiheit der Anwender*innen, und nicht um den Preis. Im Kontext von Freier Software sind dabei konkret folgende vier Freiheiten gemeint:

1. Die Freiheit, das Programm für jeden Zweck zu verwenden.

Anwender*innen dürfen nicht davon abgehalten werden, das Programm auszuführen – egal für welchen Zweck und egal auf welchem Rechner. Einschränkungen wie »nur für den schulischen Gebrauch« sind bei Freier Software nicht zulässig.

2. Die Freiheit, das Programm zu untersuchen und an eigene Bedürfnisse anzupassen.

Um dieses Recht ausüben zu können, müssen Anwender*innen zusätzlich zu dem ausführbaren Programm auch den ► *Quellcode* erhalten. Nur so ist es möglich, die Funktionsweise zu untersuchen und die Datenverarbeitung des Programms anzupassen. Die Verfügbarkeit des Quelltextes ist eine notwendige Bedingung für Freie Software, macht alleine eine Software aber noch nicht frei.

3. Die Freiheit, Kopien des Programms weiterzugeben.

Anwender*innen können jederzeit beliebige Kopien des Programms anfertigen und weitergeben. Dies ermöglicht es z.B. Lehrer*innen, Software in den Unterricht mitzubringen und mit allen Schüler*innen zu teilen.

4. Die Freiheit, modifizierte Versionen der Software zu veröffentlichen.

Software ist nie fehlerfrei. Die Erlaubnis und Möglichkeit, modifizierte, verbesserte und in ihren Funktionen erweiterte Versionen von Programmen zu veröffentlichen, ist essentiell: So profitieren auch Menschen, die nicht programmieren können, von den Freiheiten Freier Software.

Enthält die **Lizenz** einer Software diese vier Freiheiten, dann spricht man von Freier Software. Dabei handelt es sich um Rechte, nicht um Pflichten. Alle Nutzer*innen von Freier Software können diese Rechte in Anspruch nehmen, sind aber nicht verpflichtet, vorgenommene Veränderungen zu veröffentlichen.

Bei Freier Software geht es nicht um den Preis: Dadurch, dass jeder die Software frei kopieren und weitergeben kann, ist diese oft kostenlos erhältlich. Dies ist aber keine Voraussetzung.

Es gibt viele Unternehmen, die sich auf die Unterstützung und Entwicklung von Freier Software spezialisiert haben. Ein Beispiel dafür, dass Freie Software nicht immer kostenlos sein muss, ist die Verfügbarkeit von freien Apps in unterschiedlichen App Stores: Häufig stellen Entwickler*innen eine freie App in App Stores kostenpflichtig zur Verfügung, bieten diese aber auch parallel in F-DROID an, einem auf Freie Software spezialisierten App-Store für kostenlose Apps. Auch in vorliegender Publikation wird mit GCOMPRIS eine Freie Software vorgestellt (**siehe Kapitel 5**), deren Windows-Version kostenpflichtig ist.

Andere Begriffe für Freie Software

Neben Freier Software haben sich weitere Bezeichnungen etabliert, um Software zu beschreiben, die den Anwender*innen die vier definierten Freiheiten gewährt. Der zweitälteste Begriff ist »Open Source«; er wurde 1998 zusammen mit der OPEN SOURCE INITIATIVE (OSI) ins Leben gerufen. Das primäre Ziel war es, eine Marketing-Kampagne für Freie Software zu initiieren, um Unternehmen davon zu überzeugen, Software unter einer freien Lizenz zu veröffentlichen.

Während sich Freie Software auf die Freiheit der Anwender*innen bezieht, hebt Open Source die Verfügbarkeit des Quelltextes hervor.

Beide Begriffe – Freie Software und Open Source – beschreiben bis heute die gleiche Gruppe von Software, betonen jedoch unterschiedliche Eigenschaften. Während sich Freie Software auf die Freiheit der Anwender*innen bezieht, hebt Open Source die Verfügbarkeit des Quelltextes hervor. In den folgenden Jahren wurden Kombinationen der beiden Begriffe bekannt. So entstand das Akronym »FOSS«, welches für »Free and Open Source Software« steht. Später wurde es zu »FLOSS« erweitert, als Abkürzung für »Free/Libre and Open Source Software«. Der Begriff »Libre« wurde eingeführt, um die Mehrdeutigkeit des englischen Wortes »free« aufzulösen, das gleichermaßen für »frei« als auch für »kostenlos« verwendet werden kann.

Software-Modelle im Vergleich

Wie von Georg Greve in WAS MACHT EIN FREIE-SOFTWARE-UNTERNEHMEN AUS?³ beschrieben, kann man bei Software zwischen der Verwendung als Software-Modell, Entwicklungsmodell und Geschäftsmodell unterscheiden.

Bezüglich des Software-Modells lassen sich zwei Arten unterscheiden: Freie Software und ▶ *proprietäre Software*. Dieser Aspekt betont die Rechte, die Software-Lizenzen Nutzer*innen einräumen.

Beide Arten von Software können sowohl in geschlossenen als auch offenen Gruppen, von einzelnen wie von vielen entwickelt werden. Obwohl Freie Software sehr gut für ein offenes Entwicklungsmodell geeignet ist, ist dies keine Voraussetzung.

Beim Geschäftsmodell geht es darum, wie mit Software Geld verdient wird. Software braucht nicht immer ein Geschäftsmodell, bei der Entwicklung von Freier wie von proprietärer Software stehen nicht immer kommerzielle Interessen im Vordergrund. Proprietäre Software, die kostenlos zur Verfügung gestellt wird, nennt man ▶ *Freeware* oder Shareware: Sie darf nicht mit Freier Software (engl. *free software*) verwechselt werden.

Die drei Felder (Software-Modell, Entwicklungsmodell und Geschäftsmodell) sind als Achsen eines dreidimensionalen Koordinatensystems vorstellbar, in dem Softwareprojekte positioniert und beschrieben werden können. Dabei können die Software-Modelle mit nahezu jedem Entwicklungs- und Geschäftsmodell verknüpft und kombiniert werden.

Manchmal wird auch bei proprietärer Software der Quelltext mitgeliefert, dieser darf aber nicht ohne Einschränkungen modifiziert und weitergegeben werden. Die Verfügbarkeit des Quellcodes stellt kein ausschlaggebendes Kriterium für die Unterscheidung von Freier und proprietärer Software dar.

3 GREVE, GEORG C. F. (2008): Was macht ein Freie-Software-Unternehmen aus? <https://fsfe.org/freesoftware/enterprise/freesoftwarecompany.de.html>

Rechte und Rechtliches

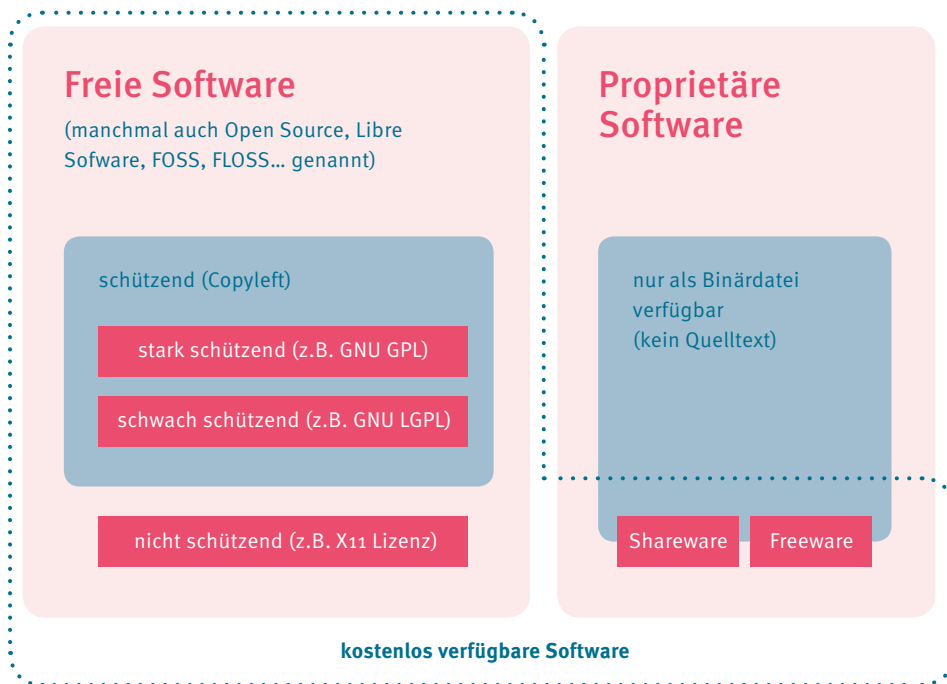
Freie Software bewegt sich nicht im rechtsfreien Raum, sie unterliegt wie jedes andere Werk dem Urheberrecht. Das bedeutet: Alle Rechte liegen bei den Autor*innen. Damit eine Software zu Freier Software wird, muss der Autor oder die Autorin das Werk unter eine Freie-Software-Lizenz stellen. Mittlerweile gibt es sehr viele Freie-Software-Lizenzen. Diese lassen sich in zwei Kategorien unterteilen:

1. Nicht schützende Lizenzen: Sie gewähren Anwender*innen die vier Freiheiten und stellen keine weiteren Bedingungen.

Software unter solchen Lizenzen kann später auch unter einer proprietären Lizenz vertrieben oder beliebig mit proprietärer Software kombiniert werden.

2. Schützende Lizenzen: Sie gewähren Anwender*innen die vier Freiheiten und verlangen, dass die Software nur unter den gleichen Bedingungen weitergegeben werden darf.

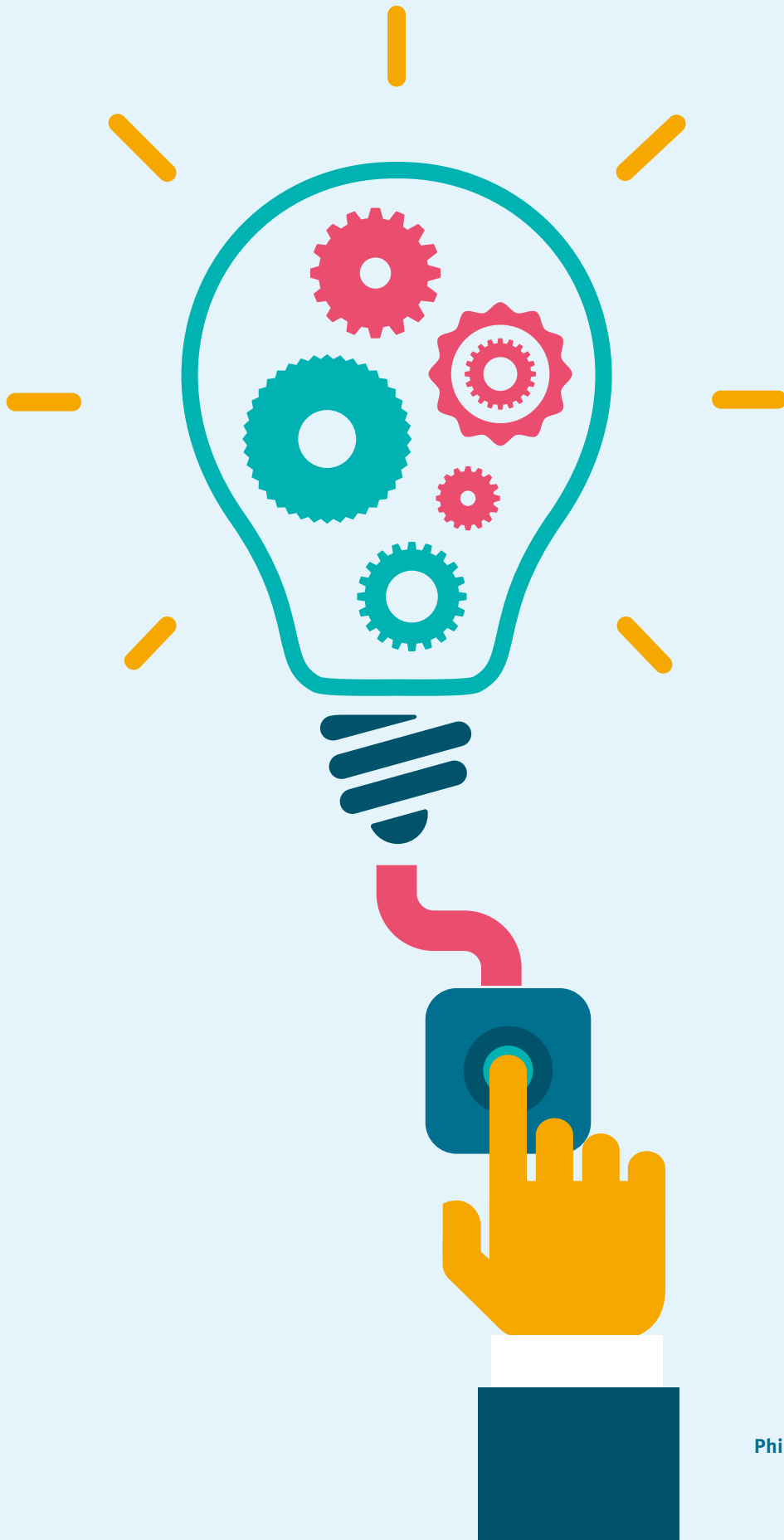
Diese Lizenzen sorgen dafür, dass ein einmal als Freie Software veröffentlichtes Programm immer frei bleibt. Jede Kopie sowie darauf aufbauende Werke müssen wieder unter der gleichen Lizenz veröffentlicht werden. Hierbei kann zwischen stark und schwach schützenden Lizenzen unterschieden werden: Stark schützende Lizenzen verlangen, dass alles, was im Sinn des Urheberrechts als abgeleitetes Werk gilt, wieder unter die gleiche Lizenz gestellt wird, während sich bei schwach schützenden Lizenzen diese Forderung nur auf Änderungen am eigentlichen Programm bezieht.



GRAFIK 1 Veranschaulicht den Zusammenhang zwischen freier Software und proprietärer Software sowie dessen verschiedenen Ausprägungen und wie diese zu kostenloser Software im Allgemeinen stehen.

Aus Anwendersicht sind Freie-Software-Lizenzen einfach zu handhaben. Solange die Software nur eingesetzt oder für den eigenen Gebrauch modifiziert wird, müssen sich Anwender*innen mit der Lizenz nicht weiter auseinandersetzen, da es hierfür keine Einschränkungen gibt. Erst in dem Moment, wo man Kopien der originalen Software oder einer abgewandelten Version vertreiben will, muss man sich die Lizenz genauer ansehen, um zu wissen, ob diese in die erste oder die zweite Kategorie fällt.





PHILOSOPHIE

Software ist ein Schlüsselprodukt in unserer Welt. Programmierer*innen sind gefragt und gut bezahlt. Warum gibt es dann Menschen, die ihre Software frei »verschenken«? Antworten finden sich in der Geschichte, in den Strukturen und Werten der Freien-Software-Bewegung. In der Freien Software fließen verschiedene Wissenskulturen mit ihren Wertesystemen zusammen. Sie entstand zu einer bestimmten Zeit in der Entwicklung der Computertechnologie.

Wissenschaft

In seiner ETHIK DER WISSENSCHAFT schreibt der Wissenschaftssoziologe Robert Merton im Jahr 1942: »Die Erkenntnisse der Wissenschaft sind das Ergebnis sozialer Zusammenarbeit und werden der Gemeinschaft zugewiesen.« Auf den Schultern von Riesen stehend, sind »Wissenschaftler dem gemeinschaftlichen Erbe verpflichtet. In wetteifernder Kooperation erkennen sie die wesentlich kooperative und kumulative Qualität wissenschaftlicher Errungenschaften an. »Durch die Grundprinzipien der wissenschaftlichen Ethik werden Eigentumsrechte auf ein Minimum gestutzt. Der Anspruch des Wissenschaftlers auf ›sein‹ geistiges ›Eigentum‹ ist begrenzt auf Anerkennung und Respekt.«

Die namentliche Anerkennung, die Überprüfbarkeit von Ergebnissen durch wissenschaftliche ► *Peers*, also Gleichrangige, und der Fortschritt des gemeinschaftlichen Erkenntnisprojektes bedingen die Veröffentlichung der Forschungsergebnisse. Forscher*innen, die ihre Ergebnisse geheim halten oder patentieren lassen, werden von der Gemeinschaft missbilligt, betont Merton. Privates ›geistiges Eigentum‹ sei mit der wissenschaftlichen Ethik unvereinbar. Mertons Grundlagentext beschäftigt die Selbstreflexion der Wissenschaft bis heute, von Fragen der Qualitätssicherung bis zur ► *Open-Access*-Bewegung, die wissenschaftliche Öffentlichkeit gegen kommerzielle Schließung sichert.

Diese Normen, die Wissen als Gemeinschaftseigentum organisieren, finden wir in der Hacker-Ethik wieder, über die sie Eingang in die Freie Software erhalten.

Hippies und Hacker

Zeitreise in die 70er Jahre. Minicomputer sind die aktuelle Generation von Rechenanlagen. Sie füllen nicht mehr Säle, sondern nur noch Wandschränke. Die Anlagen finden in Wissenschaft, Verwaltung, Banken und Versicherungen ihren Einsatz. Die kostspielige Hardware wird verkauft oder geleast, die Software gibt es kostenlos dazu. Entwickler unter den Anwendern und Herstellern tauschen ihren Quellcode offen miteinander aus. Software ist noch kein eigenständiges Produkt, sondern ein weitgehend akademisches und damit kollektiv bearbeitetes Grundlagenfeld.

Studentische und soziale Bewegungen – Hippies, Anti-Kriegs-, Frauen-, Schwulen- und Umwelt-Bewegung – bestimmten die 1970er mit Ideen von hierarchiefreien, dezentralen und selbstorganisierenden Einheiten.

Studentische und soziale Bewegungen – Hippies, Anti-Kriegs-, Frauen-, Schwulen- und Umwelt-Bewegung – bestimmten die 1970er mit Ideen von hierarchiefreien, kleinen, dezentralen und selbstorganisierenden Einheiten. Eine dieser Bewegungen waren die **Hacker**. Sie entstanden am Rande der Universität und der jungen Disziplin Informatik. Die Werte und unausgesprochenen Grundannahmen dieser Szene hat Steven Levy in seinem Buch HACKERS (1984) erstmals in einer Hacker-Ethik zusammengefasst:

- »1. Der Zugang zu Computern und allem, was einem zeigen kann, wie die Welt funktioniert, sollte unbeschränkt und vollständig sein.
2. Alle Informationen sollten frei sein.
3. Misstraue Autoritäten – fördere Dezentralisierung.
4. Beurteile einen Hacker nach dem, was er tut und nicht nach Kriterien wie Abschlüssen, Alter, Rasse oder gesellschaftlicher Stellung.
5. Man kann mit einem Computer Kunst und Schönheit schaffen.
6. Computer können dein Leben zum Besseren verändern.«

Die Parallelen zu Mertons Wissenschaftsethik sind nicht zufällig. Wissenschaft und Hacker sind Gemeinschaften, die gemeinschaftlich an Erkenntnissen arbeiten und ihr Wissen miteinander teilen. Wie die Wissenschaft und soziale Bewegungen organisieren sich Hacker selbst und lehnen Diskriminierung ab.

In Deutschland wurde 1981 der CHAOS COMPUTER CLUB (CCC) aus der Taufe gehoben.⁴ Dort fanden sich Menschen zusammen, die von der Magie des Computers, vom Hacker-Geist und einem Bewusstsein für soziale Verantwortung getrieben waren. Der CCC übernahm die beschriebene Hacker-Ethik als sein Grundlegendokument und erweiterte sie später um zwei Punkte:

⁴ Chaos Computer Club e. V. (CCC) <http://ccc.de>

- »7. Mülle nicht in den Daten anderer Leute.
8. Öffentliche Daten nützen, private Daten schützen.«⁵

5 <https://ccc.de/en/hackerethik>

Die Ergänzungen sind einerseits motiviert von Fragen der Computersicherheit, mit denen Hacker heute oft assoziiert werden. Dabei ist diese Verbindung erst in den 1980er Jahren aufgekommen, während seit den 1950ern bis heute das Wort als Ehrentitel für jemanden verwendet wird, der besonders kreative technische Lösungen findet. Andererseits begründen sie den Beginn einer Entwicklung, die sich heute unter dem Namen OPEN DATA durchsetzt.

Software wird zur Ware

Anfangs wurde Software nur zusammen mit der Hardware verkauft. Ein Kartellverfahren gegen IBM leitete 1969 die Entkopplung und damit das Entstehen einer eigenständigen Softwareindustrie ein. Sie erlebte ihren Aufstieg mit der nächsten Generation von Rechnern, die in den 1970ern erschien.

1975 war Bill Gates beinahe aus der Harvard Universität zwangsexmatrikuliert worden, weil er seinen an der Uni entwickelten ▶ *BASIC*-Interpreter geheim halten wollte, um ein Geschäft zu gründen. Nach den Universitätsstatuten musste er seine Software gemeinfrei stellen. Darauf verließ er Harvard und gründete MICROSOFT. In den 1970ern wird Software von einem Kulturgut auch zu einem Wirtschaftsgut.

Von UNIX zu LINUX

Eine Schlüsselrolle für die Geschichte der Freien Software spielt das Betriebssystem ▶ *UNIX*. An ihm zeigt sich eine Entwicklung von einer offenen Wissensgemeinschaft zu einer Ware im Wettbewerb und schließlich zu einer offenen Infrastruktur für das digitale Ökosystem.

In den 1970ern begann die Vernetzung von Computern. AT&T, das staatlich regulierte Telefonmonopol in den USA, benutzte das Betriebssystem UNIX und entwickelte ▶ *UUCP* (*Unix to Unix Copy*), um über normale Telefonleitungen Daten zwischen seinen Rechnern auszutauschen. Die Protokolle TCP und IP (▶ *TCP/IP*) entstanden, auf denen bis heute das Internet beruht. Ging es anfangs darum, kostspielige Rechenressourcen entfernten Laboren zugänglich zu machen, trat nun die Verbindung von Menschen in den Vordergrund. Auf der Basis von UUCP entstand das ▶ *USENET* mit seinen Newsgroups, das erste weltweite soziale Netzwerk, das ebenfalls bis heute existiert. Der Computer wird von einer Rechenmaschine zu einem Medium.

Als reguliertes Monopol durfte AT&T nicht in den Softwaremarkt einsteigen, gab aber den UNIX-▶ *Quellcode* zum Selbstkostenpreis an Universitäten weiter. Dort wurde UNIX aufgegriffen und weiterentwickelt. Diese von AT&T

unabhängigen Elemente von UNIX wurden an der Berkeley Universität gesammelt. Hier wurden auch die TCP/IP-Protokolle in UNIX integriert. 1983 erschien die ▶ **BSD** (BERKELEY SOFTWARE DISTRIBUTION), in der erstmals die Betriebssysteme von Computer und Netz zusammengeführt wurden. Daher verbreitete sich BSD-UNIX rasch.

1984 endete das Telefonmonopol in den USA: AT&T wurde aufgespalten und UNIX in eine eigene Firma überführt, die es nun regulär vermarktete. Noch immer konnte man Quellcode-Lizenzen dafür erwerben, doch die kosteten schon bald 100.000 US\$. Unternehmen wie IBM, HP und SIEMENS erwarben sie und passten das Betriebssystem für ihren jeweiligen Zwecke an: Zahlreiche untereinander inkompatible UNIX-Varianten entstanden.

Das LABOR FÜR KÜNSTLICHE INTELLIGENZ (KI) am ▶ **M.I.T.** in Cambridge, Massachusetts gehörte in den 1980ern zu den Hacker-Paradiesen. Dort arbeitete und lebte der Betriebssystemexperte und wichtigste Gründer der Freien-Software-Bewegung Richard Stallman. Er sah die Ideale der Hacker-Ethik bedroht. Erst mit dem Entstehen eines Softwaremarktes setzte sich die Vorstellung durch, dass Software durch das Urheberrecht und in den USA seit 1980 auch durch Patente geschützt ist. Auch Stallmans Hacker-Kolleg*innen gründeten Firmen. Die Gemeinschaft des freien Austausches löste sich auf – in Konkurrenz mit geschlossenen Quellcodes und Vertraulichkeitsvereinbarungen (▶ **NDA**).

Ein Betriebssystem besteht aus einer Vielzahl von Werkzeugen und einem Kern. Dieser Kern war das letzte fehlende Element, um aus GNU ein vollständiges Betriebssystem zu machen.

In dieser Phase startete er 1984 das ▶ **GNU**-PROJEKT. Das steht für »GNU's Not Unix«, doch genau das war sein Ziel: ein Betriebssystem zu schreiben, das funktional äquivalent zu UNIX ist, aber keine einzige Zeile von AT&T geschützten Code enthält und in freier Kooperation weiterentwickelt werden kann.

Urheberrecht war eine Gefahr für diese Kooperation und zugleich ihre Lösung. Das Gesetz gibt dem Urheber volle Verfügung über sein Werk: Er kann andere davon ausschließen oder nicht. Stallman goss die Hacker-Werte von freier Kooperation und Bildung in einen urheberrechtlichen Nutzungsvertrag, die GNU GENERAL PUBLIC LICENSE (▶ **GPL**). Darin werden die vier Freiheiten mit einer Bedingung verknüpft: Kopien und veränderte Versionen freier Programme dürfen nur mit denselben Freiheiten verbreitet werden. Damit verhindert die GPL rechtssicher, dass Freie Software unfrei gemacht wird.

Ein Betriebssystem besteht aus einer Vielzahl von Werkzeugen und einem Kern. Dieser Kern war das letzte fehlende Element, um aus GNU ein vollständiges Betriebssystem zu machen. Die erste Version dieses Kerns namens

► **LINUX** veröffentlichte der finnische Informatiker Linus Torvalds 1991 unter einer Lizenz, die die kommerzielle Nutzung verbot. Ein Jahr später stellte er LINUX unter die GNU GPL. Damit war es nun möglich, LINUX und GNU zu dem vollständigen freien Betriebssystem GNU/LINUX zusammenzuführen, das oft nur als »Linux« bezeichnet wird.

UNIX in all seinen heute meist freien Varianten ist das am weitesten verbreitete Betriebssystem. Neben GNU/LINUX ist das aus der BERKELEY UNIVERSITÄT hervorgegangene ► **BSD** wichtig, das APPLES proprietärem Betriebssystemen ► **OS X** zugrunde liegt. Dank seiner Portabilität und Entwicklungsoffenheit findet sich UNIX heute überall: auf Supercomputern und ► **Servern**, auf Laptops und Mobiltelefonen oder eingebettet in industriellen Steuergeräten, Medizintechnik, Autos oder WLAN-Routern.

Wissensallmende

Gemeinsames geistiges Gut der modernen Informationsgesellschaft wird als »Wissensallmende« bezeichnet. Um Freie Software ist eine solche Allmende-Gemeinschaft entstanden. Sie hat sich ihre Infrastrukturen geschaffen mit Code-Repositories und Kommunikationskanälen. In Konflikten und Debatten hat sie sich ethische Grundregeln für ihre Zusammenarbeit gegeben, die an die Mertonsche Wissenschaftsethik und an die Hacker-Ethik anschließen. Die zentralen Elemente ihrer freien Kooperation hat sie rechtssicher in urheberrechtlichen Lizenzen kodifiziert. Dies alles dient der Nachhaltigkeit der »Allmende-basierten ► **Peer-Produktion**« (Yochai Benkler) von Freier Software. Digitale Wissensressourcen sind von Natur aus nicht knapp. Erst Urheberrecht und Patente ermöglichen es, Werke und Erfindungen zu veröffentlichen und gleichzeitig Dritte von deren Verwertung auszuschließen. Freie Software will aber gerade niemanden ausschließen. Dass Dritte mit dem Verkauf von Dienstleistungen mit Freier Software Geld verdienen dürfen, ist Konsens in der Gemeinschaft.

Wenn aber niemand ausgeschlossen werden soll, warum schert sich die Gemeinschaft überhaupt um Rechtsfragen? Anders als Patentschutz, der durch Anmeldung entsteht, ist ein Werk im Sinne des Urheberrechts mit seiner Entstehung automatisch geschützt. Nun könnten Autor*innen Freier Software einfach auf ihre Urheberrechte verzichten und ihre Werke gemeinfrei (engl.: *public domain*) erklären. Auch dafür braucht es aber eine Lizenz, wie die UNLICENSE oder CCo (CREATIVE COMMONS ZERO), die neben dem vollständigen Verzicht auf alle Urheberrechte zudem einen Gewährleistungsausschluss enthält. Einige Softwareprojekte gehen diesen Weg, doch die meisten verwenden Freilizenzen mit weiteren Bedingungen.

Liberaler Lizenzen wie ► **BSD** und ► **MIT** verpflichten zur Nennung des Urhebers in allen Kopien und abgeleiteten Werken. Copyleft-Lizenzen wie die ► **GPL** verpflichten darüber hinaus dazu, dass Kopien und abgeleitete Software nur

mit denselben Freiheiten verbreitet werden dürfen. So wird sichergestellt, dass Freie Software immer frei bleibt und das Universum mit jeder Weiterentwicklung wächst.

Dazu braucht es eine Ressource: die Bereitschaft von Programmierer*innen, zur Entwicklung beizutragen. Diese nimmt ab, wenn sie feststellen, dass ihre Software für unfreie Entwicklungen verwendet wird. Die ethische Norm verbietet die kommerzielle Nutzung nicht, verlangt aber eine Wechselseitigkeit von Geben und Nehmen. Auf diese Weise erzeugen COPYLEFT-Lizenzen eine nachhaltige Wissensallmende.

Programmierer*innen wollen Software entwickeln und interessieren sich für Urheberrecht allenfalls, wenn es ihnen auf die Füße fällt. Genau das passiert aber, wenn sie keine Lizenz wählen. Ihre Software steht automatisch unter dem Urheberrecht. Dritte müssen sie um Erlaubnis bitten, wenn sie damit weiterarbeiten wollen. Software verwendet in aller Regel Elemente aus verschiedenen Projekten. Vor einer Veröffentlichung müssen für alle die Rechte geklärt werden, und die jeweiligen Lizenzen müssen ihre Verbindung erlauben.⁶

Freie Software läuft immer wieder Gefahr, eingehengt zu werden. So haben viele Freilizenzen auf die Bedrohung durch Softwarepatente reagiert. Die Lizenzunlust stellt eine – wenn auch unwillentliche – Gefahr für die Nachhaltigkeit der Wissensallmende dar. Die Debatte darüber in der Allmende-Gemeinschaft hat begonnen.⁷

Warum machen die das?

Wissenschaftliche Ethik, Hacker und Hippies allein hätten der Freien Software wohl nicht zum Durchbruch verholfen. Eine wichtige Rolle dabei hat der Wandel in der Informations- und Kommunikationsindustrie von Infrastrukturen (Betriebssystemen und Netzen) zu Mehrwertdiensten und Inhalten gespielt. Für die großen Internet-Unternehmen ist Software Mittel zum Zweck. IBM und APPLE verdienen ihr Geld mit Hardware, GOOGLE und FACEBOOK mit Werbung und AMAZON mit Lieferdiensten. Für ihre Plattformen auf bewährte, ausgereifte Freie Software aufbauen zu können, spart Kosten. In der jeweiligen Gemeinschaft finden sie Programmierer*innen, die sie anheuern können. Vor allem für sicherheitsrelevante Software hat sich die Erkenntnis durchgesetzt: Geheimniskrämerei ist zu misstrauen; nur Quellenoffenheit erlaubt es, zu überprüfen, dass sich in einem Programm keine Hintertüren, Schadsoftware oder Sicherheitslücken verstecken. Umgekehrt schadet es den Unternehmen nicht, ihre eigenen Entwicklungen in die Community zurück zu geben. Aus den gleichen Gründen setzt auch der öffentliche Sektor immer mehr auf Freie Software.⁸

Neben diesen pragmatischen, wirtschaftlichen Entwicklungen bleibt Freie Software in ihrem Kern eine Wissensgemeinschaft. Eine der wenigen großen Studien zur Motivation in der Freien Software ist der FLOSS Survey der

6 Für einen Vergleich von Freilizenzen und ihren Kompatibilitäten siehe:

➤ <http://gnu.org/philosophy/license-list.html>

7 ASAY, MATT (2015): The Github Kids still Don't Care about Open Source.

➤ <http://techrepublic.com/article/the-github-kids-still-dont-care-about-open-source>

8 ➤ https://de.wikipedia.org/wiki/Open-Source-Software_in_%C3%B6ffentlichen_Einrichtungen

UNIVERSITÄT MAASTRICHT VON 2002.⁹ 20% der Befragten waren Student*innen, die Mehrzahl Softwareingenieure oder Programmierer*innen, die meisten Angestellte, die in ihrer Freizeit an Freier Software arbeiten. Die angegebenen Gründe sind vielfältig: Dazu gehören pragmatische (ein Problem lösen, die Software anderer Entwickler*innen verbessern), soziale (auf eine neue Art kooperieren, mithelfen, die Idee für eine Software zu verwirklichen) und in geringem Maß auch finanzielle Gründe (sich Reputation erwerben, um die Chancen auf dem Arbeitsmarkt zu verbessern, Geld verdienen). Die am häufigsten genannten Gründe sind jedoch, neue Fertigkeiten zu lernen und Wissen an andere weitergeben zu wollen.

Wissenschaft, Hacker und die Freie-Software-Bewegung sind Gemeinschaften von Lehrenden und Lernenden. Die gedeihen am besten in Freiheit und Offenheit.

9 INTERNATIONAL INSTITUTE OF INFONOMICS, UNIVERSITY OF MAASTRICHT (2002): Free/Libre and Open Source Software: Survey and Study. FLOSS Final Report. <https://goo.gl/CGbHCI> und <http://flossproject.merit.unu.edu/report/index.htm>



FREIE SOFTWARE IN DER BILDUNG

Die Geschichte von Freier Software und Schule ist zumindest in ihren Anfängen äußerst schwierig zu erfassen, da es kaum Quellen für diese Entwicklung gibt. Es ist davon auszugehen, dass es mittlerweile kaum noch Schulen gibt, die keine Freie Software einsetzen. Da sie oftmals kostenlos im Internet verfügbar ist (wie z.B. der Webbrowser FIREFOX oder der Mediaplayer VLC) und ein hohes Maß an Qualität bietet, ist ein solcher Einsatz für Schulen genauso wie für viele andere Anwender*innen naheliegend.

Geprägt wird das Bild von einer sehr vielfältigen Nutzung Freier Software. Während ein Großteil einzelne Programme (wie Webbrowser, Office-Pakete oder Multimediaanwendungen) meist auf Basis von WINDOWS nutzt, setzen einige Schulen auf freie ▶ *Betriebssysteme*.¹⁰ Wurden umfassende Freie-Softwarelösungen in Schulen oder anderen Bildungseinrichtungen früher durch engagierte Einzelpersonen umgesetzt und gegebenenfalls durch die ansässige Computer AG unterstützt, so sind heute auch auf dem freien Betriebssystem LINUX basierende Komplettlösungen für Bildungszwecke bei kommerziellen Anbietern verfügbar.

Die Lizenzbedingungen Freier Software wirken sich auf die Schule sehr unterschiedlich aus. Sie eröffnen in vielerlei Hinsicht Chancen, erfordern aber auch in manchen Bereichen Aufmerksamkeit und Verantwortungsbewusstsein.

¹⁰ Ein sehr erfolgreiches Beispiel bietet das KATHARINEUM ZU LÜBECK:
➤ <http://katharineum.de>

Wirtschaftlichkeit und Soziales

Wer sich mit dem Konzept und den Handlungsoptionen Freier Software beschäftigt, dem sticht das Argument der Wirtschaftlichkeit ins Auge. Da Freie Software oftmals kostenfrei über das Internet zu beziehen ist und keine Lizenzgebühren anfallen, lassen sich Anschaffungskosten senken. Auch wenn große Softwareunternehmen spezielle Konditionen für Bildungseinrichtungen anbieten, so können diese in den allermeisten Fällen nicht mit dem kostenlosen

Angebot von Freier Software mithalten. Proprietäre Anwendungen bieten zwar teilweise einen größeren Funktionsumfang, dieser wird im schulischen Kontext jedoch nahezu nie benötigt oder ausgeschöpft.

Die günstigen Anschaffungs- und lizenzbedingten Betriebskosten werden von vielen als ein entscheidender Faktor beim Einsatz von Freier Software in der Schule betrachtet. Allerdings greift diese Einschätzung oft zu kurz, denn die kostenverursachenden Elemente beim Einsatz von Software sind nicht ausschließlich die Lizenzen. Kostenfaktoren sind ebenso Administration, Wartung, Weiterentwicklung und die Schulung der Nutzer*innen. Diese Aufgaben werden teilweise von externen Partnern*innen übernommen, was wiederum Ausgaben für die Schule bedeutet. Selbst die interne Schulung durch das Kollegium kostet Zeit und somit Geld – wenn auch deutlich weniger. Dennoch deuten viele Faktoren darauf hin, dass die Nutzung von Freier Software auf Dauer zu geringeren Kosten führt.

Da eine Freie-Software-Lizenz es grundsätzlich ermöglicht, eine Software auf beliebig vielen Rechnern zu jedem Zweck zu nutzen, ist die Weitergabe an Schüler*innen unproblematisch realisierbar. So haben Schüler*innen unabhängig von ihrer sozialen Herkunft die Möglichkeit, mit der gleichen hochwertigen Software wie in der Schule zu arbeiten.

Rechtssicherheit

Wer kennt nicht die Kolleg*innen, die zu Hause genutzte Software gerne in der Schule einsetzen würden. Dieser Wunsch ist nicht nur nachvollziehbar, sondern auch legitim. Prozesse lassen sich so reibungsloser gestalten. Lizenzbedingungen von Software lassen, selbst wenn sie kostenfrei verfügbar ist, so einen Schritt nicht immer zu. Oft müssen zusätzliche Lizenzen erworben werden, damit z.B. eine bestimmte Menge an Installationen der Software genutzt werden kann oder der Einsatz außerhalb des rein privaten Umfeldes erlaubt ist.

Freie Software ist sie per Definition erst dann, wenn unter anderem die Nutzung zu jedem Zweck und das beliebige Anfertigen von Kopien erlaubt sind. Schulen, die Freie Software nutzen, können sicher sein, dass sie nicht gegen Lizenzen verstoßen, wenn sie die Software beispielsweise auf allen Computern in der Schule installieren oder diese den Schüler*innen für die Nutzung zu Hause weitergeben.

Bezugsquellen

Aufgrund der Lizenzbedingungen Freier Software ist es jeder Person oder Institution möglich, eine Software z.B. als Download auf der eigenen Webseite anzubieten. Was auf der einen Seite komfortabel scheint, ist auf der anderen Seite

problematisch. Bei einer einfachen Onlinesuche nach den Begriffen »Download« und »LibreOffice« finden sich auf der ersten Seite mehr als fünf unterschiedliche Webseiten, auf denen LIBREOFFICE zum Download angeboten wird.

Rechtlich ist an dem Angebot nichts auszusetzen, da die freie Lizenz jedem das Recht zur Nutzung und Weiterverbreitung einräumt. Trotzdem sollte man sich bei Unsicherheit immer auf das Angebot des Herausgebers der jeweiligen Software verlassen und diese nur dort herunterladen. Sonst kann es z.B. passieren, dass man nicht die jeweilige Software installiert, sondern nur ein Tool zu ihrer Installation. Dieses bietet dann während des Installationsprozesses auch andere Software in Form von Werbung an.

Diese Vorsicht gilt vor allem für die Nutzer*innen der Betriebssysteme Windows und ▶ *OS X*. Die Nutzer*innen von LINUX installieren die gewünschte Software über Paketquellen und sind so meist von dieser Problematik ausgenommen.

Abspaltungen

Bei Freier Software ist es erlaubt und erwünscht, sie weiterzuentwickeln und zu verbessern. Wenn sich Entwickler*innen entscheiden, eine Kopie einer bestehenden Freien Software anzulegen und diese ihren Bedürfnissen anzupassen, so spricht man von einer »Abspaltung« bzw. einem »▶ *Fork*«. Für die Entwickler*innen sind solche Forks äußerst hilfreich, weil sie für ihre neue Software auf einer bestehenden aufbauen können; für die Nutzer*innen indirekt, weil sie von der Arbeitersparnis der Entwicklung profitieren. Ein sehr bekanntes Beispiel ist das Office-Paket LIBREOFFICE. Es entstand aus OPENOFFICE, das mittlerweile nur noch sehr langsam weiterentwickelt wird.

Teilweise ist es schwierig, den aktuellen Entwicklungsstatus und die Schwerpunktsetzung von Softwareprojekten nachzuvollziehen und zu verfolgen. Als Extrembeispiel dienen hier die ▶ *LINUX-Distributionen*. Jede der Distributionen hat einen eigenen Fokus und ist für bestimmte Zwecke ausgelegt. Mittlerweile gibt es mehrere Hundert Distributionen. Für Personen, die mit der Materie nicht vertraut sind, kann die Fülle abschreckend wirken. Relativierend ist jedoch darauf hinzuweisen, dass es nur eine überschaubare Anzahl häufig genutzter Distributionen gibt und es sich bei vielen anderen um Nischenprodukte handelt, mit denen man im schulischen Umfeld kaum oder nie in Kontakt kommt.

Unabhängigkeit

Unternehmen sichern ihren Selbsterhalt durch ihr Geschäftsmodell. Hierfür ist es z.B. notwendig, dass ein Kunde, in diesem Fall die Schule, eine Lizenz für das Produkt des Unternehmens erwirbt. Um sicherzustellen, dass der Kunde

auch weiterhin das Produkt nutzt, werden die Dateien in einem Format abgespeichert, das nur der Hersteller dieses Produktes vollständig unterstützt. So würde sich ein Softwarewechsel als schwierig oder kostenintensiv erweisen, weil die Kompatibilität der Dateien mit neuer Software nicht sichergestellt wäre. Dieser Effekt nennt sich »Vendor-Lock-in«.

Freie Software gibt die Spezifikationen der Dateiformate frei, sodass Nutzer*innen die Möglichkeit haben, das Produkt zu wechseln, ohne zwangsläufig mit hohen Kosten oder Kompatibilitätsproblemen konfrontiert zu werden. Die Nutzung Freier Software schafft so einen Raum, in dem kommerzielle Interessen zweitrangig sind und Abhängigkeiten von Großkonzernen oder Einzelunternehmen vermieden werden können.

Übereinstimmende Ideale

Die Ideen und Prinzipien, auf denen Freie Software basiert, sind auch in den Konzepten moderner Schulen – wie dem offenen, schüleraktiven Unterricht – zu finden. Wissen teilen, Partizipation und Offenheit sind entscheidende Elemente bei schulischen Arbeitsformen, beispielsweise beim kooperativen Arbeiten. Sie stehen zudem stellvertretend für das demokratische Grundprinzip, welches Teil unseres Erziehungssystems ist. Eben dieses Grundprinzip ist eng mit der Freie-Software-Bewegung verbunden, auch wenn es nicht immer eindeutig und zudem Wandlungsprozessen unterworfen ist. Das macht eine Schule, die Freie Software im Bildungskontext einsetzt, konsistenter in ihrer Ausrichtung und glaubhafter. Ein ganzheitlicher Einsatz von Freier Software kann in der Bildung realisiert werden, denn ihre Prinzipien werden hier nicht nur auf die ideelle, sondern auch auf die technische Ebene übertragen.

Flexibilität

Durch die freie Lizenzierung von Software können die Programme und/oder Betriebssysteme von einer großen Gemeinschaft angepasst und verändert werden. So entstehen auch Lösungen, die für sehr spezielle Anwendungsszenarien gedacht sind. Großer Beliebtheit erfreuen sich sogenannte LIVE-Systeme. Dabei handelt es sich um ein Betriebssystem, welches von einem externen Medium, wie einem USB-Stick oder einer DVD, gestartet werden kann.

Schulen können mithilfe von LIVE-Systemen mehrere typische Probleme auf einmal lösen. Das gravierendste ist der meist unzureichende oder arbeitsaufwändige Support, den ein schulisches Computernetzwerk oft benötigt. Durch ein LIVE-System kann eine einheitliche Lern- und Arbeitsumgebung erzeugt werden, welche nicht nur alle für die Schule nötigen Programme umfasst und zudem am heimischen Rechner genutzt werden kann, sondern auch als USB-Stick äußerst praktisch zu transportieren ist. Daten müssen nicht mehr

synchronisiert werden, da sie auf dem jeweiligen Stick abgespeichert werden. So reduziert sich der Wartungs- und Supportaufwand für die Schule. Das Vorzeigeprojekt in dieser Kategorie, der »Lernstick«, stammt von der Pädagogischen Hochschule Nordwestschweiz und wird dort am Institut für Weiterbildung und Beratung entwickelt und gepflegt.¹¹ Zeitgleich ist beim Einsatz von LIVE-Systemen Vorsicht geboten, vor allem wenn sie parallel zu einem bestehenden System genutzt werden. Denn durch das Starten eines eigenen Systems mit Zugriff auf die Festplatte des Rechners öffnet man unter Umständen die Tür und Tor.

Ein weiterer Vorteil sind in diesem Kontext die portablen Anwendungen, die auch unter dem Begriff der »digitalen Schultasche« bekannt sind. Diese Anwendungen sind ohne Installation direkt vom Trägermedium ausführbar. Durch die Erlaubnis, den Programmcode anzupassen und zu verändern, konnten diese Anwendungen einfacher für solche Zwecke nutzbar gemacht werden.

11 Mehr Informationen zu dem Projekt und zu den Partnerschulen finden sich unter <http://imediaw.ch/projekte/lernstick>

Freie Software als Werkstoff

In Fachkreisen wird auf die Möglichkeit hingewiesen, dass sich durch den Einsatz von Freier Software als Werkstoff informatische Kompetenzen erwerben lassen. Programmcodes können im Rahmen von Unterricht oder nachmittäglicher AG studiert und angepasst werden. Dies erfordert jedoch ein tiefgreifendes Verständnis der Materie und bietet sich außerhalb des Informatikunterrichts selten an.

Freie Software bietet viele Chancen für die Schule, welche sich direkt oder indirekt aus der Existenz der freien Lizenzen ableiten lassen. Diese erst geben den Nutzer*innen den rechtlichen Rahmen, von Nutzung, Weitergabe und Verbesserung des Programmcodes durch die internationale Gemeinschaft zu profitieren. Trotzdem ist durch die gesteigerte Freiheit auch mehr Verantwortung von Seiten der Schule gefordert.



ÜBERSICHT ÜBER FREIE SOFTWARE FÜR SCHULE UND UNTERRICHT

Im Folgenden wird Freie Software vorgestellt, die in den drei Betriebssystemwelten Windows, ► *GNU*/LINUX und MAC ► *OS X* verfügbar ist. Wo vorhanden, wird auf portable Versionen hingewiesen. Dabei bietet die Softwareübersicht einen Einblick in Werkzeuge aus allen für die Schule relevanten Bereichen: Präsentation und Textverarbeitung, Bildbearbeitung und Grafik, Audio und Video, Internet und Kommunikation, Unterrichtsunterstützung und Fächerspezifisches, Unterrichtsorganisation und Klassenmanagement.

Präsentation und Textverarbeitung

Das Verfassen von Texten – der Umgang mit der Schriftsprache – ist ein zentraler Aspekt von Lehren und Lernen. Im schulischen Alltag sind es Aufsätze, Dokumentationen, Beschreibungen und vieles andere mehr, die in Textform erstellt werden. Neben reinen Texten können durch den Einsatz des Computers Texte mit multimedialen Inhalten (wie Bildern, Grafiken, Klängen und Videos) ergänzt und präsentiert werden.

Das Office-Programmpaket LIBREOFFICE

Portable App: Ja **Benutzeroberfläche deutsch:** Ja **Lizenz:** MPL, LGPL 3+, GPL 3+

Das Programmpaket LIBREOFFICE ist der Nachfolger von OPENOFFICE.ORG. OPENOFFICE wird zwar noch weiterentwickelt, enthält aber wichtige Aktualisierungen und Funktionen nicht mehr. Die wichtigsten Bausteine der Office-Suite sind: WRITER für Textverarbeitung, CALC für Tabellenkalkulation, IMPRESS für Präsentationen, DRAW als Zeichenprogramm, BASE für Datenbanken. Da sich das LIBREOFFICE-Programmpaket in sehr aktiver Entwicklung befindet, unterscheiden sich die Versionen 3.x, 4.x und 5.x deutlich.

Weblinks:

Projektseite:

🔗 <https://de.libreoffice.org>

Portable App:

🔗 http://portableapps.com/de/apps/office/libreoffice_portable

Weiteren Informationen:
LIBREOFFICE-Handbuch:

➤ <https://de.libreoffice.org/get-help/documentation>

Sammlung von praktischen Anwendungsbeispielen aus dem Bildungskontext in unterschiedlichen Schulfächern:

➤ <https://de.libreoffice.org/get-help/users/#bildung>

Weblinks:

Projektseite:

➤ <https://de.libreoffice.org>

Portable App:

➤ http://portableapps.com/de/apps/office/libreoffice_portable

Weiteren Informationen:
LIBREOFFICE-Handbuch:

➤ <https://de.libreoffice.org/get-help/documentation>

EINSCHÄTZUNG: Für die Arbeit in der Schule ist es in der Regel entscheidend, dass Anwendungsprogramme einfach aufgebaut und nicht mit zu vielen Funktionen ausgestattet sind. Deswegen ist vor der Entscheidung für eine Version von LIBREOFFICE zu prüfen, wie groß der Programmumfang wirklich sein soll. Mit Stand Mitte 2016 gilt die Empfehlung, dass die Versionen 4.x für Schulen am sinnvollsten einsetzbar sind.

Das gesamte LIBREOFFICE-Programmpaket ist auch als portable Version für USB-Sticks unter dem Betriebssystem Windows erhältlich. Die Programme werden nicht direkt auf einem Computer installiert, sondern auf einem Stick gespeichert und von dort gestartet. Damit muss kein Arbeitsplatzrechner verändert werden. Gleichzeitig steht Schüler*innen eine einheitliche Arbeitsumgebung zur Verfügung, die problemlos auch zu Hause genutzt werden kann.

Um im Unterricht einzelne Arbeitsschritte mit den Programmen und unterschiedlichen Programmversionen von LIBREOFFICE (Anlegen eines Inhaltsverzeichnisses, Formatieren einer Präsentationsfolie, Darstellung von Diagrammen usw.) zu vermitteln, erscheinen ▶ *Video-Tutorials* sinnvoll. Hier kann wegen der großen Vielfalt und unterschiedlichen Qualität der Videos nur die Empfehlung gegeben werden, dass sie vor dem Unterricht auf ihre Tauglichkeit geprüft werden.

LIBREOFFICE WRITER

Portable App: Ja **Benutzeroberfläche deutsch:** Ja **Lizenz:** MPL, LGPL 3+, GPL 3+

LIBREOFFICE WRITER bietet umfangreiche Möglichkeiten zur Erstellung und Bearbeitung von Texten. Neben der allgemein üblichen Arbeit mit Formatierungen kann mit Formatvorlagen gearbeitet werden, mit denen Formatierungen global bearbeitet und zugewiesen werden. Für umfangreichere Arbeiten gibt es automatisierte Inhaltsverzeichnisse und die komfortable Einbindung von Quellenangaben. In die Dokumente können Bilder eingebunden und mit grundlegenden Funktionen wie Helligkeit, Kontrast usw. manipuliert werden. Für alle Dokumente, in denen hauptsächlich Texte gestaltet und strukturiert werden müssen: Berichte, Dokumentationen, Lebensläufe und Bewerbungsschreiben.

EINSCHÄTZUNG: LIBREOFFICE WRITER ist eine sehr gut geeignete Software für alle schulischen Bereiche. Um die Kompatibilität zu anderen Textverarbeitungsprogrammen zu erhöhen, wird empfohlen, Dokumente im ▶ *DOC*-Format abzuspeichern. Das standardmäßig verwendete ▶ *ODF*-Format wird von anderen Programmen und unter anderen Betriebssystemen teilweise nur unzureichend unterstützt. Für den Austausch von Dokumenten, die nicht mehr bearbeitet werden müssen, ist eine Exportfunktion für Dokumente in das ▶ *PDF*-Format integriert. Damit können auch Dokumente und ausfüllbare Formulare auf der Schulhomepage hinterlegt werden. Her-

vorzuheben ist, dass es für Writer redaktionell betreute Unterrichtsmaterialien gibt, die z.B. für die Vorbereitung der Prüfung des Europäischen Computerführerscheins ▶ *ECDL* genutzt werden können.

▶ *Siehe auch das Werkzeugporträt zu LIBREOFFICE WRITER.*

LIBREOFFICE IMPRESS

Portable App: Ja **Benutzeroberfläche deutsch:** Ja **Lizenz:** MPL, LGPL 3+, GPL 3+

Impress ist das Präsentationsprogramm des LIBREOFFICE-Programmpakets. Impress-Präsentationen werden aus Abfolgen von Folien erstellt, die unterschiedliche Elemente enthalten können: Textfelder, Grafiken und Multimediainhalte wie Audioclips oder Videoclips können interaktiv miteinander kombiniert werden. Neben dem Speichern in das offene ▶ *ODP*-Format ist auch ein Export der Folien in ▶ *PDF* möglich.

EINSCHÄTZUNG: Neben der klassischen Präsentation von Inhalten sind einige Funktionen hervorzuheben, die Impress im schulischen Zusammenhang sehr nützlich machen. Impress besitzt eine Importfunktion für Bilderserien. Damit lassen sich sehr schnell Dia-Shows (z.B. von Fotos einer Klassenfahrt) erstellen, die wiederum mit Musik hinterlegt werden können. Zudem können mehrere importierte Einzelbilder zu einer Animation im GIF-Format zusammengefasst und damit kurze Trickfilme realisiert werden.

Dokumente sind einfach in das PDF-Format zu exportieren und können zwischen unterschiedlichen Geräten und Plattformen ausgetauscht werden: Präsentationen können dann (z.B. als Hausaufgabe) abgegeben, das heißt verschickt werden.

LIBREOFFICE CALC

Portable App: Ja **Benutzeroberfläche deutsch:** Ja **Lizenz:** MPL, LGPL 3+, GPL 3+

CALC ist das Programm für Tabellenkalkulation aus dem LIBREOFFICE-Paket. Es dient der vielseitigen und interaktiven Erstellung von Tabellen und grafischen Darstellung von Tabelleninhalten. Mit den in den Tabellen enthaltenen Werten lassen sich vielfältig Berechnungen anstellen und diese durch mathematische und logische Funktionen manipulieren. Neben GEOGEBRA bietet sich auch CALC für die Darstellung von Wertereihen in unterschiedlichen Diagrammtypen an.

EINSCHÄTZUNG: CALC kann sinnvoll ab der Klassenstufe 6/7 eingesetzt werden. Einsatzmöglichkeiten finden sich klassischerweise in den Fächern Mathematik, Physik oder sonstigen Fächern, in denen Werte(reihen) ausgewertet werden sollen.

Weblinks:

Projektseite:

➤ <https://de.libreoffice.org>

Portable App:

➤ http://portableapps.com/de/apps/office/libreoffice_portable

Weiteren Informationen:

LIBREOFFICE-Handbuch:

➤ <https://de.libreoffice.org/get-help/documentation>

Weblinks:

Projektseite:

➤ <https://de.libreoffice.org>

Portable App:

➤ http://portableapps.com/de/apps/office/libreoffice_portable

Weiteren Informationen:

LIBREOFFICE-Handbuch:

➤ <https://de.libreoffice.org/get-help/documentation>

Weblink:

Projektseite:

➤ <https://latex-project.org>**Weiteren Informationen:**

Einführung:

➤ <http://dante.de/tex/TeXAnfaenger.html>

LATEX

Portable App: Ja **Benutzeroberfläche deutsch:** Ja **Lizenz:** LaTeX Project Public License

Mit LATEX kann man Dokumente wie wissenschaftliche Arbeiten mit hochwertigen Formatierungen erstellen. Unter LATEX versteht man dabei eine Sammlung von Software, die das Satzprogramm TEX mit verschiedenen Funktionen (sogenannten Makros) bedienbar macht. TEXTE werden dabei nicht wie in regulären Textverarbeitungsprogrammen so eingegeben, wie sie später im Ausdruck erscheinen, sondern mit Formatierungsbefehlen zum Textsatz versehen. LATEX übernimmt dann diese Formatierung.

Die Makros steuern z.B. die Anlage von Inhaltsverzeichnissen oder Quellenangaben. Anders als reguläre Textverarbeitungsprogramme wie LIBREOFFICE oder ▶ *DTP-Programme* wie SCRIBUS ist LATEX für rein textbasierte Dokumente vorgesehen, die allerdings um Bilder bzw. Grafiken ergänzt werden können.

EINSCHÄTZUNG: Zielgruppe der Software ist die gymnasiale Oberstufe, in der auf die Erstellung wissenschaftlicher Arbeiten vorbereitet wird.

Bildbearbeitung und Grafik

Bei digitalen Bildern unterscheidet man zwischen Rastergrafiken und ▶ *Vektorgrafiken*. Raster- oder Bitmapgrafiken bestehen aus Bildpunkten (Pixeln) mit einem Farbwert, die in einem zweidimensionalen Raster angeordnet sind. Mit Mal- und Zeichenprogrammen erstellte Bilder, Scans und Fotos sind Rasterbilder. Vergrößert man Rasterbilder, vergrößern sich auch die rechteckigen Pixel und es entsteht der charakteristische Treppcheneffekt. Die bekanntesten Rastergrafikformate sind ▶ *BMP*, GIF, JPEG, PNG und TIFF. Folgende Programme des Werkzeugkastens erstellen und bearbeiten Rastergrafiken: LIBREOFFICE DRAW, GIMP, PINTA, KRITA, RAWTHERAPEE.

Vektorgrafiken bestehen aus mathematischen Beschreibungen von grafischen Primitiven wie Linien, Kreisen und Polygonen. So werden für einen Kreis die Werte der Lage des Kreismittelpunkts und des Kreisdurchmessers gespeichert. Weitere Werte für Strichstärke, Farbe usw. können hinzukommen. Ein solcher Kreis oder komplexere Figuren wie ein Buchstabe oder ein Logo lassen sich ohne Treppcheneffekt vergrößern und so in bestmöglicher Qualität auf Rasterausgabegeräten wie Bildschirmen und Druckern darstellen. Sie werden unter anderem für Schriften, Druckvorlagen, technische Zeichnungen, Karten und 3D-Animationen in Kinofilmen verwendet. Vektorgrafiken entstehen durch geometrische Modellierung. Sie werden in 2D- oder 3D-Vektorgrafikprogrammen erstellt. Rastergrafiken können durch Algorithmen mit gewissen Einschränkungen in Vektorgrafiken umgewandelt werden. Vektor- und Rastergrafiken können auch kombiniert werden, z.B. in Seitenbeschreibungssprachen wie POSTSCRIPT und dem daraus abgeleite-

ten ▶ *PDF*. Die bekanntesten 2D-Vektorgrafikformate sind SVG, DXF, EPS und PDF. Folgende Programme dieses Werkzeugkastens erstellen und bearbeiten Vektorgrafiken: LIBRECAD, DIA, LATEX, INKSCAPE, BLENDER. Mit SYNFIG, LIBREOFFICE IMPRESS und SCRIBUS lassen sich sowohl Raster- wie Vektorgrafiken verarbeiten.

BLENDER

Portable App: Ja **Benutzeroberfläche deutsch:** Ja **Lizenz:** GPL 2, einzelne Module Apache 2.0

BLENDER ist ein umfangreiches Programmpaket zum Erstellen von fotorealistischen Grafiken, Animationen bis hin zu Animationsfilmen und Software. Wesentliche Funktionen sind das Modellieren, ▶ *Rendern* und Animieren von dreidimensionalen Körpern. Mit den in Blender enthaltenen ▶ *Tools* können Oberflächen gestaltet, das Verhalten von Flüssigkeiten und die Bewegung von Körpern simuliert, Videos geschnitten und interaktive Spiele erstellt werden. Durch die Möglichkeit, ▶ *STL*-Dateien zu importieren, zu verarbeiten und zu exportieren, können mit Blender erstellte virtuelle Körper mit 3D-Druckern oder ▶ *CNC*-Fräsen physisch realisiert werden. Die Software verfolgt im Besonderen ein tastaturbasiertes Bedienkonzept, das heißt, dass viele Funktionen über Tastaturkurzbefehle aufzurufen sind. Blender kann in allen Zusammenhängen eingesetzt werden, in denen virtuelle dreidimensionale Körper und Räume eine Rolle spielen.

EINSCHÄTZUNG: Den gesamten Umfang der Software zu nutzen erscheint zunächst sehr schwierig. Zudem stellt das Erlernen und die Nutzung der Tastaturkurzbefehle eine besondere Hürde dar. Empfehlenswert ist es, Teilaspekte (z.B. die Gestaltung von Oberflächen, das Erstellen von Körpern oder die Animation von sehr einfachen Körpern) zu thematisieren. Die Zielgruppe ist am ehesten in Wahlpflicht- oder besonderen Informatikkursen ab Klassenstufe 9/10 zu verorten.

Weblink:

Projektseite (englisch):

🔗 <https://blender.org>

Weiteren Informationen:

Wikipedia:

🔗 [https://de.wikipedia.org/wiki/Blender_\(Software\)](https://de.wikipedia.org/wiki/Blender_(Software))

Weblinks:

Projektseite:

➤ <http://dia-installer.de>**Portable App:**➤ http://sourceforge.net/projects/portableapps/files/Dia%2520Portable/DiaPortable_0.97.2.paf.exe/download**Weiteren Informationen:**

Handbuch:

➤ <http://dia-installer.de/doc/de/index.html>**Wikipedia:**➤ https://de.wikipedia.org/wiki/Dia_%28Software%29**Weblinks:**

Projektseite:

➤ <https://gimp.org>**Portable App:**➤ http://portableapps.com/de/apps/graphics_pictures/gimp_portable**Weiteren Informationen:**

Handbuch:

➤ <https://docs.gimp.org/2.8/de>**Tutorials (englisch):**➤ <https://gimp.org/tutorials>**Wikipedia:**➤ <https://de.wikipedia.org/wiki/GIMP>

DIA

Portable App: Ja **Benutzeroberfläche deutsch:** Ja **Lizenz:** GPL2

DIA ist ein Zeichenprogramm, das auf Diagramme spezialisiert ist. Es unterstützt das Erstellen von Beziehungs-, Fluss- und Netzwerkdiagrammen, Programmablaufplänen und vielem mehr. Symbolsätze aus der Chemie, Elektronik, Optik usw. können geladen werden. Die Ergebnisse können sowohl als Raster- als auch als Vektorgrafiken exportiert werden. Letzteres ist hilfreich, wenn die Diagramme ausgedruckt werden sollen.

EINSCHÄTZUNG: Mit DIA lassen sich zeitliche Abläufe, das Organigramm einer Schule, ein Familienstammbaum oder Geschäftsprozesse einfach strukturiert darstellen. Das Erlernen fällt sehr leicht und ist intuitiv. Dia ist z.B. für das Darstellen von Schaltkreisen gut geeignet.

GIMP

Portable App: Ja **Benutzeroberfläche deutsch:** Ja **Lizenz:** GPL v3

Das GNU IMAGE MANIPULATION PROGRAM (GIMP) ist ein Bildbearbeitungsprogramm für Rasterbilder wie Fotos und Grafiken. Das seit 1995 entwickelte Programm verfügt über einen großen Funktionsumfang. Dazu gehören zahlreiche Mal- und Auswahlwerkzeuge, Ebenen und Masken sowie ein umfassendes Farbmanagement. Filter erlauben es, Fotos zu schärfen und rote Augen zu korrigieren oder sie in Filmstreifen, ein Puzzle oder ein kubistisches Gemälde zu verwandeln. Wiederkehrende Bildbearbeitungsvorgänge lassen sich über Skripte automatisieren.

EINSCHÄTZUNG: GIMP ist als universelles Bildbearbeitungsprogramm für sehr vieles geeignet: von einfachen Aufgaben, wie dem Zuschneiden von Bildern, über die Bearbeitung von Fotos und Grafiken für Präsentationen, Web und Druck oder die Erstellung von Animationen, die als GIF exportiert werden, bis hin zu aufwendigen Bildkompositionen.

► *Siehe auch das Werkzeugporträt zu GIMP.*

INKSCAPE

Portable App: Ja **Benutzeroberfläche deutsch:** Ja **Lizenz:** GPL v3

INKSCAPE ist ein leistungsstarkes Programm zur Erstellung und Bearbeitung von zweidimensionalen Vektorgrafiken. Inkscape verwendet das SVG-Format (SCALABLE VECTOR GRAPHICS), importiert aber auch andere Vektorformate wie EPS und PDF sowie Rastergrafiken in allen üblichen Formaten (PNG, JPG, TIFF). Zudem exportiert es in zahlreiche Vektorformate sowie in PNG.

Die Form-Werkzeuge bieten grafische Primitive wie Rechteck, Ellipse, Polygone und 3D-Figuren, die mit Konturen und Füllungen versehen werden können, darunter Texturen und Farbverläufe. Freihandzeichnungen mit der Maus oder dem Stift eines Grafiktablets verwandelt Inkscape in Pfade, deren Knoten individuell bewegt werden können, um die gewünschte Form zu erzeugen. Text und andere Objekte können an einem Pfad ausgerichtet werden.

Eine besondere Stärke von Inkscape ist seine Fähigkeit, Rastergrafiken zu vektorisieren, also Pixelinformationen in einen Satz von Kurven umzuwandeln. Damit lässt sich z.B. aus einem Foto eine Zeichnung machen oder eine eingescannte Zeichnung in hoher Qualität rekonstruieren. Zahlreiche Filter und Erweiterungen bieten nützliche Funktionen und spektakuläre Effekte.

Für die OS X-Version muss zunächst die X11-Window-Umgebung (XQUARTZ) installiert werden, um Inkscape benutzen zu können.

EINSCHÄTZUNG: INKSCAPE eignet sich als Universalwerkzeug für Vektorgrafiken. Damit lassen sich Logos und Diagramme, Textkunst (z.B. Titel für Schülerzeitung oder Videos), aber auch Illustrationen und Comics erstellen. Designbegeisterte haben das Handbuch schnell gelesen und werden Freude haben, die Funktionen des Programms zu entdecken.

► *Siehe auch das Werkzeugporträt zu INKSCAPE.*

KRITA

Portable App: Nein **Benutzeroberfläche deutsch:** Ja **Lizenz:** GPL 2

KRITA (schwedisch für Kreide) ist ein Mal- und Zeichenprogramm, das auf Illustrationen, Texturen und visuelle Effekte in der Postproduktion von Filmen spezialisiert ist. KRITA bietet eine umfassende Farbverwaltung, zahlreiche Pinselformen und Transformationen sowie die Bearbeitung von HDR-Fotos. Assistenten unterstützen bei Perspektiven oder extremem Weitwinkel und stabilisieren die Linienführung bei Freihandzeichnungen. Animationswerkzeuge sowie verbesserte Werkzeuge für Text und Vektorgrafik sind in der Entwicklung. KRITA ist Teil der CALLIGRA SUITE, die außerdem Textverarbeitung, Tabellenkalkulation, Datenbank, Projektmanagement unter anderem enthält.

Weblinks:

Projektseite:

➤ <https://inkscape.org/de>

portable App:

➤ https://inkscape.org/en/gallery/item/4460/InkscapePortable_0.91.paf.exe

Weiteren Informationen:

Handbuch:

➤ <https://inkscape.org/de/lernen>

Video-Tutorials (englisch):

➤ <https://inkscape.org/de/lernen/videos>

Wikipedia:

➤ <https://de.wikipedia.org/wiki/Inkscape>

Weblink:

Projektseite:

➤ <https://krita.org>

Weiteren Informationen:

Dokumentation und

Video-Tutorials (englisch):

➤ <https://krita.org/learn>

Video-Tutorials:

➤ <https://youtu.be/joywLfsXW8k>

➤ https://youtu.be/QJoYmn_fN44

Beispiel für Erstellung einer Illustration mit Krita:

➤ <https://youtu.be/fKtMVVDbBmw>

Wikipedia:

➤ <https://de.wikipedia.org/wiki/Krita>

Weblinks:

Projektseite (englisch):

➤ <http://librecad.org/cms/home.html>

portable App:

➤ http://portableapps.com/apps/graphics_pictures/librecad-portable

Weiteren Informationen:

Video-Tutorial:

➤ https://youtu.be/SUImeNSQpoU?list=PLKh3tkUblv17uqt1sEmQl9nA_bc4Qrz

Weblinks:

Projektseite:

➤ <https://de.libreoffice.org>

Portable App:

➤ http://portableapps.com/de/apps/office/libreoffice_portable

EINSCHÄTZUNG: Die Einsatzmöglichkeiten von KRITA reichen von Illustrationen für Webseiten oder Drucksachen über das Erstellen von Mangas oder Karikaturen bis zur Frame-für-Frame-Bearbeitung von Videos, um Verfremdungen oder besondere Effekte zu erzielen.

LIBRECAD

Portable App: Ja **Benutzeroberfläche deutsch:** Ja **Lizenz:** GPL 2

LIBRECAD ist ein 2D-► *CAD-Programm* zum Zeichnen und Konstruieren. Mit ihm lassen sich technische Zeichnungen als Vektorgrafiken erstellen. Es enthält die klassischen Elemente technischer Zeichenprogramme: Linien, Radien, Bemaßungen und Schraffuren, die über eine Konsole eingegeben und manipuliert werden können. Aus speziellen Bibliotheken können Objekte aus den Bereichen Architektur, Elektrotechnik und Elektronik eingefügt werden.

Mit LIBRECAD können technische Zeichnungen in zwei Dimensionen erstellt werden. Damit bietet es sich zur Verwendung im Fach Technik zum Konstruieren von Werkstücken, aber auch in Kunst zum Entwurf von Gebäuden an.

EINSCHÄTZUNG: Der Einstieg in die Materie »Technisches Zeichnen und computergestütztes Konstruieren« von der 2D- zur 3D-Technik gelingt mit LIBRECAD sehr gut. Die Steuerung und Zeichnungsgestaltung über die Konsole bildet die Brücke zu Anwendungen aus der professionellen Produktgestaltung und dem Erstellen von ► *G-Code* zur Programmierung von ► *CNC*-Maschinen wie Lasern, Fräsen und Druckern.

LIBREOFFICE DRAW

Portable App: Ja **Benutzeroberfläche deutsch:** Ja **Lizenz:** MPL, LGPL 3+, GPL 3+

LIBREOFFICE DRAW ist das Zeichenprogramm aus dem LIBREOFFICE-Programmpaket. Zeichnungen werden mit Elementen wie Textfeldern, Linien, geometrischen Formen und standardisierten Symbolen erstellt. Diese Elemente sind vielfältig zu bearbeiten und mit Bildern zu kombinieren. Bilder können mit grundlegenden Funktionen wie der Einstellung von Helligkeit, Kontrast usw. manipuliert werden, einzelne Bearbeitungsfunktionen entsprechen denen von ► *CAD-Programmen*. Damit ist es zwischen einem ► *DTP*-, einem Bildbearbeitungs- und einem sehr einfachen CAD-Programm

angesiedelt. LIBREOFFICE DRAW ist in der Schule durch seine Nähe zur Textverarbeitung WRITER sehr leicht einzusetzen.

EINSCHÄTZUNG: LIBREOFFICE DRAW ist als Allround-Programm zu verstehen: Während größere Layout-Projekte eher mit Scribus zu verwirklichen sind, für Bildbearbeitung GIMP und für CAD z.B. LIBRECAD zu empfehlen sind, besticht Draw durch die einfachen Funktionen für kleine Projekte wie Plakate und Flyer. Mit der Exportfunktion, mit der die erstellten Dateien in vielen Bildformaten wie JPEG, PNG oder GIF abzuspeichern sind, lassen sich sehr einfach Logos oder Collagen für die Schulhomepage erstellen. Die grundlegende Funktion eines 2D-CAD-Programms, unter anderem mit der Möglichkeit, Objekte zu bemaßen, erleichtert den Einstieg in das Technische Zeichnen.

PINTA

Portable App: Nein **Benutzeroberfläche deutsch:** Ja **Lizenz:** MIT

PINTA ist ein einfaches Mal- und Zeichenprogramm. Es bietet die üblichen Werkzeuge und Effekte, wie Pinsel, Stift, Radiergummi, Formen, Farbsteuerung usw., mit denen Fotos und Pixelgrafiken bearbeitet werden können. Die Vorteile dieser Software gegenüber komplexeren Werkzeugen wie GIMP sind die schnellere Einarbeitungszeit, die geringeren Ansprüche an die Rechenleistung des Computers und die Größe der Software. Für die OS X-Version muss zuerst ► *MONO* installiert werden, die freie Implementierung von ► *Microsoft .NET Framework*.

EINSCHÄTZUNG: PINTA ist für einfache Aufgaben geeignet, z.B. Bilder beschneiden oder auf einem Bildschirmfoto Pfeile einfügen, aber auch für künstlerische Zeichnungen oder Comics. Für komplexere Bildbearbeitung empfiehlt sich GIMP.

RAWTHERAPEE

Portable App: Ja **Benutzeroberfläche deutsch:** Ja **Lizenz:** GPL 3

RAWTHERAPEE ist ein Programm für die Konvertierung und Bearbeitung von Digitalfotos in ► *RAW*-Formaten, also den Rohdaten des Bildsensors von digitalen Spiegelreflexkameras (DSLR) und manchen Kompaktkameras. Anders als Bildformate wie PNG oder JPG sind RAW-Bilder unkomprimiert und noch nicht in einen bestimmten Farbraum transformiert; sie enthalten somit die volle ursprüngliche Farbtiefe.

EINSCHÄTZUNG: RAWTHERAPEE bietet Funktionen für Weiß- und Schwarzabgleich, die Verbesserung von Kontrast und Detail, Rauschfilterung und die Verhinderung von Bildartefakten. Es dient als Vorstufe für die Entwicklung von Fotos in hoher Qualität, die im Anschluss mit einem Bildbearbei-

Weiteren Informationen:

LIBREOFFICE-Handbuch:
➤ <https://de.libreoffice.org/get-help/documentation>

Video-Tutorial:

➤ <https://youtu.be/gPpd3tTZ2uc>

Weblink:

Projektseite:
➤ <https://pinta-project.com>

Weiteren Informationen:

Video-Tutorials (englisch)
(jeweils 1 Min.):
➤ <https://pinta-project.com/pintaproject/pinta/howto/pinta-on-web>

Wikipedia:

➤ https://de.wikipedia.org/wiki/Pinta_%28Software%29

Weblinks:

Projektseite:
➤ <http://rawtherapee.com>

portable App:

➤ http://portableapps.com/apps/graphics_pictures/rawtherapee-portable

Weiteren Informationen:

Handbuch (Version 4.0.10):

➤ <https://docs.google.com/document/d/1zLqYclOaZnovooj3QLRwUjToR9ixeaoijFy09xZvzCI>

Dokumentation RawPedia

(englisch):

➤ <http://50.87.144.65/~rt/w>

Forum (englisch):

➤ <https://discuss.pixls.us/c/software/rawtherapee>

Wikipedia:

➤ <https://de.wikipedia.org/wiki/RawTherapee>

Weblinks:

Projektseite:

➤ <http://synfig.org>

portable App:

➤ <http://winpenpack.com/main/download.php?view.1275> und
➤ <http://portableapps.com/node/32918>

Weiteren Informationen:

Dokumentation

(z.T. deutsch):

➤ http://wiki.synfig.org/Main_Page/de

Video-Tutorial:

➤ <https://gumroad.com/l/synfig-training-1-de>

Forum:

➤ <http://synfig.org/forums>

Wikipedia:

➤ <https://de.wikipedia.org/wiki/Synfig>

tungsprogramm wie GIMP weiterverarbeitet werden können. So können z.B. alle Fotos einer Publikation mit Profilen aufeinander abgestimmt werden. Das mächtige Programm gibt Fotoenthusiast*innen eine sehr weitreichende Kontrolle über ihre Werke.

SYNFIG

Portable App: Ja **Benutzeroberfläche deutsch:** Ja **Lizenz:** GPL 3

SYNFIG ist ein 2D-Animationsprogramm. Ausgangspunkt ist ein Drehbuch mit Schlüsselbildern, die in einem Vektor- oder Rastergrafikprogramm erstellt werden. SYNFIG generiert die Einzelbilder zwischen zwei Schlüsselbildern, sodass eine flüssige Bewegung entsteht. Ein Knochensystem erlaubt es, z.B. die Körperteile einer Figur zu kontrollieren und die verwendeten Vektor- und Rastergrafiken entsprechend zu verformen. Mit einfachen Audiofunktionen lässt sich die Animation mit einem Soundtrack synchronisieren. Das Ergebnis kann in verschiedenen Videoformaten (z.B. DV, THEORA, MPEG), als Folge von Einzelbildern (z.B. PNG, OPENEXR) oder als animierte Grafik (GIF, MNG) ausgegeben werden.

EINSCHÄTZUNG: Mit SYNFIG lassen sich Videos erstellen, die komplexe Zusammenhänge visualisieren.

SCRIBUS

Portable App: Ja Benutzeroberfläche deutsch: Ja Lizenz: GPL 2

SCRIBUS ist ein Desktop-Publishing- (► *DTP*) oder Seitenlayout-Programm. Auf einer ausgewählten Formatseite lassen sich Text, Bild und Grafik in Rahmen und auf verschiedenen Ebenen frei platzieren. Formatierter Text in freien Formaten wie ► *ODT* oder ► *HTML* sowie Raster- und Vektorgrafiken in allen gängigen Formaten können eingelesen werden. SCRIBUS verfügt über typografische Funktionen wie Randausgleich, Unterschneidung und Grauwertoptimierung. Weitere Funktionen, wie Fußnoten und Marginalien, vertikale Keile und Absatzkontrolle, liegen in der aktuellen Entwicklerversion (Stand: Mitte 2016) bereits vor. Innerhalb von Scribus können spezielle Grafiken mithilfe externer Programme ► *gerendert* werden (z.B. L^AT_EX für Formelsatz und L^IL^YP_ON_D für Notensatz).

SCRIBUS exportiert in Raster- und Vektorgrafik, EPS und ► *PDF*. Es hat Stärken in der Druckvorlagenherstellung. Dazu gehört ein komplexes Farbmanagement mit mehr als 150 Farbpaletten, die Einrichtung von Schneidemarken, Farbbalken usw., eine automatische Druckvorlagenüberprüfung und die Ausgabe farbseparierter Dateien sowie in die genormten Formate, die von Druckereien oft gefordert werden (z.B. PDF/X-1a).

EINSCHÄTZUNG: Mit SCRIBUS lassen sich Dokumente für Web und Druck professionell gestalten. Von Visitenkarten über Einladungen, Plakate, Programmhefte und Broschüren bis zu Zeitschriften und Büchern ist alles möglich.

► *Siehe auch das Werkzeugporträt zu SCRIBUS.*

Weblinks:

Projektseite:

✂ <https://scribus.net>

Portable App:

✂ http://portableapps.com/apps/office/scribus_portable

Weiteren Informationen:

Dokumentation:

✂ <https://wiki.scribus.net/canvas/Hauptseite>

Tutorial:

✂ https://wiki.scribus.net/canvas/Einstieg_in_Scribus

How-to:

✂ https://wiki.scribus.net/canvas/Category:HOWTO_DE

Video-Tutorials:

✂ <https://www.youtube.com/playlist?list=PL103A931AE65DF703>

Wikipedia:

✂ <http://de.wikipedia.org/wiki/Scribus>

Audio und Video

Bei der Arbeit mit Audio- und Videoinformationen begegnen einem zahlreiche Formate. Dabei ist es nützlich, den Unterschied zwischen einem Codec- und einem Containerformat zu kennen und etwas über mögliche rechtliche Gefahren zu wissen.

Codecs

Ein analoges Signal (z.B. ein Audiosignal) hat einen kontinuierlichen, stufenlosen Verlauf. Wird es in diskrete Einheiten wie Buchstaben oder Morsecode umgesetzt, spricht man von Kodierung. Bei einer digitalen Kodierung misst zunächst ein Sensor (z.B. ein Mikrofon) die jeweilige physikalische Größe (z.B. den Schalldruck) in Form einer analogen elektrischen Spannung. Dieser Spannungsverlauf wird in regelmäßigen Zeitabständen abgetastet und der jeweilige diskrete Wert digital erfasst, um so das Audiosignal mithilfe eines Computers speichern, übertragen und verarbeiten zu können. Um den Klang wieder hör-

bar zu machen, wandelt ein dem Kodierer komplementärer Dekodierer den binären Datenstrom im selben Zeitverlauf zurück in analoge Signale, die über einen Lautsprecher ausgegeben werden können. Da Codierer und Decodierer zusammen gehören, spricht man meist von ► *Codecs*. Werden bereits kodierte Daten in einen anderen Code übersetzt, spricht man von Transkodierung oder Konvertierung.

Während analog gespeicherte Inhalte (z.B. auf einer Schallplatte) mit jedem Kopiervorgang an Qualität verlieren, können digitale Daten ggf. mithilfe von Fehlerkorrekturalgorithmen verlustfrei kopiert werden. Zudem können digitale Daten komprimiert, d.h. entbehrliche Information entfernt werden, um die Datenmenge und somit den benötigten Speicherplatz und die Übertragungszeit zu reduzieren. Man unterscheidet zwischen verlustfreier Kompression, bei der die Originaldaten vollständig aus den komprimierten Daten zurückgewonnen werden können, und verlustbehafteter Kompression, bei der ein Teil der Information verloren geht.

Für Audio gehören zu den bekanntesten verlustbehaftet komprimierenden Codecs MP3 und AAC der MPEG und das freie OGG VORBIS. Zu den verlustfrei komprimierenden Audiocodecs gehören AIFF und FLAC. Videocodecs komprimieren in der Regel verlustbehaftet: Dazu gehören MPEG-1, MPEG-2, MPEG-4 und WMV sowie das freie DivX und das freilizenzierte VP8 und VP9. Bilder lassen sich sowohl verlustfrei wie -behaftet komprimieren: mit JPEG, TIFF GIF und dem freien WebP sowie verlustfrei mit dem freien PNG.

Containerformat

Ein Verbunddokument wie ein Film oder eine Schülerzeitschrift in PDF besteht aus verschiedenen Datenströmen, die in einer Containerdatei zusammengeführt werden. Eine Filmdatei z.B. enthält neben Metadaten mindestens eine Video- und eine Audiodatei, die jeweils mit unterschiedlichen Codecs erstellt worden sein können. Überdies kann sie mehrere Audiodateien in verschiedenen Sprachen, Untertitel, Menüs usw. enthalten. Die Metadaten der Containerdatei sichern, dass jeder Datenstrom vom jeweiligen Codec dekodiert und alle synchron dargestellt werden. ► *Containerformate* sind auf bestimmte Anwendungen wie DVD, Webstreaming oder Digitalfernsehen optimiert.

WebM z.B. ist ein freies Containerformat, das in HTML5-Browsern Videos ohne zusätzliche Plugins wie den Flashplayer abspielt. Es kann Videos der Codecs VP8 und VP9 sowie Audios der Codecs Vorbis und Opus enthalten. Verbreitete proprietäre Audio-Video-Containerformate sind VOB, MP4, MPG, FLV, AVI und WMV. Die bekanntesten freien Containerformate sind Ogg, MKV und WebM.

Rechtliches

Die Unterscheidung zwischen freien und sonstigen Codecs verweist auf eine ähnliche Entwicklung wie bei proprietärer und Freier Software – mit dem

Unterschied, dass es hier um Patente geht. MPEG (MOVING PICTURE EXPERTS GROUP), eine 1988 gegründete Arbeitsgruppe der Internationalen Organisation für Normung (ISO), entwickelt Standards für Video- und Audiokodierung. Codecs können als technische Erfindungen patentiert sein. Für ihre Benutzung verlangen die Patentinhaber in der Regel Lizenzgebühren.

In der MPEG LICENSING ADMINISTRATION (MPEG LA) führen Unternehmen und Forschungseinrichtungen, die Technologien für Video- und Audiocodecs entwickeln, ihre Patente zu Patentpools zusammen, um sie einheitlich zu lizenzieren. Hersteller von Unterhaltungselektronik und Softwareprodukten und Dienstleister wie Fernsehveranstalter erwerben diese Lizenzen, sodass sich private und gewerbliche Nutzer nicht darum kümmern müssen.

Anders sieht es für Freie-Software-Projekte aus: Wenige könnten für Produkte, die kostenlos verbreitet werden, die hohen Lizenzgebühren bezahlen. Zwar gibt es für viele patentierte Codecs freie Implementierungen, aber anders als im Urheberrecht, das den Nachbau von Funktionen mit eigenen Mitteln nicht verbietet und für die Entwicklung kompatibler Produkte sogar ausdrücklich erlaubt, verbieten Patente genau diesen Nachbau der patentierten Funktionen.

► **FFMPEG** ist das zentrale Freie-Software-Projekt für Codecs. Als eine Art Schweizer Messer der freien audiovisuellen Software wird FFMPEG unter anderem von den hier vorgestellten Programmen AUDACITY, VLC, AVIDEMUX und SHOTCUT verwendet. Seine Bibliothek LIBAVCODEC enthält über 50 Video- und 30 Audiocodecs, größtenteils völlig neu geschrieben und weitestgehend konform mit MPEG- und anderen Standards. Damit hängt die Gefahr von Patentklagen über dem Projekt. Auch Projekte wie OGG, das eigene Codecs entwickelt, können leicht unwillentlich und unwissentlich Patente verletzen. Audacity kommt daher von Hause aus nur mit freien Codecs, erlaubt es den Nutzer*innen aber, externe Bibliotheken wie LAME (eine freie Implementierung von MP3) und FFMPEG hinzu zu installieren.

Auf seiner Webseite stellt FFMPEG klar, dass die reinen Nutzer*innen freier, wahrscheinlich patentverletzender Codecs für den privaten und schulischen Gebrauch sich keine Sorgen machen sollten.¹² Produzenten kommerzieller Softwareprodukte, die FFMPEG verwenden, hätten jedoch, sobald sie Geld verdienen, Lizenzforderungen der MPEG LA erhalten.

12 ➤ <https://ffmpeg.org/legal.html>

AUDACITY

Portable App: Ja **Benutzeroberfläche deutsch:** Ja **Lizenz:** GPL 2

AUDACITY ist ein Programm für die Aufnahme und Bearbeitung von Klanginformationen. Über Mikrofon oder Line-in lassen sich damit Interviews, Musik und andere Töne auf Festplatte aufnehmen. AUDACITY importiert und exportiert alle unkomprimierten und patentfreien Audioformate. Zur

Weblinks:
Projektseite:
➤ <http://audacityteam.org>

portable App:

➤ http://portableapps.com/de/apps/music_video/audacity_portable

Weiteren Informationen:

Handbuch und Tutorials (englisch):

➤ <http://manual.audacityteam.org>

Video-Tutorials

(Version 1.3.6):

➤ <https://vimeo.com/album/45014>

Forum:

➤ <http://audacity-forum.de/forum>

Wikipedia:

➤ <https://de.wikipedia.org/wiki/Audacity>

Weblinks:

Projektseite:

➤ <http://avidemux.sourceforge.net>

portable App:

➤ http://sourceforge.net/projects/prapper/files/avidemux/AvidemuxPortable_2.5.5_Dev_Test_1.paf.exe

Weiteren Informationen:

Handbuch (englisch):

➤ <http://avidemux.org/admWiki/doku.php?id=build:doctop>

Video-Tutorial:

➤ <https://youtu.be/HoutsdthP6Y>

Wikipedia:

➤ <https://de.wikipedia.org/wiki/Avidemux>

Unterstützung weiterer Formate können externe Bibliotheken wie LAME für MP3 und FFmpeg für weitere MPEG-Formate installiert werden.

Die Bearbeitungsmöglichkeiten reichen vom einfachen Trimmen eines Interviews über Mehrspurkompositionen bis zu einer Fülle von Klangeffekten und Möglichkeiten, Klänge zu analysieren und zu verbessern. Einschränkung: In einer Audiospur nach Gehör zu navigieren ist nicht möglich. Auch Effekte werden nicht in Echtzeit umgesetzt, sondern müssen erst berechnet werden, bevor sie hörbar sind.

AUDACITY ist auf Barrierefreiheit optimiert: Es ist fast vollständig über die Tastatur zu steuern; es gibt Werkzeuge, die sichern, dass ein Sprachsignal sich gemäß den Anforderungen der Barrierefreiheit vom Hintergrund abhebt; auf Windows werden die meisten Screen-Reader unterstützt; für MAC OSX ist die Version 2.1.1 auf Barrierefreiheit ausgerichtet.

EINSCHÄTZUNG: AUDACITY eignet sich als Universalwerkzeug für alle Audioarbeiten. Mit einem Rekorder aufgenommene Interviews und O-Töne können mit AUDACITY geschnitten, mit Moderation versehen und mit Musik unterlegt werden, um so eine Radiosendung oder einen Podcast zu erstellen. Bei einer Videoproduktion kann die Audiospur mit AUDACITY verbessert und abgemischt werden. Zudem bietet das Programm Möglichkeiten, im Physik- und Musikunterricht mit Klängen zu experimentieren.

► *Siehe auch das Werkzeugporträt zu AUDACITY.*

AVIDEMUX

Portable App: Ja Benutzeroberfläche deutsch: Ja Lizenz: GPL 2

AVIDEMUX ist ein Videobearbeitungsprogramm. Es eignet sich besonders, um Videos zu trimmen (Anfang und Ende beschneiden), ohne sie erneut zu enkodieren, sie mit Filtern zu verändern oder in andere Formate zu **transkodieren**. Dank FFmpeg unterstützt es alle gängigen Formate. Mithilfe von Skripts lassen sich Dateien automatisiert bearbeiten.

EINSCHÄTZUNG: AVIDEMUX eignet sich, um ohne Qualitätsverluste Abschnitte aus Videos (z.B. Werbung) zu entfernen oder umgekehrt zu extrahieren. Für komplexe Arbeiten ist ein Videoschnittprogramm wie SHOTCUT besser geeignet.

KODI

Portable App: Ja **Benutzeroberfläche deutsch:** Ja **Lizenz:** GPL

KODI ist ein Medienzentrum, das erlaubt, vorhandene Sammlungen von Videos, Audios und Bildern zu katalogisieren, zu organisieren und abzuspielen. Alle physischen Medien einschließlich unverschlüsselter Blu-Rays lassen sich damit abspielen, ebenso wie die gängigen Streamingformate. KODI ruft dabei zusätzliche Informationen wie Darstellerlisten und Untertitel aus dem Internet ab. Mithilfe von Plugins kann Kodi auf Videoportale wie YouTube und Hulu sowie auf die Mediatheken von ARD und ZDF zugreifen. Die Oberfläche gleicht einem Smart-TV und ist neben Tastatur und Maus auch mit Fernbedienung, WIIOTE und Smartphone zu bedienen. Das Programm wurde ursprünglich als XBOX MEDIA CENTER (XBMC) für die Spielkonsole XBOX entwickelt. Mit dem Wechsel von der XBOX zu einer plattformunabhängigen Version wurde es in Kodi umbenannt.

EINSCHÄTZUNG: Mit KODI lassen sich private und unterrichtsbezogene audiovisuelle Mediensammlungen organisieren und auf ansprechende Weise navigieren und darstellen.

MEDIATHEKVIEW

Portable App: Ja **Benutzeroberfläche deutsch:** Ja **Lizenz:** GPL 3

MEDIATHEKVIEW ist ein ►*JAVA*-Programm, mit dem sich die Mediatheken der deutschsprachigen öffentlich-rechtlichen Sender in Deutschland, Österreich und der Schweiz einheitlich durchsuchen lassen. Dazu lädt man zunächst die aktuelle Filmliste herunter, die sämtliche verfügbare Sendungen enthält. Die Liste lässt sich nach Sender, Titel, Zeitraum und Stichwörtern durchsuchen. Mithilfe eines externen Players wie VLC lassen sich die Sendungen abspielen und herunterladen. Regelmäßige Sendungen und Serien können abonniert und automatisch auf der Festplatte gespeichert werden.

EINSCHÄTZUNG: Wer Fernsehinhalte schätzt, sie aber nicht mehr linear wahrnimmt, findet in MEDIATHEKVIEW ein Werkzeug, um aus Tausenden Sendungen die interessantesten herauszusuchen und anzuschauen. Für den Unterrichtsgebrauch lassen sich damit Sammlungen von Sendungen anlegen. Für den Bereich Medienkompetenz und Fernsehkritik können Nachrichten, Magazinbeiträge, Talkshows usw. zu verschiedenen Themen für Besprechungen und Analysen zusammengestellt werden.

Weblinks:

Projektseite:

➤ <https://kodi.tv>

portable App:

➤ <http://kodi-hilfe.tk/Windows/Kodi%20Installieren.html>

Weiteren Informationen:

Kodi-Wiki (englisch):

➤ http://kodi.wiki/view/General_topics

Forum (englisch):

➤ <http://forum.kodi.tv>

Wikipedia:

➤ https://de.wikipedia.org/wiki/Kodi_%28Software%29

Weblinks:

Projektseite:

➤ <http://zdfmediathk.sourceforge.net>

Portable App:

➤ <https://sourceforge.net/p/zdfmediathk/wiki/Anleitung/#starten-im-portablen-modus-mediathekview-portable>

Weiteren Informationen:

Wiki mit Anleitungen:

➤ <https://sourceforge.net/p/zdfmediathk/wiki>

Weblinks:

Projektseite:

➤ <https://shotcut.org>

portable App:

➤ <https://archive.org/details/shotcutportable>**Weiteren Informationen:**

Kurzanleitung:

➤ https://shotcutapp.com/howtos/getting-started/Erste_Schritte_mit_Shotcut.pdf

Video-Tutorial:

➤ <https://youtu.be/ixFrr39aSSs>

Video-Tutorials (englisch):

➤ <https://shotcut.org/tutorials/>

Wikipedia:

➤ <https://en.wikipedia.org/wiki/Shotcut>**Weblinks:**

Projektseite:

➤ <http://videolan.org/vlc>

Webseite:

➤ <http://vlc.de>

portable App:

➤ http://portableapps.com/de/apps/music_video/vlc_portable**Weiteren Informationen:**

Handbuch:

➤ <http://vlc.de/VLC-Handbuch.pdf>

Forum:

➤ <https://vlc-forum.de>

Wikipedia:

➤ https://de.wikipedia.org/wiki/VLC_media_player

SHOTCUT

Portable App: Ja **Benutzeroberfläche deutsch:** Ja **Lizenz:** GPL 3

SHOTCUT ist ein Videobearbeitungsprogramm. Es speichert Daten von Kameras und Mikrofonen und liest dank FFmpeg Dateien in allen gängigen Video-, Audio- und Bildformaten und in einer Auflösung bis 4K. Die Materialien werden auf den Spuren einer Timeline angeordnet und können geschnitten (Übergänge, Blenden) und nachbearbeitet (Farb- und Tonkorrekturen, zahlreiche Filter) werden. Shotcut gibt es nur in 64-Bit-Versionen.

EINSCHÄTZUNG: SHOTCUT ist geeignet für die Erstellung von einfachen und komplexeren Videoprojekten. Für das einfache Trimmen eines Videos (Beschneiden von Anfang und Ende) bietet sich AVIDEMUX an.

VLC

Portable App: Ja **Benutzeroberfläche deutsch:** Ja **Lizenz:** GPL 2, LGPL 2.1

VLC ist ein Multimediaplayer, der neben fast allen Dateiformaten auch DVDs, Audio-CDs, Kameras und Streams abspielt. Außerdem kann VLC verwendet werden, um die unterstützten Video- und Audioformate ineinander zu konvertieren sowie Mediendateien zu streamen.

EINSCHÄTZUNG: VLC eignet sich als Universalwerkzeug für die Darstellung von Video- und Audio-Informationen aus sämtlichen Quellen. Bildschirmfotos von Schlüsselszenen helfen, Videowerke zu analysieren und zu besprechen.

Auf dem eigenen Rechner erlaubt VLC schnellen Zugriff auf alle Video-, Audio- und Bilddateien auf der Festplatte, die sich in Wiedergabelisten und Medienbibliotheken organisieren lassen. Auch Podcasts und andere Streams lassen sich abonnieren. Dank Tastaturbedienung und optionaler Fußtaste (Pause, Vor- und Zurückspringen, langsames Abspielen) ist VLC gut geeignet, um Audio- und Videomaterial zu transkribieren.

Internet und Kommunikation

Im Folgenden werden neben einem E-Mail-Programm unter anderem auch zwei Webbrowser vorgestellt, also Programme zur Darstellung von Webseiten sowie Dokumenten und Daten, die es ermöglichen im World Wide Web (WWW) zu surfen.

BLUEFISH

Portable App: Ja **Benutzeroberfläche deutsch:** Nein **Lizenz:** GPL

BLUEFISH ist ein Editor, der speziell für die Erstellung und Bearbeitung von Dateien im Zusammenhang mit Webseiten geeignet ist. Dazu bietet er spezielle Sprachunterstützung für das HTML-Format, PHP und diverse andere Programmiersprachen wie JAVA, PERL, PYTHON, RUBY, C und PASCAL.

Bluefish offeriert für die üblichen HTML-Bausteine eine Symbolleiste und Dialoge für die Eigenschaften dieser Bausteine. In der Regel sind die auf dem Computer installierten Browser programmintern verknüpft. Die von den Schüler*innen erarbeiteten Änderungen an den Webseitendateien können sofort sichtbar gemacht werden.

EINSCHÄTZUNG: Moderne Webseiten basieren seit Beginn des World Wide Web auf einem HTML-Grundgerüst. Zur Vermittlung der HTML-Sprache im Unterricht erscheint BLUEFISH ideal, die Hervorhebungsfunktion für die unterschiedlichen Sprachen hilft bei der Vermittlung. Da auch aktuelle Techniken wie CSS und HTML5 implementiert sind, bietet sich eine große Bandbreite an Inhalten für den Informatikunterricht.

Weblink:

Projektseite (englisch):
<http://bluefish.openoffice.nl/index.html>

Weiteren Informationen:

Bluefish-Wiki (englisch):
http://bfwiki.tellefsen.net/index.php/Main_Page

CHROMIUM/CHROME

Portable App: Ja **Benutzeroberfläche deutsch:** Ja **Lizenz:** BSD

CHROME ist ein von der Firma GOOGLE entwickelter Browser. Er wird im Projekt CHROMIUM quelloffen entwickelt und um weitere, nicht-freie Funktionen ergänzt, die im Wesentlichen die Installation individualisieren. Zudem ist FLASH standardmäßig integriert. Das Projekt CHROMIUM gibt dem vollständig quelloffenen Browser-Pendant CHROMIUM seinen Namen. Neben der Unterstützung aller modernen Webtechnologien haben die installierbaren Add-ons (Erweiterungen) im schulischen Kontext besondere Bedeutung.

CHROMIUM ist für die meisten ► *LINUX-Distributionen* aus den Paketquellen installierbar, für die Betriebssysteme WINDOWS und MAC OS nur mit einigem Aufwand zu installieren. Es gibt eine Vielzahl von Erweiterungen und Apps, die über den für CHROME wie für CHROMIUM verfügbaren WEB STORE von GOOGLE installiert werden können. Dabei handelt es sich

Weblink:

Projektseite (englisch):
<https://chromium.org>

Weiteren Informationen:

Darstellung der Unterschiede zum Browser Chrome:
[https://de.wikipedia.org/wiki/Chromium_\(Browser\)#Unterschiede_zu_Google_Chrome](https://de.wikipedia.org/wiki/Chromium_(Browser)#Unterschiede_zu_Google_Chrome)

teilweise um Verweise auf Webseiten, aber auch um Apps, die denen von Smartphones bzw. installierten Programmen ähneln. Die Erweiterungen sind teilweise frei, teilweise proprietär und sowohl kostenpflichtig als auch kostenlos zu beziehen.

EINSCHÄTZUNG: Browserbasierte Apps und Erweiterungen bieten die Möglichkeit, betriebssystemunabhängig Programme zu nutzen. Die installierbaren Erweiterungen im GOOGLE WEB STORE befinden sich in ständiger Entwicklung. Ein Schwerpunkt der angebotenen Apps liegt im technischen Bereich (z.B. in der Programmierung); durch das vorwiegend englischsprachige Angebot durchaus auch im Fach Englisch einsetzbar.

FIREFOX (alle Fächer)

Portable App: Ja **Benutzeroberfläche deutsch:** Ja **Lizenz:** MPL, GPL, LGPL

FIREFOX ist ein Internetbrowser, der alle gängigen Standards unterstützt und sich durch Plugins um viele nützliche Funktionen erweitern lässt. Internetbrowser werden heute nicht mehr nur zum Betrachten von Webseiten verwendet. Auch wenn dies immer noch das Haupteinsatzgebiet ist, werden sie immer mehr zu einer Plattform, um Webanwendungen auszuführen. Durch moderne Standards wie WEBRTC eignen sie sich als Plattform für Videotelefonie, Chats zwischen zwei Personen oder in größeren Gruppen, wodurch sie ein sehr nützliches Werkzeug für Gruppenarbeiten oder »virtuelle Klassenzimmer« sind. FIREFOX ist zudem die Basis für weitere Browser.

EINSCHÄTZUNG: FIREFOX ist auf so gut wie jeder Plattform verfügbar. Die gute Unterstützung von Webstandards und die einfache Erweiterbarkeit durch Plugins sorgen dafür, dass der Browser in vielen Bereichen des Unterrichts einsetzbar ist. Hinweis: Die Plugins sind nicht immer Freie Software.

► *Siehe auch das Werkzeugporträt zu FIREFOX.*

THUNDERBIRD

Portable App: Ja **Benutzeroberfläche deutsch:** Ja **Lizenz:** MPL, GPL, LGPL

THUNDERBIRD ist ein E-Mail-Programm, das es ermöglicht E-Mails zu lesen, zu schreiben und zu versenden. Als Desktop-Anwendung kann diese viele Aufgaben auch ohne eine bestehende Internetverbindung erledigen. THUNDERBIRD ist für alle gängigen Betriebssysteme verfügbar: LINUX, MACOS, WINDOWS.

Das E-Mail Programm verfügt über eine Schnittstelle für Erweiterungen (Plugins). Dadurch lassen sich viele nützliche Funktionen (wie z.B.

Weblinks:

Projektseite:

➤ <https://mozilla.org/de/firefox>

Portable App:

➤ http://portableapps.com/de/apps/internet/firefox_portable

Weiteren Informationen:

Erweiterungen:

➤ <https://addons.mozilla.org/de/firefox>

Tor-Browser:

➤ <https://torproject.org/projects/torbrowser.html>

Wikipedia:

➤ https://de.wikipedia.org/wiki/Mozilla_Firefox

Weblinks:

Projektseite:

➤ <https://mozilla.org/de/thunderbird>

portable App:

➤ http://portableapps.com/de/apps/internet/thunderbird_portable

Ende-zu-Ende-Verschlüsselung mit ▶ *GnuPG*) realisieren. Darüber hinaus stehen Erweiterungen zur Einbindung von Kalendern und Adressbüchern zur Verfügung, wodurch THUNDERBIRD zu einer vollständigen ▶ *Groupware*-Lösung wird.

EINSCHÄTZUNG: THUNDERBIRD zeichnet sich dadurch aus, dass es auf mehreren Betriebssystemen zur Verfügung steht. Zudem lässt es sich mit vielen nützlichen Funktionen erweitern, die weit über das Schreiben von E-Mails hinausgehen.

Weiteren Informationen:

Erweiterungen:

➤ <https://addons.mozilla.org/de/thunderbird>

Wikipedia:

➤ https://de.wikipedia.org/wiki/Mozilla_Thunderbird

Unterrichtsunterstützung und Fächerspezifisches

Freie Software bietet diverse Anwendungsmöglichkeiten in der Unterstützung des Unterrichts für alle Fächer sowohl für Lehrer*innen als auch jüngere und/oder ältere Schüler*innen in allen Klassenstufen. Teilweise stehen ausgearbeitete Unterrichtsreihen und Video-Tutorials für die Programme zur Verfügung. Die Themen sind vielfältig: Fachspezifisch für Mathematik, Informatik, Chemie, Musik und Geografie (spezielle Mathematiksoftware, Molekül- und Kartendarstellungen, grafische Programmiersprachen, Notensatzprogramm) sowie fachübergreifend (zum Erstellen von Mindmaps oder E-Learning-Content, für die Arbeit mit interaktiven Whiteboards, als elektronische Lernkartei, Simulation oder Lernsoftware).

EXEARNING (alle Fächer)

Portable App: Ja **Benutzeroberfläche deutsch:** Ja **Lizenz:** EL v2.1: GPL 2+

Mit dem e-Learning-Autorentool EXEARNING (eLearning XHTML editor, kurz eXe) können Texte leicht mit Webelementen kombiniert und zu Lernaktivitäten ergänzt werden. Das Ergebnis kann als (lokale) Webseite verwendet oder in ein ▶ *Lern-Management-System (LMS)* wie ▶ *Moodle* exportiert werden. Die Bedienung ist gut strukturiert, es muss jedoch mit mehreren Stunden Lernaufwand gerechnet werden. Die relativ kleine Anwendung verfügt über eine deutsche Benutzeroberfläche. Die Hilfe-Funktion steht nur online zur Verfügung. Aktive Elemente wie ▶ *JAVA-Applets*, Wiki-Artikel, aber auch Multimediadateien wie Videos können eingebunden werden. Kollaborationsfunktionen fehlen. Die Anwendung ist weit verbreitet, weshalb es eine Vielzahl an Einführungsvideos und Tutorials gibt. Zu eXe gibt es mehrere Anschlussprojekte, die weitere Features ergänzen, ein Beispiel ist »The new eXeLearning«.

EINSCHÄTZUNG: Die Beherrschung von EXEARNING lohnt sich, wenn man E-Learning-Content produzieren will. Es ist allem für Lehrkräfte geeignet, in längeren Projekten aber auch für ältere Schüler*innen handhabbar.

▶ *Siehe auch das Werkzeugporträt zu eXeLearning.*

Weblinks:

Projektseite v1.x:

➤ <http://exelearning.org>

Projektseite v2.x:

➤ <http://exelearning.net>

portable App (Version 1):

➤ <https://sourceforge.net/projects/exe/files/eXe-1.04/exe-ready2run-1.04.exe/download>

Weiteren Informationen:

Tutorials:

➤ <http://exelearning-tutorials.de> und

➤ <http://paddelhannes.de/exetutorial-de/index.html>

Weblink:

Projektseite:

➤ <http://freemind.sourceforge.net/wiki/index.php/Download>

portable App:

➤ <http://portableapps.com/node/10952>

Weiteren Informationen:

Einführung:

➤ http://lehrerfortbildung-bw.de/werkstatt/praes/free-maps/pix/freemind_intro.pdf

Weblink:

Projektseite:

➤ <http://gcompris.net/index-de.html>

Weiteren Informationen:

Wikipedia:

➤ <https://de.wikipedia.org/wiki/GCompris>

FREEMIND (alle Fächer)**Portable App:** Ja **Benutzeroberfläche deutsch:** Ja **Lizenz:** GPL

Mit FREEMIND (FM) können von Lehrer*innen und Schüler*innen schnell Mindmaps in Einzelarbeit erstellt werden. Die fertigen Maps können in diversen Formaten sowie als Vektorbild (SVG-Format) exportiert werden. So können die Maps leicht in anderen Programmen skaliert und weiter bearbeitet werden.

Die Bedienung lässt sich in Minuten erlernen, die Anwendung ist übersichtlich. Es gibt eine deutsche Benutzeroberfläche und die Hilfe ist lokal verfügbar. Allerdings muss JAVA installiert sein. Da die Anwendung recht verbreitet ist, finden sich im Web Vorlagen und Beispieldokumente sowie Einführungsvideos und Tutorials. Die Gestaltungsmöglichkeiten sind jedoch begrenzt und Kollaboration ist nicht vorgesehen.

EINSCHÄTZUNG: Die einfache Bedienung und die übersichtliche Anzahl an Optionen führt zu kurzer Einarbeitungszeit und schnellen Erfolgen. Die Konzentration auf die Inhalte fällt daher leicht. FREEMIND ist ab Klasse 5 gut geeignet.

GCOMPRIS (Grundschule)**Portable App:** Nein **Benutzeroberfläche deutsch:** Ja **Lizenz:** GPL 3

GCOMPRIS ist eine Lernsoftware, die aus einer Sammlung von über 100 Aktivitäten besteht. Mit ihnen können zum einen die Grundfunktionen des Computers eingeübt werden (Bedienung von Maus, Übungen zum Tastaturlayout), zum anderen gibt es Übungen, die den Fächern Deutsch/Sprachen, Mathematik, Geografie und grundständigem Lernen (Ablesen der Uhrzeit usw.) zuzuordnen sind. Viele Aktivitäten animieren durch einen ansteigenden Schwierigkeitsgrad und positive Rückmeldungen bei erfolgreichen Übungen.

Die Aktivitäten können als einzelne Übungen im Unterricht eingebaut werden oder evtl. auch als Entdeckungsreise gestaltet sein, indem den Schüler*innen das Programm zur individuellen Beschäftigung zur Verfügung gestellt wird.

EINSCHÄTZUNG: Die Aktivitäten sind überwiegend für die Primarstufe oder die Förderschule geeignet. Sie haben eher spielerischen Charakter und sind nicht an einem Lehrplan orientiert, deswegen sollten zunächst die Lerninhalte klar definiert und dann passende Übungen ausgewählt werden.

HINWEIS: Für das Betriebssystem Windows stehen alle Aktivitäten nur nach einer kostenpflichtigen Aktivierung zur Verfügung, daneben gibt es auch Versionen für die mobilen Betriebssysteme.

OPEN-SANKORÉ (alle Fächer)

Portable App: Nein **Benutzeroberfläche deutsch:** Ja **Lizenz:** GPL 3, CC BY-SA

OPEN-SANKORÉ (OSÉ) ist primär für die Arbeit an elektronischen interaktiven Tafeln (»interaktive Whiteboards«, IWB) konzipiert. Mit OSÉ können Animationen, Eingabeübungen etc. für interaktive Tafeln erstellt werden. Das Programm funktioniert aber auch auf Tablets etc. Das Bild wird dann von einem angeschlossenen Beamer projiziert. Hauptvorteil ist, dass das Datenformat der erstellten Materialien offen ist und daher leicht von anderen Programmen gelesen werden kann. Es funktioniert mit allen bekannten IWBs.

Das relativ schlanke Programm kann auf fast allen Rechnern auch ohne angeschlossenes IWB installiert werden. Der Funktionsumfang ist geringer als der mit den IWBs gelieferten Software, reicht für die meisten Zwecke aber aus. Dadurch ist die Einarbeitungszeit kürzer. Nachteilig ist, dass die Software nur unvollständig auf Deutsch arbeitet und die meisten Materialien dazu auf Französisch und wenige auf Englisch sind. Dafür stehen die erstellten Materialien boardunabhängig zur Verfügung.

EINSCHÄTZUNG: Ein in Deutschland eher unbekanntes Programm, das im Prinzip die Herstellersoftware ersetzen kann. Die Einarbeitungszeit ist ähnlich groß wie bei den proprietären Lösungen und die große Community lässt erwarten, dass interessante Erweiterungen folgen werden. Bisher gibt es aber nur wenige deutschsprachige Materialien.

PAUKER (alle Fächer)

Portable App: Nein, nur Handy-Version **Benutzeroberfläche deutsch:** Ja **Lizenz:** GPL 3

PAUKER ist eine elektronische Lernkartei nach dem Sebastian-Leitner-System. Dabei werden Begriffe auf Karteikarten geschrieben und möglichst täglich geübt. Wurde ein Begriff fünf Mal hintereinander gewusst, so kann man davon ausgehen, dass der Begriff sicher beherrscht wird. Das Programm bildet eine solche Kartei elektronisch nach. Lernkarteien können leicht gemeinsam erstellt und ausgetauscht werden. PAUKER enthält keine multimedialen Elemente und benötigt eine installierte **Java-Laufzeitumgebung (JRE)**. Durch die Version für portable Geräte wie Handys können Wartezeiten produktiv genutzt werden.

EINSCHÄTZUNG: Auch für jüngere Schüler*innen ist Pauker zum Wiederholen gut geeignet. Es gibt relativ viele Karteien, die leicht erweitert werden können. Die übersichtliche Oberfläche benötigt fast keine Einarbeitungszeit.

Weblink:

Projektseite:

➤ <http://open-sankore.org>

Weiteren Informationen:

Video-Tutorial (englisch)
(5 Min.):

➤ <https://youtu.be/p2ygg0slnNE>

Video-Tutorials

(französisch):

➤ <http://open-sankore.org/en/tutorials>

Weblink:

Projektseite:

➤ <http://pauker.sourceforge.net/pauker.php?page=home&lang=de>

Weiteren Informationen:

Lektionen:

➤ <http://pauker.sourceforge.net/pauker.php?page=home&lang=de>

Weblink:

Projektseite:

➤ <http://energietycoon.de>**Weiteren Informationen:**

Video-Tutorial:

➤ <https://youtu.be/5YLowDQBpq8>**Weblinks:**

Projektseite:

➤ <https://geogebra.org>

Portable App:

➤ https://geogebra.org/manual/en/Reference:GeoGebra_Installation**Weiteren Informationen:**

Materialien des Landesbildungsservers Baden-Württemberg:

➤ <http://schule-bw.de/unterricht/faecher/mathematik/3material/sek1/geometrie/dyngeo/geogebra>

Die App ist im Werkzeugkasten »Lernen und Lehren mit Apps« beschrieben:

➤ <http://medien-in-die-schule.de/werkzeugkasten/werkzeugkasten-lernen-lehren-mit-apps/apps/geogebra>**ENERGIE TYCOON** (Simulationsspiel Politik, Wirtschaft etc.)**Portable App:** Nein **Benutzeroberfläche deutsch:** Ja **Lizenz:** MIT-Lizenz

ENERGIE TYCOON (ET) ist eine grafische Wirtschaftssimulation in Spielform mit strategischen Elementen. Das Programm kann einerseits als Wirtschaftssimulation verwendet werden, bei der der Einfluss von Strategie und Ressourceneinsatz auf das Ergebnis analysiert werden kann. Das Ziel besteht darin, die eigene Stadt mit Energie zu versorgen und möglichst viele Kund*innen zu gewinnen. Dazu müssen Kraftwerke gebaut werden, neue Technologien erforscht und es muss gehandelt werden. En passant können wirtschaftswissenschaftliche Grundlagen anschaulich vermittelt werden. Auf der Metaebene kann das Programm auch dazu dienen, Möglichkeiten und Grenzen von Simulationen zu besprechen. Da umfangreiche Hilfestellungen zur Verfügung gestellt werden, ist der Einstieg leicht.

EINSCHÄTZUNG: Das Programm ist eine Bereicherung für den Unterricht. Durch den spielerischen Zugang ist eine hohe Schülermotivation zu erwarten. Ökonomische Themen, für die im Unterricht meist keine Zeit ist, können anschaulich vermittelt werden, z.B. wirtschaftliche Zwänge. Geeignet ab der Mittelstufe. (Hinweis: Für OS X und LINUX muss der **► Quellcode** selbst noch kompiliert werden.)

GEOGEBRA (Mathematik)**Portable App:** Ja **Benutzeroberfläche deutsch:** Ja **Lizenz:** GPL, CC BY-NC

GEOGEBRA (GG) ist eine leistungsstarke Mathematiksoftware für die Schule mit dem Schwerpunkt Geometrie. GG erlaubt die Konstruktion und Darstellung geometrischer Figuren auf verschiedenen Wegen. Weiterhin enthält es einen Funktionenplotter, eine Tabellenkalkulation und ein leistungsfähiges **► CAS** (Computer-Algebra-System). Die deutsche Benutzeroberfläche ist gut strukturiert und meist intuitiv zu bedienen. Es gibt Versionen für die meisten Betriebssysteme (auch Handys und Tablets) und eine portable Windowsversion. Hinweis: Teile der Software sind nicht frei und müssen für den kommerziellen Einsatz (außerhalb der Schule) kostenpflichtig lizenziert werden.

EINSCHÄTZUNG: GeoGebra ist eine in Deutschland sehr weit verbreitete Mathematiksoftware für die Schule mit breiter Nutzerbasis und diversen Anwendungsmöglichkeiten. Es gibt viele ausgearbeitete Unterrichtsreihen und eine spezielle Version für Abitur-Prüfungen.

JMOL (Chemie)

Portable App: Nein **Benutzeroberfläche deutsch:** Ja **Lizenz:** LGPL

Komplexe Moleküle sind oft nur schwer vorstellbar und können mit den in Schulen vorhandenen Molekülbaukästen nicht in ausreichender Anzahl aufgebaut werden. Jmol lädt via Internet aus Moleküldatenbanken diese Strukturen herunter und stellt sie dar. Die Ansicht kann gedreht und gezoomt werden. Die angezeigten Bilder können in hoher Qualität exportiert und nachbearbeitet werden.

Es können jedoch keine eigenen Strukturen mit Jmol aufgebaut werden. Dies leistet das schwerer zu bedienende Programm Avogadro (<http://avogadro.cc>). Jmol benötigt eine installierte **► JAVA-Laufzeitumgebung (JRE)**.

EINSCHÄTZUNG: Jmol ist ein intuitives, kleines Programm, das nach kurzer Einarbeitung auch für Schüler*innen geeignet ist. Durch die einfache Bedienung kann das Programm innerhalb einer Unterrichtsstunde problemlos mehrfach von den Schüler*innen genutzt werden.

MARBLE (Geografie)

Portable App: Ja **Benutzeroberfläche deutsch:** Ja **Lizenz:** LGPL

Marble ist eine Software, die geografische und historische Karten auf einem virtuellen Globus darstellt. Neben der Darstellung von Offline-Karten, bei der im Wesentlichen die Daten des OpenStreetMap-Projektes und andere freie Quellen genutzt werden, können verschiedene Online-quellen Karten in Echtzeit **► rendern**. Damit lassen sich weltweit die Bewölkung und die Position von Raumschiffen bzw. Satelliten darstellen. Besondere Bedeutung haben historische Karten, die die zeitgenössischen Standardkarten ersetzen, und Darstellungen von Planeten. Mit interaktiven Funktionen kann Marble Routen als Fußgänger, Radfahrer und als Kraftfahrer berechnen.

Marble erweitert die Möglichkeiten eines konventionellen Globus um viele Aspekte: Die Interaktivität und verschiedenen Darstellungsarten können von Schüler*innen zur eigenen Erkundung von bestimmten Fragestellungen im Geografieunterricht genutzt werden.

EINSCHÄTZUNG: Marble erscheint als Universalwerkzeug für den Geografieunterricht. Mit der Nutzung von OpenStreetMap-Karten und weiteren Datenquellen sensibilisiert es zudem für das Thema **► Open Data**.

Weblink:

Projektseite:

<http://jmol.sourceforge.net>

Weiteren Informationen:

Jmol-Wiki:

http://wiki.jmol.org/index.php/Main_Page

Tutorial:

http://www.bioc.uzh.ch/plueckthun/nanowelt/Jmol/Tutorial1/JmolInteraktiv/01_Installation.html

Weblink:

Projektseite:

<http://marble.kde.org>

Weiteren Informationen:

Wikipedia:

[https://de.wikipedia.org/wiki/Marble_\(Computerprogramm\)](https://de.wikipedia.org/wiki/Marble_(Computerprogramm))

Weblink:

Projektseite:

➤ <https://musescore.org/de>**Weiteren Informationen:**

Handbuch (auch als PDF):

➤ <https://musescore.org/de/handbuch>

Video-Tutorial (20 Min.):

➤ <https://youtu.be/BX5ek8TxNpA>**Weblink:**

Projektseite:

➤ <https://scratch.mit.edu/scratch2download>**Weiteren Informationen:**

Materialien des Bildungsservers Berlin-Brandenburg:

➤ <http://bildungsserver.berlin-brandenburg.de/index.php?id=scratch>

Informationen und Tutorials

(M.I.T.):

➤ <https://scratch.mit.edu/about>

Die Programmiersprache ist im Werkzeugkasten »DIY und Making« beschrieben:

➤ <http://medien-in-die-schule.de/werkzeugkasten/werkzeugkasten-diy-und-making/maker-werkzeuge/scratch>

MUSESCORE (Musik)

Portable App: Ja **Benutzeroberfläche deutsch:** Ja **Lizenz:** GPL 2

MuseScore (MS) ist ein ► *WYSIWYG*-Notensatzprogramm, mit dem ansprechende Notenblätter erzeugt werden können. Mit MS können Kompositionen als Partitur zu Papier gebracht werden. Das Ergebnis kann abschließend ausgespielt werden. MS arbeitet mit vielen anderen Musiksoftwarewerkzeugen (auch kommerziellen) zusammen.

Das Programm ist multilingual und kann mit MIDI-fähigen Keyboards zusammenarbeiten. Andernfalls gibt man die Noten per Maus und Tastatur ein. Kompositionen können auch aus dem Web herunter- bzw. hochgeladen werden. Ein eingebauter Sequenzer spielt das Werk auf Wunsch ab. Für Windows existiert eine portable Version. Auf der Projektseite gibt es ein deutsches Handbuch und ein Video-Tutorial.

EINSCHÄTZUNG: MS ist ein sehr leistungsfähiges Spezialprogramm für Nutzer*innen, die bereits komponieren können und die nicht unerhebliche Einarbeitungszeit nicht scheuen. Für Schüler*innen ist es eher weniger geeignet.

SCRATCH 2.0 OFFLINE EDITOR (Informatik)

Portable App: Nein **Benutzeroberfläche deutsch:** Ja **Lizenz:** CC BY-SA 2.0

Der Scratch 2.0 Offline Editor ist die ► *grafische Programmiersprache* des ► *M.I.T.* speziell für jüngere Kinder und für alle Klassenstufen mit schnellen Erfolgserlebnissen für Anfänger*innen, der in dieser Version auch ohne Internetverbindung funktioniert. Die Nutzer*innen müssen Bildsymbole auswählen und in geeigneter Reihenfolge anordnen. Damit können z.B. Animationen schnell erstellt werden. Während die ältere Version die erstellten Programme stets im Web speicherte, werden sie bei dieser Version lokal gespeichert. Die in Deutschland häufig eingesetzte Programmiersprache stellt vergleichsweise hohe Ressourcenanforderungen an die Hardware und setzt ein installiertes Adobe ► *AIR* voraus.

EINSCHÄTZUNG: Die Einarbeitung ist weitgehend intuitiv und benötigt wenig Zeit. Es existiert eine Vielzahl an Beispielen und Beschreibungen, sodass der Einstieg auch jüngeren Schüler*innen sehr leicht fällt. Durch diverse Weiterentwicklungen von Scratch ist der Anschluss an andere Themen möglich, z.B. mit S4A (*Scratch for Arduino*) im Bereich des ► *Physical Computing*. Die zugrunde gelegten Konzepte bilden ein tragfähiges Gerüst für einen aktuellen Informatikunterricht.

Unterrichtsorganisation und Klassenmanagement

Der Unterricht lässt sich auch mithilfe von Freier Software verwalten, organisieren und managen.

GRADEMAN (für Lehrkräfte)

Portable App: Ja **Benutzeroberfläche deutsch:** Ja **Lizenz:** AGPL

Mit GradeMan (GM) verwaltet man komfortabel seinen Unterricht: Kurse, Schüler*innen, Stunden und Fehlzeiten werden lokal in einer Datenbank gespeichert und per Browser eingesehen. GM ist eine kleine, eher wenig bekannte Anwendung zur Unterrichtsverwaltung und ist nicht mit Funktionen überfrachtet. Sie läuft unter LINUX sowie WINDOWS, aber nicht unter OS X.

Das Programm startet ohne Installation. Die eingegebenen Daten werden in einer Datenbank verwaltet und können als ▶ CSV-Datei ex- und importiert werden. Auch ein Namen-Lernprogramm mit Schülerfotos ist eingebaut. Zu den Kursen werden die einzelnen Stundeninhalte gespeichert. Seit 2013 wurde das Programm nicht mehr weiterentwickelt. Eine Android-Version ist geplant.

EINSCHÄTZUNG: GradeMan ist zur einfachen Stundenverwaltung durch Lehrer*innen gut geeignet. Alles in Allem eine kleine Anwendung, die die wesentlichen Tätigkeiten der Unterrichtsverwaltung unterstützt. Hinweis: Bitte den Datenschutz beachten! Die Verarbeitung personenbezogener Daten ist in den jeweiligen Landesnormen verbindlich geregelt.

Weblink:

Projektseite:

➤ <http://polarwinkel.de/index.php?id=76>

Weiteren Informationen:

Video-Tutorial:

➤ https://youtu.be/y_RygSKRC4g

iTALC

Portable App: Nein **Benutzeroberfläche deutsch:** Ja **Lizenz:** GPL

iTALC ist ein vielseitiges Werkzeug zum Monitoring und zur Verwaltung von Arbeitsplatzrechnern in Computerräumen in Echtzeit. Es bietet die Möglichkeit, auf Bildschirminhalte einzelner Arbeitsplätze zuzugreifen oder die Anzeige eines Arbeitsplatzes auf andere Rechner zu spiegeln. Ferner können vom Hauptrechner Mitteilungen an die Arbeitsplätze gesandt, Rechner ferngesteuert oder auch Unterrichtsstunden aus einem in andere Räume übertragen werden.

EINSCHÄTZUNG: iTALC erweitert die Möglichkeiten eines Schulservers und bietet Möglichkeiten der Echtzeit-Administration während des Unterrichts.

Weblink:

Projektseite (englisch):

➤ <http://italc.sourceforge.net>

Weiteren Informationen:

Wikipedia:

➤ <https://de.wikipedia.org/wiki/ITALC>



WERKZEUGPORTRÄTS

Die Werkzeugporträts geben neben einer detaillierten Beschreibung der Software auch einen Einblick in mögliche Anwendungsszenarien. Kurzanleitungen helfen, sich mit der Software vertraut zu machen. Außerdem enthalten die Porträts eine pädagogische Einschätzung, die dazu beitragen kann, die Werkzeuge sinnvoll in der Unterrichtsvorbereitung und im Unterricht einzusetzen.

GIMP

Beschreibung

Das GNU Image Manipulation Program (GIMP) ist ein Bildbearbeitungsprogramm für Rasterbilder wie Fotos und Grafiken. Mit GIMP lassen sich sowohl einfache (z.B. Zuschneiden von Bildern, Größenänderungen für Web und Druck, Korrektur von blitzbedingten roten Augen) als auch mittelschwere Aufgaben (z.B. Weißabgleich, Helligkeit, Kontraständerung) lösen. Zudem können sehr komplexe Arbeiten (z.B. mit Ebenen und Masken) mit GIMP erledigt werden.

GIMP kann mit nahezu allen Formaten, in denen Pixelbilder und -grafiken abgespeichert sind, umgehen. Zahlreiche Import-Filter erlauben es den Nutzer*innen, Grafiken auch aus anderen Anwendungen zu bearbeiten. Wiederkehrende Bildbearbeitungsvorgänge lassen sich über Skripte automatisieren.

Einsatzmöglichkeiten in der Schule


Von der Arbeit mit und Manipulation von pixelbasierten Bilddateien können alle Fächer, vor allem die, die mit Bildern und Grafiken zu tun haben, profitieren. Für die Schule und schulische Zwecke sind die Möglichkeiten einfacher Pixelbildbearbeitung meist schon ausreichend. Aufgrund des hohen Funktionsumfangs lässt GIMP viele Anwendungsmöglichkeiten zu: Bei jüngeren Schüler*innen ist es vielleicht die Funktion »Zuschneiden«, mit deren Hilfe


übungshalber Personen aus einem Bild ausgeschnitten werden, bei älteren kann die Perspektive (Manipulation etc.) geändert werden.

Die komplexeren Funktionen von GIMP erfordern eine gewisse Einarbeitung und Übung, das Wissen um Ebenen und Masken ist sehr hilfreich. Aufgaben wie die Korrektur fallender Linien und Verzerrungen können einfach und professionell zugleich gelöst werden. Typische Digitalkameraprobleme (Weißabgleich) sind durch Filter schnell und unkompliziert lösbar.

Der frei zugängliche Quelltext der Software offeriert Interessierten die Möglichkeit, einen Blick in die Werkstatt einer mächtigen und ausgereiften Anwendung werfen zu können.

Erste Schritte

 **Beispiel 1: Weißabgleich** Bei der Durchführung eines Weißabgleichs, kann man sich beispielhaft mit der Software vertraut machen. Eine Kamera hat keine Information darüber, welche Bildpunkte als »weiß« anzusehen sind; sie legt dafür irgendeine Fläche fest. Diese ist aber nicht immer zutreffend. GIMP bietet einen automatischen Weißabgleich an, der sich hinter dem Reiter »Farben/Automatisch/Weißabgleich« verbirgt. Nach dem Abgleich können Ergebnisse direkt als komprimierte Datei im JPEG-Format (Reiter »Datei/Exportieren als...«) exportiert werden. Sollten die Ergebnisse von schlechterer Qualität als das Original sein, erlaubt die Tastenkombination »Strg+Z«, den letzten Arbeitsschritt rückgängig zu machen.

 **Beispiel 2: Zuschneiden** Ein digitales Bild ist selten in dem Verhältnis von Länge mal Breite, das der Aufnehmende im Sinne hatte. In der Dunkelkammer wurde das Negativ früher durch verschiebbare Schablonen soweit abgedeckt, das das Ergebnis stimmte. Diese Aufgabe übernimmt die Funktion »Zuschneiden«.

Im Werkzeugkasten befindet sich ein Symbol, das aussieht wie ein Papiermesser: Mit dessen Hilfe wird ein Rahmen um den aus zuschneidenden Teil gezogen. Ist der Rahmen stimmig, wird die Eingabetaste betätigt und das Ergebnis ist zu sehen. Mit »Datei/Exportieren als...« kann der Ausschnitt abgespeichert werden.

Hinweise

Über »Einstellungen« kann die standardmäßig dreiteilige Arbeitsoberfläche den eigenen Bedürfnissen angepasst werden, die Vielfalt kann zu Beginn ein wenig verwirren.

Es empfiehlt sich, die Zwischenschritte jeweils unter eigenen Dateinamen (mit Erweiterung -1 bzw. -2) abzuspeichern. So bleibt das Original erhalten.

Zudem ist es effizienter, am Ende Zwischenschritte zu löschen, als sie wiederholen zu müssen.

Die automatische Korrektur »Abgleichen« erlaubt (mitunter) die Verbesserung gleich mehrerer Funktionen; unter anderem Helligkeit/Kontrast, Farbton/Sättigung. Die Arbeit mit komplexeren Bildbearbeitungsfunktionen wie Ebenen und Masken bedarf deutlich mehr Aufmerksamkeit und Zeit.

Bildgestaltung und Manipulation

Hinter »Werkzeuge« verbirgt sich unter anderem der Reiter »Transformation«: Bilder gradweise zu drehen und Perspektivverzerrungen zu korrigieren sind bereits anspruchsvolle Aufgaben, die geübt werden müssen. Einige Digitalkameras stellen Pixelbilder her, die sehr perspektivverzerrte Ergebnisse abliefern. Darin liegt eine didaktische Chance: Ein um zwei Grad nach links gedrehtes Bild erweckt einen beschreibbar anderen Eindruck, ein z.B. in Sepia konvertiertes Bild wirkt nahezu antik, ein gezackter Rahmen unterstützt diese Impression. Mit Lernenden über solche Wirkungen zu diskutieren, schärft das Bewusstsein zum Thema Bildgestaltung, aber auch für die mögliche Manipulation von Bildern und ermöglicht damit eine medienkritische Reflexion.

Weiterführende Informationen

Projektseite¹³: [➤ https://gimp.org](https://gimp.org)

Portable App: [➤ http://portableapps.com/de/apps/graphics_pictures/gimp_portable](http://portableapps.com/de/apps/graphics_pictures/gimp_portable)

Handbuch: [➤ https://docs.gimp.org/2.8/de](https://docs.gimp.org/2.8/de)

Wikibook: [➤ https://de.wikibooks.org/wiki/GIMP](https://de.wikibooks.org/wiki/GIMP)

Tutorials (englisch): [➤ https://gimp.org/tutorials](https://gimp.org/tutorials)

Wikipedia: [➤ https://de.wikipedia.org/wiki/GIMP](https://de.wikipedia.org/wiki/GIMP)

¹³ WINDOWS- und MAC OS-Nutzer*innen bekommen das Programm unter der Internetadresse [➤ https://gimp.org](https://gimp.org), LINUX-Nutzer*innen über ihre distributionseigenen Softwarequellen.

INKSCAPE

Beschreibung

INKSCAPE (ein Kofferwort aus *ink*, »Tinte« und *-scape* wie in *landscape*, »Landschaft«) ist ein leistungsstarkes, plattformübergreifendes Programm zur Erstellung und Bearbeitung von zweidimensionalen **► Vektorgrafiken**.

Man unterscheidet grundsätzlich zwei Arten, Bilder und Grafiken zu speichern: Die Pixelgrafik, die für jedes einzelne Pixel einen zugehörigen Farbwert bestimmt, und die Vektorgrafik, die Vektoren und deren Farbänderung bestimmt. Um z.B. einen Kreis zu zeichnen, werden bei ihr lediglich zwei Werte (Mittelpunkt und Kreisdurchmesser) benötigt. Dazu kommen gegebenenfalls noch Farbe, Strichstärke und Füllmuster.

INKSCAPE ist sehr übersichtlich aufgebaut, alle Icons geben beim Überfahren Informationen darüber, was sich dahinter verbirgt. Klickt man auf ein Icon auf der linken Seite, werden jene Möglichkeiten angezeigt, die mit ihm verbunden sind. Die unter dem Arbeitsschirm abgebildete Farbleiste ermöglicht die schnelle Zuordnung von Farben zu Formen. Die Statusleiste informiert über alles, was im Moment wichtig sein könnte (unter anderem Standort Cursor, Farbwert einer Stelle, über die man gerade mit der Maus fährt). Auf der rechten Seite des Bildschirms findet sich die Hauptleiste mit Zugriff auf alle Funktionen. Eingezeichnete Objekte lassen sich über variable Anfasserpunkte nach Belieben verformen, verschieben, drehen, in der Größe ändern, klonen, vereinigen, kombinieren, zerlegen, schrumpfen und vieles mehr.

Der Import und Export vieler Formate in Inkscape erleichtert die Zusammenarbeit mit anderen Programmen. Der Export von SVG-Grafiken in z.B. SCRIBUS wertet dessen Produkte deutlich auf. Das Abspeichern des Produkts in PDF erlaubt die Fixierung des Ergebnisses in einem definierten Zustand. Selbst die Vektorisierung von Bitmaps ist möglich.


Ist die grobe Arbeit mit der Maus auf der Arbeitsfläche erledigt, bietet INKSCAPE millimetergenaue Positionierungen.

Einsatzmöglichkeiten in der Schule

Die Arbeit mit INKSCAPE erfordert eine längere, aber notwendige Einarbeitungszeit. Sie empfiehlt sich bei vektorbasierten Projekten (Poster, Logos, Diagramme, Landkarten, Plakate in Großformaten) oder PDF-Dateien. INKSCAPE liefert mit geringem Einarbeitungsaufwand akzeptable Ergebnisse, vertieftes Einsteigen in die Anwendung erfordert viel Zeit, dafür erzielt man hervorragende Resultate. Für die Arbeit mit Pixelgrafiken nutzt man besser GIMP.

In der Schule sind Vektorgrafiken vor allem für Hintergründe, Logos, Bilder mit Texten oder Poster und Plakate (z.B. zur Ankündigung eines Schulfestes) einsetzbar. Das Fach Erdkunde verwendet in der Regel vektorbasierte Landkarten. Auch die Entnahme von Informationen aus einer PDF und die weitere Verwendung kann mit INKSCAPE erledigt werden. Zielgruppe sind Lehrkräfte und ältere Schüler*innen. Für die Erstellung von z.B. Schulwebseiten oder Logos gehört die Software in die Hand der Oberstufe (gestaltende Fächer).

Erste Schritte

 **Beispiel 1:** **Import, Umwandlung und Export einer Pixelgrafik (im Format PNG)**

Mit Inkscape können Pixelgrafiken importiert, in eine skalierbare Vektorgrafik umgewandelt und als deutlich größere Pixelgrafik exportiert werden. Das ist eine der einfacheren Anwendungen für Inkscape:

Die zu kleine, als Datei vorliegende Pixelgrafik wird in Inkscape importiert (»Datei/Importieren«). Beim Import die Optionen »Einbetten« und »Glätten« wählen. Das Originalbild erscheint auf der Arbeitsfläche. Um das Bild befinden sich acht Anfasser, die eine stufenlose Vergrößerung (beim zweiten Klick auch Drehen, Stauchen und anderes mehr) erlauben. Ist die gewünschte Zielgröße erreicht, kann das Bild wieder in eine Pixelgrafik exportiert werden. Soll die Grafik skaliert verändert werden (das Verhältnis von Länge mal Breite bleibt gleich), dann wird die Grafik markiert, unter »Objekt/Transformation« der Reiter »Maßstab« gewählt und das Feld »Proportional skalieren« angehakt. Ist die ursprüngliche Pixelgrafik bereits von geringer Qualität, wird das Ergebnis mitunter zu wünschen übrig lassen. Bilder und Grafiken können mit Inkscape auch in eine Vektorgrafik umgewandelt werden. Das erfordert weitere Kenntnisse. Rechts von der Arbeitsfläche sind alle Möglichkeiten (z.B. Einheiten, Bildgröße und Speicherort für die Exportdatei) aufgeführt, die zu dieser Aktion gehören können.

 **Beispiel 2:** **Import aus der »Open Clipart Library«**

Hat der Rechner Verbindung ins Internet, können (über »Datei/Clipart importieren«) Cliparts direkt aus der OPEN CLIPART LIBRARY (deren Lizenzen geklärt sind und die für Bildungszwecke unbeschränkt genutzt werden können) importiert, bearbeitet und verwendet werden. Da sie in der Regel als Vektorgrafik (SVG) vorliegen, können sie in jeder Textverarbeitung oder auch in einem ► *DTP-Programm* problemlos weiterverwendet werden.

 **Beispiel 3:** **Bearbeiten einer PDF-Datei**

Alle aktuellen Versionen von INKSCAPE können Elemente aus ► *PDF-Dateien* bearbeiten. Dazu wird die PDF geöffnet (»Datei/Öffnen«). Bevor die einzelnen Elemente bearbeitet werden können, muss im Dokument die Gruppierung gelöst werden (»Objekt/Entsperren«). Bilder und Grafiken können dann bearbeitet werden.

Für die Bearbeitung von Text aus einer PDF sind zwei Voraussetzungen erforderlich. Erstens: Die im Original verwendeten Schrifttypen müssen zur Verfügung stehen. Zweitens: Auch das Textobjekt muss gesondert markiert (Textsymbol) und entsperrt werden (»Text/Manuelle Unterscheidungen entfernen«). Erst danach kann das Textobjekt bearbeitet werden. Inkscape kann lediglich eine Seite bearbeiten. Für die Bearbeitung mehrerer Seiten müsste die PDF in Einzelseiten aufgetrennt werden.

14 Windows- und Mac OS-Nutzer*innen bekommen das Programm unter der Internetadresse <https://inkscape.org/de>, LINUX-Nutzer*innen über ihre distributions-eigenen Softwarequellen.

Weiterführende Informationen

Projektseite¹⁴: <https://inkscape.org/de>

Portable App: https://inkscape.org/en/gallery/item/4460/InkscapePortable_0.91.paf.exe

Handbuch: <https://inkscape.org/de/lernen>

Wikibook: <https://de.wikibooks.org/wiki/Inkscape>

Informationen (unter anderem Erläuterung der Symbole):

<https://wiki.ubuntuusers.de/Inkscape>

Kurs: Das Freie Magazin (<http://freiesmagazin.de>) enthält in den Nummern 4/2016 und 5/2016 einen sehr übersichtlichen und praxisorientierten zweiteiligen Kurs zu Inkscape.

Video-Tutorials (englisch): <https://inkscape.org/de/lernen/videos>

Wikipedia: <https://de.wikipedia.org/wiki/Inkscape>

SCRIBUS

Beschreibung

Für die Erstellung von Arbeitsblättern werden oftmals Textverarbeitungen wie LIBREOFFICE WRITER oder MICROSOFT WORD genutzt. Diese sind jedoch eher für das Schreiben von Texten als für die Gestaltung von kompletten Seiten geeignet. Für letzteres gibt es Desktop-Publishing-Programme (► *DTP-Programme*) wie SCRIBUS. Das Seitenlayout-Programm eignet sich zur Gestaltung von z.B. Glückwunschkarten, Plakaten, Broschüren und Arbeitsblättern.

Im Unterschied zur seitenorientierten Textverarbeitung arbeitet Scribus mit frei verschiebbaren Objekten (die beiden wichtigsten sind Textobjekt und Bildobjekt) und eignet sich dadurch für die Gestaltung von Einzelseiten. SCRIBUS exportiert in Raster- und Vektorgrafik, EPS und PDF. Es hat Stärken in der Druckvorlagenherstellung.

SCRIBUS erfordert eine gewisse Einarbeitungszeit, liefert aber sehr gute Resultate. Ist die Einarbeitung abgeschlossen und sind bereits einige Ergebnisse vorhanden sowie in die Scribus-Bibliothek eingefügt, kann auf Elemente daraus immer wieder zugegriffen werden. Im Bildungsalltag vermutlich eher selten verwendete Spezialitäten (ausgefeilte Umbruchfunktionen wie die Korrektur von typografischen Satzfehlern) fehlen, Silbentrennung gibt es nur für das komplette Textobjekt.

Die Regeln guter typografischer Gestaltung (nur eine Schriftfamilie und ein Schriftschnitt je Dokument) sollten im Sinne guter Wahrnehmung dringend beachtet werden.


Ruft man SCRIBUS auf, erscheint zunächst eine Startseite, auf der viele Vorgaben für ein Projekt eingestellt werden: Bemaßung, Ränder, Format des geplanten Blattes (z.B. A4). Hier werden alle wesentlichen Entscheidungen für

Umfang und Design des Projekts getroffen. An dieser Stelle ist es von großem Vorteil, eine Idee davon zu haben, wie das fertige Produkt (Flyer, Karte, A4-Seite, Text in Spezialformat, etwa in einer Ellipse und vieles andere mehr) aussehen könnte.

Einsatzmöglichkeiten in der Schule

SCRIBUS ist für schulische Zwecke optimal aufgestellt: Klein, überschaubar und einfach zu nutzen. Geeignet ist es vor allem für Lehrende, aber auch ältere Schüler*innen. Anders als bei einer reinen Textverarbeitung werden die Nutzer*innen angeregt, über die Beziehung von Form und Inhalt nachzudenken, indem sie Objekte positionieren müssen und die Gestaltung reflektieren können.


Erste Schritte


 **Beispiel 1: Gestaltung eines Textobjekts** Nach der Grundeinrichtung der geplanten Seite (z.B. zweispaltig, A4, Hochformat) öffnet sich ein frei gestaltbares Blatt mit wenigen Symbolen oberhalb. Hinter »Ansicht« verbergen sich weitere und differenziertere Einstellungsmöglichkeiten wie Millimeterpapier (für die exakte Positionierung) oder magnetische Hilfslinien, die die Festlegung des Objekts auf der Arbeitsfläche enorm erleichtern.


Ein Klick auf das Textrahmensymbol mit dem kleinen Buchstaben a bietet die Möglichkeit, einen Textrahmen zu öffnen. Er wird rot gekennzeichnet und hat acht erkennbare Punkte, an denen die Rahmengröße auch nachträglich (selbst mit Text im Rahmen) geändert werden kann.

Ein Rechtsklick in dieses Fenster zeigt alle Aktionsmöglichkeiten an, die derzeit mit diesem Fenster durchgeführt werden können. Zumeist handelt es sich um die Funktion »Text bearbeiten« (auch mit der Tastenkombination »Strg+G« aufrufbar). Diese Aktion öffnet einen sogenannten »Story-Editor«.

Alle möglichen Aktionen für die Arbeit im Textrahmen (z.B. Schrifttyp, Schriftgröße, Unterschneidung, Buchstabenabstand, Schriftfarbe oder Bündigkeit) werden hier erledigt. Mit der Taste F2 sind weitere Einstellungsmöglichkeiten erreichbar. Wie in den meisten Anwendungen lassen sich mit der Tastenkombination »Strg+Z« Arbeitsschritte rückgängig machen. Texte aus anderen Anwendungen (etwa LIBREOFFICE oder einem einfachen Texteditor) können hier hineinkopiert werden. Soll ein fertiger Text weiterbearbeitet werden, muss er zuvor markiert werden.

 **Beispiel 2: Textobjekte verketteten** Soll an verschiedenen Positionen auf dem Blatt Text fortlaufend dargestellt werden, so ist die Funktion »Textobjekte verketteten« hilfreich. Sie sorgt dafür, dass überfließender Text im nächsten Rahmen erscheint. Der Textrahmen mit der Textüberfluss wird angeklickt, ein zweiter Rahmen wird aufgezogen. Die Symbolleiste oberhalb des Blattes enthält das Symbol »Textrahmen verketteten«: Der überschüssige Text läuft in den nächsten Rahmen. Das kann beliebig oft wiederholt werden.

 **Beispiel 3: Bildobjekte gestalten** Nach einem Klick auf das Bildsymbol kann ein Rahmen aufgezogen werden. Mit der rechten Maustaste wird ein »Bild geladen«. Zumeist wird das Bild größer oder kleiner als der Rahmen sein. Dann kann entweder der Rahmen oder das Bild angepasst werden. Oft erweist sich die Funktion »Bild an Rahmen anpassen« als adäquat. Mit F2 können Bilder in der Darstellungsgröße (auch skaliert) verändert, werden, sodass das Verhältnis von Länge zu Breite gleich bleibt. In bescheidenem Umfang können Bilder manipuliert werden. Allerdings empfiehlt es sich, die Bearbeitung des Bildes abgeschlossen zu haben (z.B. Pixelbilder mit GIMP; Vektorgrafiken mit Inkscape). Alle verbreiteten Bildformate (Pixel- wie Vektorgrafiken) sowie Cliparts lassen sich einfügen.

 **Beispiel 4: Bildobjekte mit Textobjekten verknüpfen** Sind die Textobjekte erarbeitet und die Bildobjekte angepasst, dann wird für die Gestaltung der Seite interessant, wie z.B. der Text um ein Bild oder eine Grafik herumfließt.

Folgende Schritte sind dazu notwendig: Zwei Textrahmen sind miteinander verkettet. Nun wird z.B. in der Mitte der Spalten über das Bildsymbol ein Bild platziert. Anschließend ruft man mit F2 den Eigenschaftsinspektor auf und wählt »Form«. Nun hat man auch die Möglichkeit, den »Rahmenumriss« zu benutzen: Der Text fließt um das Bild herum.

Ebenso kann der Abstand vom Bild zum Text eingestellt und dem Text eine andere Außenform als das übliche Rechteck (etwa ein Herz) gegeben werden. Mithilfe des Eigenschaftsinspektors können an den Objekten zahlreiche weitere Einstellungen vorgenommen werden – Scribus verfügt über eine große Spanne an Möglichkeiten.

Fertig gestaltete Seiten lassen sich als ► **PDF** exportieren. Das SCRIBUS-Dokumentformat »SLA« basiert auf ► **XML**, sodass eine Transformation in andere Formate ebenfalls möglich ist.

Weiterführende Informationen

Projektseite¹⁵: [➤ https://scribus.net](https://scribus.net)

Portable App: [➤ http://portableapps.com/apps/office/scribus_portable](http://portableapps.com/apps/office/scribus_portable)

Templates: [➤ http://scribusstuff.org/?xcontentmode=642](http://scribusstuff.org/?xcontentmode=642)

Dokumentation: [➤ https://wiki.scribus.net/canvas/Hauptseite](https://wiki.scribus.net/canvas/Hauptseite)

Tutorial: [➤ https://wiki.scribus.net/canvas/Einstieg_in_Scribus](https://wiki.scribus.net/canvas/Einstieg_in_Scribus)

How-to: [➤ https://wiki.scribus.net/canvas/Category:HOWTO_DE](https://wiki.scribus.net/canvas/Category:HOWTO_DE)

Video-Tutorials: [➤ https://youtube.com/playlist?list=PL103A931AE65DF703](https://youtube.com/playlist?list=PL103A931AE65DF703)

Vergleich Textverarbeitung mit DTP-Seitenlayout (englisch):

[➤ https://wiki.scribus.net/canvas/Word_Processing_vs_DTP](https://wiki.scribus.net/canvas/Word_Processing_vs_DTP)

Beispiel-Projekt: [➤ https://www.gemeindebriefdruckerei.de/fileadmin/o1_redaktion/Downloads/Anleitung_Kunden/Satzhilfe-Scribus_1.4.2.pdf](https://www.gemeindebriefdruckerei.de/fileadmin/o1_redaktion/Downloads/Anleitung_Kunden/Satzhilfe-Scribus_1.4.2.pdf)

Wikipedia: [➤ https://de.wikipedia.org/wiki/Scribus](https://de.wikipedia.org/wiki/Scribus)

15 Windows- und Mac OS-Nutzer*innen bekommen das Programm unter die Internetadresse [➤ https://scribus.net](https://scribus.net), Linux-Nutzer*innen über ihre distributionseigenen Softwarequellen.

FIREFOX

Beschreibung

FIREFOX ist der von der MOZILLA-STIFTUNG initiierte Webbrowser, der für alle Betriebssysteme verfügbar ist. Er bietet die klassische Funktionalität, Webseiten aufzurufen unter besonderer Berücksichtigung der Privatsphäre und der Sicherheit persönlicher Daten. Eine als »Sync« bezeichnete Funktion bietet die verschlüsselte Speicherung von Webseitenpasswörtern und deren Synchronisation auf unterschiedlichen Computern und Smartphones.


FIREFOX bietet mit Themes, Plugins und Erweiterungen Möglichkeiten, ihn auf persönliche Bedürfnisse anzupassen. Mit Themes kann das Aussehen des Programms verändert werden. Das Angebot reicht von einfachen Farbverläufen bis hin zu Bildhintergründen. Plugins stehen hauptsächlich im Zusammenhang mit multimedialen Inhalten von Webseiten. So ermöglichen Plugins für verschiedene Audio- und Videoformate die Anzeige von Audio- und Videodateien direkt im Browser, ohne dass auf dem Computer oder dem Smartphone ein weiterer Multimediaplayer installiert werden muss. Mit Erweiterungen (Add-ons) werden zusätzliche Funktionen ergänzt. Viele dieser Erweiterungen stehen im Zusammenhang mit dem Themenfeld Internet: Sie erhöhen Sicherheit, Datenschutz oder ermöglichen die Analyse und Bearbeitung von Webseiten. Installierbare Themes, Plugins und Erweiterungen sind teilweise nicht frei. Außerdem sind sie unter Umständen unsicher und verlangsamen das Programm.


Als besondere Version gibt es den »Tor-Browser«, der anonymes Browsen über das [► Tor-Netzwerk](#) ermöglicht.

Einsatzmöglichkeiten in der Schule

FIREFOX ist für jede Arbeit mit Webseiten bzw. ► *HTML*-Dokumenten geeignet. Über diesen allgemeinen Aspekt der Nutzung hinaus ist er besonders im Informatikunterricht für die Erstellung und Analyse von Webseiten zu nutzen. Eine Vielzahl von Werkzeugen zu diesem Zweck befindet sich im Menü unter »Extras« im Unterpunkt »Entwicklerwerkzeuge« bzw. im rechten Kontrollmenü unter »Entwicklerwerkzeuge«. Da sich die einzelnen Werkzeuge zum einen in starker Entwicklung befinden und zum anderen für sehr spezielle Aufgaben im Unterricht eignen, empfiehlt es sich, die Dokumentation bzw. die Hilfeseiten zum FIREFOX zu nutzen.

Erste Schritte

 **Beispiel 1: Datenschutz-einstellungen** Da beim Surfen grundsätzlich Daten auf dem lokalen Rechner gespeichert und die Arbeitsplatzrechner häufig von mehreren Schüler*innen genutzt werden, sind sinnvolle erste Schritte, die Datenschutzeinstellungen zu verändern. Andernfalls werden z.B. Login-Daten automatisch gespeichert und ermöglichen den Zugriff auf fremde Konten. Je nach Version und Grundeinstellung des FIREFOX befinden sich die entsprechenden Einstellmöglichkeiten an unterschiedlichen Orten: Entweder über den Menüpunkt »Bearbeiten« oder das Kontrollmenü auf der rechten Seite sind die »Einstellungen« und der Reiter »Datenschutz« aufzurufen. Dort ist es sinnvoll, ein Häkchen bei »Schutz vor Aktivitätenverfolgung in privaten Fenstern verwenden« zu setzen. Dies ermöglicht den Aufruf von Internetseiten, ohne dass Informationen der Browsersitzung auf dem lokalen Rechner gespeichert werden. Ausnahmen sind Downloads. Des Weiteren sollte auch die »Chronik« angepasst und auf die Einstellung »Nach benutzerdefinierten Einstellungen anlegen« gesetzt werden. Um grundsätzlich ohne Speichern von lokalen Dateien zu surfen, kann der »private Modus« dauerhaft aktiviert werden. Für eine etwas feinere Abstufung bietet es sich an, ► *Cookies* zwar zu akzeptieren, aber nicht von Drittanbietern, die dort in der Regel Analysezwecken dienen. Mit der Aktivierung des Menüpunkts »Cookies behalten, bis der Browser geschlossen wird« werden alle Login-Daten nach Schließen des Programmfensters gelöscht.

 **Beispiel 2: Sicherheit** Eine weitere wichtige Grundeinstellung ist unter »Einstellungen« und weiter dem Reiter »Sicherheit« vorzunehmen. Dort sollten die angebotenen Möglichkeiten zur Verhinderung der automatischen Add-on-Installation und die Webseitenblockierung bei möglichen Angriffen und bei Betrugsversuchen ausgewählt werden. Zugangsdaten sollten wegen der in der Regel zahlreichen Benutzer*innen nicht gespeichert und die entsprechende Funktion deaktiviert werden.

Hinweis

Eine Entscheidung muss auch hinsichtlich der an die MOZILLA-STIFTUNG übermittelten Daten getroffen werden: Bei der Erstinstallation wird abgefragt, ob Leistungsdaten, die Nutzung bestimmter Funktionen und ein Absturzmelder bestimmte Daten an die Stiftung übermitteln. Zum einen wird damit die Möglichkeit der Analyse und der Anpassung der Software gegeben, zum anderen können diese Metadaten das Nutzungs- und Surfverhalten individuell protokollieren. Die gespeicherten Daten unterliegen amerikanischen Datenschutzgesetzen.

Weiterführende Informationen

Projektseite: ➤ <https://mozilla.org/de/firefox>

Portable App: ➤ http://portableapps.com/de/apps/internet/firefox_portable

Erweiterungen: ➤ <https://addons.mozilla.org/de/firefox>

Tor-Browser: ➤ <https://torproject.org/projects/torbrowser.html>

Wikipedia: ➤ https://de.wikipedia.org/wiki/Mozilla_Firefox

LIBREOFFICE WRITER

Beschreibung

LIBREOFFICE WRITER bietet umfangreiche Möglichkeiten zur Erstellung und Bearbeitung von Dokumenten, in denen hauptsächlich Texte gestaltet und strukturiert werden müssen: Berichte, Dokumentationen, Lebensläufe und Bewerbungsschreiben. Neben der allgemein üblichen Arbeit mit Formatierungen kann mit Formatvorlagen gearbeitet werden, mit denen Formatierungen global bearbeitet und zugewiesen werden. Für umfangreichere Arbeiten gibt es automatisierte Inhaltsverzeichnisse und die komfortable Einbindung von Quellenangaben. In die Dokumente können Bilder eingebunden und mit grundlegenden Funktionen wie Helligkeit, Kontrast usw. manipuliert werden.

LIBREOFFICE befindet sich in intensiver Entwicklung, die durch die THE DOCUMENT FOUNDATION koordiniert wird. Deswegen existieren insgesamt drei Hauptprogrammversionen (mit Stand Mitte 2016: 3.x, 4.x, 5.x). Die Entwicklungszweige stehen jeweils für einen deutlich umfangreicheren Funktionsumfang von WRITER bzw. des LIBREOFFICE-Pakets. Frühe Programmversionen (wie z.B. 5.01) erhalten in kurzer Folge Updates vor allem zur Fehlerbehebung. Alle drei großen Entwicklungszweige sind gut im Unterricht nutzbar. Die zusätzlichen Funktionen der Versionen 4.x und 5.x sind im Hinblick auf die Vermittlung von Anwendungskompetenzen nicht erforderlich. Mit Stand Mitte 2016 kann eine Empfehlung für die Version 4 gegeben werden, die stabil funktioniert und unter dem Entwicklungsaspekt als ausgereift gilt.

Einsatzmöglichkeiten in der Schule

Die Software eignet sich für alle schulischen Bereiche und Fächer. Für den Austausch von Dokumenten, die nicht mehr bearbeitet werden müssen, ist eine Exportfunktion in das ► *PDF*-Format integriert. Damit können auch elektronisch ausfüllbare Formulare auf der Schulhomepage hinterlegt werden.

Hervorzuheben ist, dass es für Writer redaktionell betreute Unterrichtsmaterialien gibt, die z.B. für die Vorbereitung der Prüfung des Europäischen Computerführerscheins ► *ECDL* genutzt werden können.

Ein wesentlicher Schritt zur Nutzung von Writer im Unterricht ist die Wahl und Entscheidung für eine Programmversion. Damit ist gewährleistet, dass sich für alle die Menüpunkte am selben Ort befinden und es keine Abweichungen gibt. Das erleichtert die Vermittlung der einzelnen Funktionen. Zu beachten ist, dass sich auch die Hilfeseiten auf bestimmte Programmversionen zu beziehen und neuere in der Regel weniger detailliert dokumentiert sind.

Erste Schritte

 **Beispiel 1:**
Seite
formatieren

Für Dokumente gelten unterschiedliche Formatierungsvorschriften bzw. -empfehlungen. Im Folgenden soll eine Seite, wie sie z.B. in einem Aufsatz, einer Beschreibung oder einer schriftlichen Dokumentation verwendet wird, mit definierten Rändern, einer Kopfzeile mit dem Titel der Arbeit und dem Namen sowie einer Seitenzahl versehen und eine Schriftart festgelegt werden.

Die Formatierungsfunktionen sind im Menü »Format« zu finden. Dort wählt man zunächst »Seite« aus. Voreingestellt ist das gängige Papierformat DIN A4. Unter dem Menüpunkt »Seitenränder« werden die entsprechenden Werte eingestellt.

Vor allem bei mehrseitigen Dokumenten empfiehlt sich das Einfügen des Schülernamens, da beim Ausdruck die Seiten sonst nur schwer zuzuordnen sind. Dazu wählt man im Menü den Punkt »Einfügen« und »Kopfzeile« aus. Die Auswahl »Alle« ist passend, solange man keine separate Titelseite einfügt. Nach dem Auswählen kann man direkt in die Kopfzeile schreiben – die Informationen werden auf allen Seiten angezeigt. Möchte man später diese Eingaben ändern, klickt man in den Kopfzeilenbereich einer Seite über dem Stammtext.

Zum Einfügen einer Seitenzahl wiederholt man den Vorgang mit »Einfügen« einer »Fußzeile«. Befindet sich der Cursor im Fußzeilenbereich, klickt man erneut den Menüpunkt »Einfügen« und dann »Feldbefehl«: Aus den verschiedenen Optionen wählt man nun »Seitennummer«. Sollen Kopf- oder Fußzeile formatiert werden, geschieht das wiederum über den Menüpunkt »Format« und »Seite« und die Wahl der Registerkarte »Kopfzeile« bzw. »Fußzeile«.

Alle Eigenschaften der Schrift werden über den Menüpunkt »Format« und »Zeichen« festgelegt.

Die genannten Einstellungen beeinflussen das aktuell verwendete Dokument. Sollen bestimmte Formatfunktionen voreingestellt und bei jedem Dokument zur Verfügung stehen, wählt man unter »Format« die »Formatvorlagen« (auch mit der Funktionstaste F11) und ändert dort mit einem Rechtsklick die Einstellungen der Formatvorlage »Standard«.

 **Beispiel 2:**
Speichern
als DOC

Bei der Verwendung von LIBREOFFICE kommt es in der Schule nicht selten zu Kompatibilitätsproblemen, die jedoch einfach zu beheben sind. Sie beruhen auf der Verwendung von nicht standardisierten bzw. dokumentierten Dokumentformaten. Um Komplikationen beim Dokumentenaustausch zu minimieren, empfiehlt sich die Verwendung des älteren ► *DOC*-Formates, das mit MICROSOFT OFFICE populär wurde. Dazu wählt man nach Erstellen eines Dokumentes den Menüpunkt »Datei« und »Speichern unter«. Statt der Voreinstellung »Alle Formate« entscheidet man sich für die Option »Microsoft Office 97–2003« und die zugehörige Dateiendung »DOC«. Im schulpraktischen Alltag sind damit nur geringste Einschränkungen verbunden, die einzelne Formatierungseinstellungen betreffen, aber deutlich von den Vorteilen überwogen werden: Auch Nutzer*innen älterer MICROSOFT-Produkte können dieses Format problemlos lesen und bearbeiten.

 **Beispiel 3:**
Export als
PDF

Sollen Dokumente (z.B. schulinterne Mitteilungen im Mailverkehr) zunächst nicht weiter bearbeitbar sein, empfiehlt sich das PDF-Format, das auch von mobilen Geräten wie Smartphones oder Tablets sehr gut unterstützt wird.

Für den PDF-Export wird im erstellten Dokument unter dem Menüpunkt »Datei« »Exportieren als PDF« ausgewählt, eine Funktion, die in allen Programmen des LIBREOFFICE-Paketes zur Verfügung steht. Die Voreinstellungen sind in der Regel ausreichend. Die häufiger benutzte Funktion, nur bestimmte Seiten zu exportieren, findet sich unter »Bereich«.

PDF-Dokumente werden häufig auf Schulhomepages als Download zur Verfügung gestellt. Standardmäßig speichert Writer die Benutzerdaten im PDF-Dokument. Um diese Funktion zu deaktivieren, wählt man unter dem Menüpunkt »Datei« die »Eigenschaften« und entfernt das Häkchen bei »Benutzerdaten verwenden« auf der Registerkarte »Allgemein«.

Weiterführende Informationen

Projektseite: ➤ <https://de.libreoffice.org>

Portable App: ➤ http://portableapps.com/de/apps/office/libreoffice_portable

Handbuch: ➤ <https://de.libreoffice.org/get-help/documentation>

Wikipedia: ➤ <https://de.wikipedia.org/wiki/LibreOffice>

AUDACITY

Beschreibung

AUDACITY ist ein Programm für die Aufnahme und Bearbeitung von Klanginformationen. Es bietet viele Möglichkeiten und setzt der Kreativität im Umgang mit Audiodateien kaum Grenzen: ob beim Mischen verschiedener Musikstücke, Herausfiltern von Störungen, Verändern der Abspielgeschwindigkeit oder Hinzufügen von Klangeffekten. AUDACITY beherrscht alle gängigen Audiiformate, darunter MP3, Ogg Vorbis, WAC, MIDI und AIFF und erlaubt das Konvertieren von Audiodateien zwischen diesen Formaten. Neben der Tonbearbeitung kann man AUDACITY auch zur Aufnahme verwenden und die so erzeugten Dateien direkt weiter verarbeiten und auf der Festplatte abspeichern.

Einsatzmöglichkeiten in der Schule

AUDACITY kann in vielen Bereichen und Fächern im Unterricht eingesetzt werden. Es fördert die Kreativität der Schüler*innen und ermöglicht es, theoretisches Wissen mit anderen Sinnen wahrzunehmen und so besser zu verstehen. Im Folgenden sollen ein paar mögliche Einsatzgebiete beschrieben werden: Durch das Einbinden von Tonaufnahmen kann eine computergestützte Präsentation bzw. ein Referat in verschiedenen Fächern schnell abwechslungsreicher und unterhaltsamer gestaltet werden. In der Regel können Tonaufnahmen aber nicht eins-zu-eins in eine Präsentation übernommen werden: AUDACITY hilft beim Schneiden und Kombinieren von Audiodateien sowie beim Anpassen von Lautstärke und Abspieldauer.

AUDACITY kann dazu verwendet werden, Interviews (im Rahmen einer praktischen Hausaufgabe, einer AG, der Schülerzeitung oder des Schulradios) mit einem Computer aufzunehmen, zu schneiden und mit anderen Aufnahmen zu kombinieren. Hintergrundgeräusche lassen sich mit den passenden Rauschfiltern entfernen.

Im Deutschunterricht kann man AUDACITY nutzen, um Aufnahmen von Theateraufführungen zu bearbeiten und z.B. gezielt bestimmte Teile des Stücks als Rezitation einzuüben; darüber hinaus können eigene Theaterstücke, Hörbücher oder Geschichten aufgenommen und mit Musik und Klangeffekten unterlegt werden.

Im Fremdsprachenunterricht oder in der Arbeit mit Schüler*innen mit Migrationshintergrund können sehr einfach kontrollierte Sprachübungen durchgeführt werden: Um die eigene Aussprache mit dem Original zu vergleichen verwendet man die Aufnahmefunktion.

AUDACITY besitzt einen Tongenerator – ein interessantes Werkzeug für den Physikunterricht. Mit ihm lassen sich Töne unterschiedlicher Wellenformen, Frequenzen, Amplituden und Dauer generieren. Dadurch können Schüler*innen

nicht nur auf theoretischer Ebene lernen und verstehen, wie sich Änderungen an den Parametern auswirken, sondern diese auch akustisch wahrnehmen.

Im Musikunterricht bietet AUDACITY viele Möglichkeiten der Bearbeitung von Aufnahmen: Schüler*innen verstehen auf kreative Art, wie Musik entsteht. Gezielte Manipulationen vermitteln ein Gefühl davon, wie sich der Charakter eines Musikstücks ändern kann. Auch zum Einüben eigener Musikstücke (z.B. für ein Schulfest) kann die Aufnahmefunktion von AUDACITY sehr hilfreich sein.

Erste Schritte

Beispiel: **Rausch-** **entfernung**

Wer schon mal einen Kassettenrecorder oder ein Diktiergerät bedient hat, dem werden viele Bedienelemente von Audacity schnell bekannt vorkommen. Die meisten Laptops verfügen heutzutage über ein eingebautes Mikrofon. Ist das nicht der Fall, kann ein externes Mikrofon oder ein Headset angeschlossen werden. Die Aufnahme wird mit einem Klick auf den Knopf mit dem roten Aufnahmezeichen gestartet. Anschließend kann die aufgenommene Tonspur über die Wiedergabetaste angehört werden, wobei sich ein unangenehmes Hintergrundrauschen feststellen lässt.

Um dieses zu entfernen wird eine zweite Aufnahme für ein paar Sekunden bei absoluter Stille gemacht. Sie dient dazu, ein Profil des vom Mikrofon erzeugten Rauschen zu erstellen. Mit der Maus kann die Aufnahme des Rauschens markiert werden. Danach wird die Funktion »Rauschentfernung« im Menü »Effekte« ausgewählt. Es öffnet sich ein Dialog, in dem die Operation »Rauschprofil ermitteln« ausgeführt wird. Jetzt kann die zweite Tonspur, bei der nur das Rauschen aufgenommen wurde, über das »x« neben »Tonspur« wieder gelöscht werden. Im nächsten Schritt wird die eigentliche Aufnahme markiert und die Funktion »Rauschentfernung« im Menü »Effekte« aufgerufen. Nun wählt man unten im Abschnitt »Rauschen« die Aktion »Entfernen« aus und klickt auf OK. Damit sollte das Hintergrundrauschen entfernt sein. Zur Überprüfung kann die Aufnahme über die Wiedergabetaste erneut abgespielt werden: Die Tonaufnahme sollte jetzt deutlicher und ohne Rauschen zu hören sein.

Weiterführende Informationen

Projektseite: ➤ <http://audacityteam.org>

Portable App: ➤ http://portableapps.com/de/apps/music_video/audacity_portable

Handbuch und Tutorials (englisch): ➤ <http://manual.audacityteam.org>

Video-Tutorials (Version 1.3.6): ➤ <https://vimeo.com/album/45014>

Forum: ➤ <http://audacity-forum.de/forum>

AUDACITY im Physikunterricht: <http://pluslucis.univie.ac.at/PlusLucis/111/S18.pdf>

Wikipedia: <https://de.wikipedia.org/wiki/Audacity>

EXEARNING

Beschreibung

Mit dem Autorentool EXEARNING (EL) können Lernaktivitäten erstellt und für das Internet oder ► *Lern-Management-Systeme* wie ► *Moodle* exportiert werden. Der Workflow ist dabei der Arbeit mit einer Textverarbeitung ähnlich: Man erstellt einen Rahmentext und ergänzt multimediale Elemente wie Videos etc. Dann muss festgelegt werden, auf welche Eingaben der späteren Benutzer*innen hin wie reagiert werden soll, z.B. bei einer (falschen) Eingabe in einem Multiple-Choice-Test: Ob zur nächsten Frage weitergegangen oder eine (andere) Erklärung (wieder) eingeblendet werden soll. EL ist dabei relativ leicht zu erlernen; man benötigt (fast) keine Programmierkenntnisse. Andere, jüngere Abkömmlinge wie das »The new eXeLearning« ergänzen spezielle Funktionen.

Einsatzmöglichkeiten in der Schule

EL entlastet die Lehrkraft in der Unterrichtsstunde, indem es die Aktivität der Schüler*innen lenkt und diesen erlaubt, in ihrem persönlichen Tempo zu arbeiten und (bei gut gemachten Aktivitäten) verschiedene Lernpfade (z.B. nach Schwierigkeitsgrad gestuft) anbietet. EL unterstützt einen differenzierten Unterricht.

Das Material wird auf eine Webseite (oder in ein lokales Netzwerk) gestellt. Alternativ kann es in einem ► *Lern-Management-System* (z.B. ► *MOODLE*) gespeichert werden, auf dem sich die Schüler*innen anmelden. In beiden Fällen folgen sie den Anweisungen und werden dabei durch die zur Verfügung gestellten Materialien (Wikis, Videos, Webseiten etc.) unterstützt. Diese können bei Bedarf einzeln ausgetauscht oder angepasst werden, sodass das Material stets leicht auf dem aktuellen Stand gehalten werden kann.

Der vergleichsweise hohe Aufwand für die Erstellung des Materials wird langfristig durch die geringere Vorbereitungszeit bei wiederholter Durchführung kompensiert. Werden die Materialien arbeitsteilig erstellt und getauscht, so ist der Arbeitsaufwand kaum größer als bei »traditioneller« Unterrichtsvorbereitung für Arbeitsblätter.

Die ersten Schritte

Beispiel: Miniquiz

Im Folgenden soll ein Miniquiz mit Bewertung der Antwort (richtig/falsch) erstellt und als Webseite exportiert werden. Diese Quiz-Seite kann direkt im Anschluss lokal ausprobiert und hochgeladen werden. Diese Übung benötigt rund 30 Minuten.

Für einen ersten Einstieg ist es ratsam, die portable Version von EL herunterzuladen und zu starten. Es öffnet sich ein Fenster mit einem Pulldown-Menü oben, zwei Eingabefeldern links (»Übersicht« und »iDevices«), einem großen Eingabefeld rechts mit den Reitern »Autorenebene« und »Projekteigenschaften« sowie einer großen Statuszeile am unteren Rand. Die Übersicht zeigt die einzelnen Seiten des Materials an, die Autorenebene den Inhalt der aktuell ausgewählten Seite; hinter den iDevices verbergen sich die möglichen Bestandteile einer Seite.

Zunächst wird das iDevice »Freier Text« ausgewählt, um die Überschrift für das Quiz zu erstellen. In der Autorenebene wird daraufhin eine Textverarbeitung eingeblendet, in die der Text »Ein kleines Quiz« eingetragen und formatiert werden kann. Zum Abschluss klickt man den Bestätigungshaken (»Fertig«) links unterhalb des Textes, das Ergebnis wird sofort angezeigt.

Als nächstes iDevice wird die »Mehrfachauswahl« ausgewählt: Es dient als Multiple-Choice-Vorlage. In der Autorenebene wird der Text »Mehrfachauswahl« durch »Wozu dient eXeLearning?« ersetzt. Im Feld darunter ist der Text »Um Lernvideos zu schneiden.« einzutragen. Danach klickt man auf den Knopf »Eine weitere Antwort hinzufügen« und gibt in dem erscheinenden Textfeld »Um Online-Lernaktivitäten zu erstellen.« ein und setzt dahinter ein Häkchen bei »Richtig«. Nach einem Klick auf das Fertig-Icon links unten wird die fertige Übung angezeigt und kann ausprobiert werden. Markiert man nun die zweite Antwort als richtig und klickt auf »Feedback zeigen«, blendet EL »Richtig« bzw. »Falsch« ein.

Um die fertige Übung als Webseite zu exportieren, klickt man im Pulldown-Menü auf »Datei«, »Exportieren«, »Webseite« und schließlich auf »selbstbeschreibender Ordner«. Man wählt einen neuen, leeren Ordner mit einem aussagekräftigen Namen und bestätigt mit »ok«. Nun wird dieser Ordner und der Unterordner »newPackage« (z.B. mit dem Datei-Explorer) geöffnet. Darin finden sich rund 50 Dateien – darunter auch die Datei »index.html«. (Je nach Einstellung des Explorers wird die Endung »HTML« nicht angezeigt.) Mit einem Doppelklick auf diese Datei wird sie im Standardbrowser geöffnet und die erste eLearning-Aktivität kann ausprobiert werden. Wird der Ordner in das Web hochgeladen, so kann weltweit darauf zugegriffen werden.

Zum Abschluss wird die Arbeit für eine spätere Erweiterung gespeichert: mittels »Datei«, »Speichern unter«, Wahl des Ordners »Uebung« und Eingabe des Dateinamens »Quiz eXeLearning v0«. Mit dem Knopf »Speichern« werden alle notwendigen Informationen abgespeichert und eine kurze Bestätigung angezeigt: »Package saved to....« Mit »Datei«, »Beenden« kann EL verlassen werden.

Weiterführende Informationen

Projektseite v1.x: [↗ http://exelearning.org](http://exelearning.org)

Projektseite v2.x: [↗ http://exelearning.net](http://exelearning.net)

Portable App (Version 1): [↗ https://sourceforge.net/projects/exe/files/eXe-1.04/exe-ready2run-1.04.exe/download](https://sourceforge.net/projects/exe/files/eXe-1.04/exe-ready2run-1.04.exe/download)

Tutorials: [↗ http://exelearning-tutorials.de](http://exelearning-tutorials.de)

Einführung: [↗ http://paddelhannes.de/exetutorial-de/index.html](http://paddelhannes.de/exetutorial-de/index.html)

Video-Tutorials: [↗ https://youtube.com/channel/UCeyagnkRKDv9CYkT9-y36kw](https://youtube.com/channel/UCeyagnkRKDv9CYkT9-y36kw)

Bücher: [↗ http://exelearning-tutorials.de/b%C3%BCcher-zu-exelearning](http://exelearning-tutorials.de/b%C3%BCcher-zu-exelearning)

GLOSSAR



| **AGPL** – Akronym für *GNU Affero General Public License*. Bei dieser Lizenzform der GPL ist die Zurverfügungstellung des Quellcodes obligatorisch.

| **AIR** – Programm (oft: Adobe AIR; engl. *ADOBE INTEGRATED RUNTIME*), welches Webanwendungen auf Desktop-Computern laufen lässt und dazu zusätzlich installiert werden muss.

| **App** – Bezeichnung für Programme (engl. *applications*) für Mobilgeräte wie Smartphones oder Tablets. Technisch gibt es keine Besonderheit zu klassischen Programmen.

| **BASIC** – Akronym für *Beginner's All-purpose Symbolic Instruction Code*, deutsch: »symbolische Allzweck-Programmiersprache für Anfänger«. Mitte der 1960er entwickelte Programmiersprache, die in verschiedene Versionen seit den 1970ern vor allem auf PCs populär wurde; in der Microsoft-Welt ist Visual Basic bis heute verbreitet, z.B. in der Programmierung von Makros.

| **Betriebssystem** – Die Software eines Computers (dazu zählen auch Smartphones, Spielkonsolen, Fernseher usw.), die grundlegende Funktionen ermöglicht. Beispiele für Betriebssysteme sind *WINDOWS*, *OSX*, *LINUX*, *ANDROID*, *iOS* und weitere.

| **BMP** – Kurzform für Bitmap; ein von Microsoft entwickeltes Dateiformat zur Darstellung von Rastergrafiken.

| **BSD** – BSD (Akronym für *Berkeley Software Distribution*) bezeichnet die an der University of California in Berkeley selbst erstellten und von anderen Hochschulen gesammelten Weiterentwicklungen von AT&Ts Betriebssystem *UNIX*, die 1977 erstmals von Bill Joy unter dem Namen BSD veröffentlicht wurden. Zu den bedeutenden Beiträgen der Universität Berkeley

gehört die Integration von TCP/IP in *UNIX* und die BSD-Lizenz. Aus BSD sind die für verschiedene Zwecke optimierten Versionen *FreeBSD*, *NetBSD* und *OpenBSD* entstanden. *FreeBSD* liegt auch *APPLES OSX* zugrunde.

| **BSD-Lizenz** – Kurzform für »Berkeley-Software-Distribution-Lizenz«; eine Freie-Software-Lizenz, die im Gegensatz zur GPL nicht über ein Copyleft verfügt.

| **CAD-Programm** – CAD (engl. *computer-aided design*, deutsch: computerunterstütztes Konstruieren) bezeichnet im Wesentlichen das Erstellen und Bearbeiten von technischen Zeichnungen oder virtueller Modelle von dreidimensionalen Objekten.

| **CAS** – Kurzform für Computer-Algebra-System. CAS-Programme können nicht nur numerisch rechnen, sondern auch symbolisch umformen – also mathematische Gleichungssysteme zumindest teilautonom lösen.

| **Client** – Programm, das auf dem Endgerät eines Netzwerks ausgeführt wird und mit einem Server kommuniziert. Man nennt auch ein Endgerät selbst, das Dienste von einem Server abrufen, Client.

| **CNC-Technik** – CNC steht für *Computer Numeric Controlled* und bezeichnet computergesteuerte Maschinenteknik. Mit CNC-Technik werden alle möglichen Werkzeugmaschinen wie Laser, Fräsen und auch 3D-Drucker betrieben. Diese Maschinen werden sehr häufig mit G-Code angesteuert.

| **Codec** – Verfahren, das Daten oder Signale digital kodiert und dekodiert. Werden bereits kodierte Daten in einen anderen Code übersetzt (von einem Format in ein anderes umgewandelt), spricht man von Konvertierung, bei Audio- und Videodateien auch von Transkodierung.

| **Containerformat** – Ein Verbunddokument wie z.B. ein Film besteht aus verschiedenen Datenströmen (Video- und eine Audiodateien, die jeweils mit unterschiedlichen Codecs erstellt sein können), die in einer Containerdatei zusammengeführt werden. Die Metadaten der Containerdatei sichern, dass jeder Datenstrom vom jeweiligen Codec dekodiert und alle synchron dargestellt werden.

| **Cookies** – Cookies sind sehr kleine Dateien, die beim Aufruf von Webseiten Informationen auf dem lokalen Rechner speichern. Sie enthalten verschiedene Informationen, die das Browsen personalisieren und Daten zum verwendeten Rechner (z.B. Betriebssystem, Browserversion, Bildschirmgröße und vieles mehr) enthalten. Sie sind notwendig, um Sitzungen (z.B. bei erforderlichen Logins) zu ermöglichen, reduzieren allerdings auch den Datenschutz, da andere Webseitenbetreiber diese Cookies ebenfalls analysieren. Man hinterlässt einen ›digitalen Fußabdruck‹.

| **Copyleft** – Beim Copyleft (eine Anspielung auf Copyright) handelt es sich um eine Klausel in Lizenzen, welche vorschreibt, dass eine bearbeitete oder veränderte Version des Ursprungswerks nur unter den gleichen oder sehr ähnlichen Lizenzbedingungen veröffentlicht werden darf.

| **CPU** – (engl. *central processing unit*) Im Deutschen oft als Prozessor bezeichnet; ein Chip, der die Maschinenspracheprogramme ausführt.

| **CSV** – (engl. *comma-separated values*) Dateien mit der Endung ».csv« enthalten Daten in Textform, wobei die Daten durch ein besonderes Zeichen voneinander getrennt sind – meist ein Komma. Dieses Dateiformat dient als kleinster gemeinsamer Nenner zum Datenaustausch, beispielsweise zwischen unterschiedlichen Textverarbeitungsprogrammen.

| **DOC** – Die Dateinamenendung ».doc« ist das veraltete Dokumentenformat der Textverarbeitung Microsoft Word, dessen Spezifikationen inzwischen veröffentlicht sind.

| **DTP-Programm** – Ein Desktop-Publishing-Programm beinhaltet Funktionen, die aus dem Layout- und Druckbereich stammen. Anders als ein Textverarbeitungsprogramm können mit ihm Texte und Grafiken gesetzt werden.

| **ECDL** – Die *European Computer Driving Licence* (Europäischer Computer-Führerschein) ist eine Zertifizierung von Wissen und Fertigkeiten im Umgang mit dem Computer und Anwendungsprogrammen. Er kann in der Schule sehr gut als Lehrplan für die Arbeit mit Computern und Anwendungsprogrammen genutzt werden. Mit dem ECDL wird keine erweiterte Medienkompetenz vermittelt: Ihm fehlen Elemente wie Kritik-, Analyse- oder Wertekompetenz in digitalen Medien.

| **FFMPEG** – Wichtiges Freie-Software-Projekt für Codecs. Es besteht aus einer Reihe von freien Computerprogrammen und Programm-bibliotheken, die digitales Video- und Audio-material aufnehmen, konvertieren, streamen und in verschiedene Containerformate verpacken können. Seine Bibliothek libavcodec enthält eine umfangreiche Sammlung von Audio- und Videocodecs.

| **Fork** – Abspaltung in der Softwareentwicklung; weiterentwickelte und veränderte Softwareprojekte.

| **Freeware** – Kostenlose proprietäre Software. Nicht zu verwechseln mit Freier Software.

| **G-Code** – Steuerungsbefehle für computer-unterstützte Laser, Fräsen und 3D-Drucker.

| **GNU** – Akronym für »GNU's Not Unix« (deutsch: »GNU ist nicht Unix«), das von Richard Stallman 1984 gestartete Projekt, ein Betriebssystem zu schreiben, das funktional äquivalent zu UNIX ist, aber keine einzige Zeile geschützten Code enthält und in freier Kooperation weiterentwickelt werden kann. Diese Freiheit wird durch die Lizenz GPL abgesichert.

| **GNUPG** – Kurzform für *GNU Privacy Guard* (GNU-Privatsphärenschutz); System zum Ver- und Entschlüsseln von Daten.

| **GPL** – Die *GNU General Public License* ist eine in der Freien Software weit verbreitete Copyleft-Lizenz des GNU-Projektes, die alle Freiheiten gewährt und vorschreibt, dass Kopien und veränderte Versionen freier Programme nur mit denselben Freiheiten verbreitet werden dürfen.

| **Grafische Programmiersprache** – Während die meisten Programmiersprachen textuelle Befehle erwarten, werden bei dieser Gattung die Befehle durch Icons symbolisiert. Programmieren bedeutet dann, grafische Symbole in die gewünschte Reihenfolge auf dem Bildschirm zu bringen.

| **Grafische Benutzeroberfläche** – (engl. *graphical user interface*, kurz GUI) Die Interaktion zwischen Mensch und Maschine passiert hierbei durch das Hantieren mit Symbolen anstatt der Ein- und Ausgabe von Text. Heute de facto Standard für die Computerbedienung.

| **Groupware** – Eine Software, deren Ziel es ist, die Zusammenarbeit verschiedener Parteien durch das Bereitstellen von Werkzeugen zu vereinfachen und zu verbessern. Hierzu zählen z.B. Kalender, E-Mail oder Werkzeuge aus dem Projektmanagement.

| **Hacker** – (1) Als Ehrentitel für jemanden, der besonders kreative und kunstvolle techni-

sche Lösungen findet; nachgewiesen seit den 1950ern; in diesem Sinne gibt es Stellenausschreibungen für »Unix-Hacker«. (2) Eine Kultur von Computer- und Netz-Faszinierten, die sich in Clubs (z.B. CCC) zusammenfindet und sich Infrastrukturen (z.B. GitHub, Lizenzen) und eine Hacker-Ethik gegeben hat. Im Sinne von (1) und (2) ist die Kultur der Freien Software eng mit der der Hacker verwandt. (3) Seit dem Internet auch als Bezeichnung für Sicherheitsexperten, die Schwachstellen entdecken, um sie aufzudecken oder um sie auszunutzen; vgl. Unterschied zwischen White-Hat Hacker (gesetzestreu) und Black-Hat Hacker (handelt mit krimineller Energie).

| **HTML** – (engl. *hypertext markup language*) Mit dieser Sprache werden Webseiteninhalte so gestaltet, dass sie von Browsern angezeigt werden können. Inzwischen bildet HTML nur noch das Grundgerüst einer Webseite, das durch vielfältige – vor allem interaktive und multimediale Abschnitte – ergänzt wird.

| **JAVA-Applets** – Programme, die in der Programmiersprache JAVA geschrieben wurden und meist aus dem Internet geladen werden, jedoch dann lokal im Browser durch die JAVA-Laufzeitumgebung (JRE) ausgeführt werden und keine Daten mit dem Internet austauschen müssen.

| **JAVA-LAUFZEITUMGEBUNG (JRE)** – (engl. *JAVA RUNTIME ENVIRONMENT*) Damit JAVA-Applets unter möglichst jedem Betriebssystem und für jede CPU funktionieren, müssen die JAVA-Programme durch das JRE für das jeweils aktuelle System übersetzt werden. Deshalb benötigen JAVA-Applets eine installierte JRE.

| **Lern-Management-System (LMS) / Lernplattform** – (engl. *learning management system*) Ein komplexes Softwaresystem, welches Lernmaterialien bereitstellt, deren

Bearbeitung protokolliert und möglichst situative Kommunikationswerkzeuge zur Verfügung stellt.

| LGPL – Die *GNU Lesser General Public License* ist eine Freie-Software-Lizenz, die es Entwickler*innen ermöglicht Software unter LGPL mit eigener unfreier Software zu kombinieren. Im Gegensatz zur GPL gibt es nicht die Verpflichtung die eigenen Programmcode Teile ebenfalls unter einer Freie-Software-Lizenz zu veröffentlichen.

| Linux-Distribution – Eine Sammlung aufeinander abgestimmter Software, die zusammen ein vollwertiges Betriebssystem ergibt.

| Lizenz – Ein Vertrag über die Nutzungsbedingungen urheberrechtlich geschützter Werke. Man unterscheidet zwischen Individuallizenzen, die jeweils einzeln ausgehandelt werden, und Pauschallizenzen, die jedermann die jeweiligen Nutzungsmöglichkeiten erlauben, sofern die Lizenzbedingungen eingehalten werden und die automatisch widerrufen werden, wenn gegen diese Bedingungen verstoßen, z.B. eine Freie Software ohne Hinweise auf Autoren, Lizenz und Quellcode weiterverbreitet wird. In der Freien Software werden Pauschallizenzen wie GPL oder BSD-Lizenz verwendet. Für freie Inhalte wie Bilder, Klänge und Texte haben sich vor allem die Pauschallizenzen des Creative-Commons-Projektes durchgesetzt.

| M.I.T. – Massachusetts Institute of Technology; US-amerikanische Eliteuniversität, die im Bildungsbereich sehr aktiv ist.

| Microsoft .NET Framework – Teil der von Microsoft herausgegebenen .NET-Software-Plattform, welche zur Entwicklung und Ausführung anderer Anwendungssoftware genutzt wird.

| MIT-Lizenz – Kurzform für »Massachusetts-Institute-of-Technology-Lizenz«; eine Freie-Software-Lizenz, die im Gegensatz zur GPL nicht über ein Copyleft verfügt.

| Mono – Freie Implementierung von Microsofts .NET Framework.

| Moodle – Ein weit verbreitetes freies Lern-Management-System (LMS), welches Materialien und Lernaktivitäten (z.B. Aufgaben, Wikis, Tests) in »Kursräumen« zur Verfügung stellt.

| MPL – Die MOZILLA PUBLIC LICENSE ist eine Freie-Software-Lizenz, die hauptsächlich in Projekten von MOZILLA vorkommt (z.B. FIREFOX oder THUNDERBIRD). Sie enthält ein schwaches Copyleft und kann als möglicher Kompromiss zwischen GPL und BSD-Lizenz gesehen werden.

| NDA – (engl. *non-disclosure agreement*) Vertraulichkeitsvereinbarung, auch Geheimhaltungsvereinbarung; macht ein Unternehmen Angestellten oder Externen (Kund*innen, Partner*innen) sensitive Information zugänglich, verpflichtet es sie mit einem NDA zu Nichtweitergabe.

| ODT – OpenDocument-Text; ein offenes Dokumentenformat für Texte; siehe hierzu auch ODF.

| ODF – Das OpenDocument-Format ist ein offener und international genormter Standard für Dokumente. Er ist entwickelt worden, um programm- und betriebssystemübergreifend einen problemlosen Austausch von Dokumenten zu ermöglichen.

| ODP – OpenDocument-Presentation; ein offenes Dokumentenformat für Präsentationsfolien; siehe hierzu auch ODF.

| **Open Access** – In den 1990er Jahren eskalierten die steigenden Preise für wissenschaftliche Publikationen und die nicht mitwachsenden Etats der Bibliotheken zur Zeitschriftenkrise. Die Wissenschaft besann sich auf ihre Mertonschen Werte und startete die Open-Access-Bewegung. Ihr Ziel ist, dass wissenschaftliche Ergebnisse der Gemeinschaft frei zugänglich sind, zumal, wenn sie aus öffentlich geförderter Forschung hervorgehen. Die Bewegung hat das wissenschaftliche Verlagswesen grundlegend verändert und Open Access zur Auflage in der öffentlichen Forschungsförderung gemacht.

| **Open Data** – Die freie Verfügbar- und Nutzbarkeit von Daten, vor allem der öffentlichen Hand und der Wissenschaft, die ähnlich wie Freie Software zur freien Weiternutzung veröffentlicht werden: Daten über Bevölkerung, Verkehr, Wetter, Karten, Genome bis hin zu Studien des wissenschaftlichen Dienstes des Bundestages. Immer mehr Bundesländer gehen im Zuge der Entwicklung von Informationsfreiheits- und Transparenzgesetzen dazu über, ihre Daten aktiv zu veröffentlichen. Datenjournalismus und neue Formen datengetriebener Wissenschaft sind die ersten Früchte dieser Bewegung.

| **OS X** – APPLES Desktop-Betriebssystem; geht auf NeXTStep von NeXT zurück, jenes Unternehmen, das Steve Jobs 1985 nach seinem vorübergehenden Ausscheiden bei APPLE gegründet hatte. Darwin, das Kern-Betriebssystem von OS X, ist ein freies UNIX, in dem wichtige Teile von BSD stammen; die über Darwin liegenden Schichten bis zur grafischen Benutzeroberfläche sind proprietär; iOS ist eine abgewandelte Form von OS X, die APPLE für seine mobilen Geräte verwendet.

| **Peers** – In der Soziologie sind Peers Gleichrangige. In der Wissenschaft ist der Peer-Review einer der wichtigsten Qualitätssicherungsprozesse, unter Gleichen und blind gegenüber den beteiligten Personen. In der Informatik bezeichnen »Peer-to-Peer« Netze, in denen jeder Knoten sowohl Server als auch Client ist. Anhand von Freier Software und Wikipedia hat der Harvard-Rechtsprofessor Yochai Benkler die »Allmende-basierte Peer-Produktion« als dritte Art der Produktionsorganisation neben Märkten und zentraler Planwirtschaft identifiziert.

| **Pixelgrafik** – siehe Rastergrafik

| **Portable App** – Programm (häufig für das Betriebssystem Windows), das nur auf einem USB-Stick installiert wird und damit den Computer selber unverändert lässt.

| **PDF** – Das *Portable Data Format* PDF ist ein Format für Dokumente, die nicht mehr bearbeitet werden sollen.

| **Physical Computing** – Bezeichnung für meist edukative oder künstlerische Low-Cost-Systeme aus Hard- und Software, bei denen Sensoren und Aktoren Verwendung finden. Teil der Maker-Bewegung.

| **Portable Software** – Programme, meist für Windows, die nicht installiert werden müssen, sondern direkt von einem Datenträger (oft ein USB-Stick) gestartet werden können. Daher benötigen portable Programme fast keine Rechte auf dem Betriebssystem und lassen das Betriebssystem (meist) unverändert.

| **Proprietäre Software** – Software, die die Rechte der Anwender*innen und Nutzer*innen hinsichtlich Verbreitung, Anpassung und Nutzung stark einschränkt. Sie ist das Gegenteil von Freier Software und kann kostenlos oder kostenpflichtig angeboten werden.

| **Programmiersprache** – Künstliche Sprachen, die relativ leicht zu lernen sind und von Computern »verstanden« werden. Sie werden benötigt, um mit relativ geringem Aufwand Programme zu schreiben.

| **Quellcode** – (engl. *source code*) Anweisungen in einer Programmiersprache, die relativ leicht verstanden werden können und – nach einer Übersetzung in Maschinensprache – von Computern ausgeführt werden können.

| **Rastergrafik** – Rastergrafiken bestehen aus Bildpunkten (Pixeln) mit einem Farbwert, die in einem zweidimensionalen Raster angeordnet sind. Mit Mal- und Zeichenprogrammen erstellte Bilder, Scans und Fotos sind Rasterbilder. Vergrößert man Rasterbilder, vergrößern sich auch die rechteckigen Pixel und es entsteht der charakteristische Treppcheneffekt.

| **RAW** – Format der Rohdaten des Bildsensors von digitalen Spiegelreflexkameras (DSLR) und manchen Kompaktkameras. Anders als Bildformate wie PNG oder JPG sind RAW-Bilder unkomprimiert und noch nicht in einen bestimmten Farbraum transformiert, sie enthalten somit die volle ursprüngliche Farbtiefe.

| **Rendern** – Rendern bezeichnet unter anderem das computerbasierte Erstellen einer Grafik aus Daten zu dreidimensionalen Objekten bzw. die Verarbeitung von Rohdaten zu Grafiken.

| **Server** – Meist ein Computer, der anderen Rechnern seine Dienstleistungen vorwiegend per Netzwerk zur Verfügung stellt. Beispielsweise eine Datenbank, in der die angeschlossenen Rechner (Clients) ihre Daten ablegen und gemeinsam benutzen.

| **STL** – Dateien im STL-Format enthalten geometrische Informationen von dreidimensionalen Körpern.

| **TCP/IP** – Technische Grundlage des Internet. Das ab 1973 entwickelte TCP (engl. *transmission control protocol*) dient der zuverlässigen Übertragung von Datenpaketen über unzuverlässige Kanäle. Für die zeitkritische Sprachübertragung kam 1978 IP (engl. *internet protocol*) hinzu, welches das *user datagram protocol* (UDP) spezifiziert. Heute wird der Wechsel des seit 1982 verwendeten IP v4 auf IP v6 vorbereitet.

| **Tool** – Bei einem Tool (engl. für »Werkzeug«) handelt es sich um eine (meist im Funktionsumfang kleine aber hilfreiche) Softwareanwendung.

| **Tor-Netzwerk** – Das Tor-Netzwerk besteht aus einem Netz von Servern, die das Surfen und allgemein Aktivitäten im Internet anonymisieren. Zur Nutzung des Netzwerks müssen Browser bzw. die Computersysteme in spezieller Art und Weise konfiguriert werden.

| **Transkodieren** – Beschreibt das Umwandeln von Audio- oder Videodateien von einem Format in ein anderes.

| **UNIX** – Das am weitesten verbreitete und eines der einflussreichsten Betriebssysteme der Computergeschichte. UNIX steht allgemein für Betriebssysteme, die ihren Ursprung im UNIX-System von AT&T Bell Labs haben oder dessen Konzepte implementieren. Dank seiner Portabilität und Entwicklungsoffenheit finden sich unixoide Systeme und UNIX-Derivate heute überall: auf Supercomputern und Servern, auf Laptops und Mobiltelefonen oder eingebettet in Steuergeräten, Medizintechnik, Autos oder WLAN-Routern.

| **UUCP** – (engl. *unix to unix copy*) 1976 an den AT&T Bell Labs entwickeltes Protokoll, um über Telefonwählleitungen Daten zwischen UNIX-Rechnern auszutauschen; ermöglicht Einrichtungen, die sich keine Standleitung leisten

können, die Nutzung von asynchronen Internet-Diensten wie E-Mail oder Newsgroups.

| **USENET Newsgroups** – Das erste soziale Netzwerk vor dem und im Internet, dezentral, mit zahllosen Diskussionsforen, in denen Text- oder Binär-E-Mails ausgetauscht werden. Teilnehmen kann man über einen Newsreader wie MOZILLA THUNDERBIRD, über ein Webbrowser-Add-on z.B. für MOZILLA FIREFOX oder über eine Webschnittstelle wie z.B. GOOGLE GROUPS.

| **Video-Tutorial** – Eine Anleitung in Form eines Filmes, in der z.B. die Nutzung einer Software dargestellt wird.

| **Vektorgrafik** – Vektorgrafiken bestehen (anders als Pixelgrafiken) nicht aus einzelnen Bildpunkten, sondern aus Bildinformationen, die Daten aus einem Koordinatensystem ähneln. Die Bilder werden damit nicht aus einzelnen Punkten, sondern aus den „Verbindungslinien“ der Eigenschaften (Farbe, Breite und Ähnlichem) erstellt.

| **WYSIWYG** – Akronym aus *What-You-See-Is-What-You-Get*. Damit ist gemeint, dass man während der Arbeit an einem Material (z.B. einem Werbeflyer) bereits sehen kann, wie der aktuelle Arbeitsstand ausgedruckt aussehen würde. WYSIWYG-Systeme benötigen relativ leistungsstarke Hardware.

| **XML** – Eine maschinenlesbare Sprache, mit deren Hilfe hierarchisch strukturierte Daten in Form von Textdateien dargestellt werden können. Sie hilft, Daten unabhängig von der genutzten Plattform auszutauschen.

Blank page with horizontal dotted lines for writing.



Medien
in die
Schule

Alle Materialien der Unterrichtsreihe » Medien in die Schule « sowie zahlreiche Zusatz-
informationen sind online verfügbar unter www.medien-in-die-schule.de.

Ein Projekt von

FSM

...fsf FREIWILLIGE
SELBSTKONTROLLE
FERNSEHEN

Google