

## Topthemen dieser Ausgabe

### Bildbearbeitung mit GIMP – Teil 1: Bildteile entfernen

Eine oft gewünschte Funktion ist, in Bildern einzelne, störende Teile zu entfernen. Diese Aufgabe ist weniger trivial, als sie auf den ersten Blick wirkt, denn: Was soll anstelle der entfernten Bereiche im Bild erscheinen? Es gibt daher eine ganze Reihe von Möglichkeiten, diese Aufgabe zu bewerkstelligen. Der erste Teil der GIMP-Reihe soll zeigen, wie es geht. ([weiterlesen](#))

Seite 5



### Tiny Tiny RSS – Ein web-basierter Feed-Aggregator

Das Internet – unendliche Weiten. Wir schreiben das Jahr 2015. Dies sind die Abenteuer des Autors, der mit zwei Workstations, einem Laptop und je einem Smartphone und Tablet unterwegs ist und News-Feeds liest. Viel in der Welt unterwegs, dringt er in ungeahnte Problem-Welten vor, die sicherlich einigen Lesern bekannt vorkommen. ([weiterlesen](#))

Seite 25



### Fuchs in Flammen: Mozilla Flame im Test

Der Markt für mobile Betriebssysteme ist 2014 fest in einer Hand: Googles Android dominiert weltweit den ganzen Markt. Einzig Apples iOS kann teilweise noch seine Stellung im zweistelligen Prozentbereich behaupten. Es gibt aber auch noch zahlreiche andere Betriebssystemhersteller, die einen Fuß in die Tür bekommen wollen. Einer davon ist Mozilla mit Firefox OS. Dieser Artikel stellt das Betriebssystem zusammen mit dem Referenz-Entwickler-Gerät namens Flame vor. ([weiterlesen](#))

Seite 35



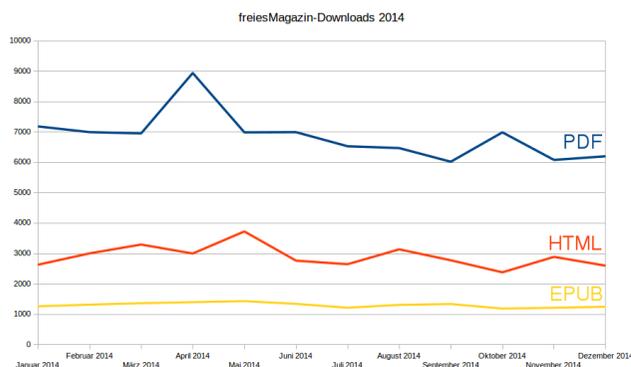
## Editorial

### Jahresabschluss 2014

Jahreswechsel dienen sehr oft dazu auf das Vergangene zu blicken und für die Zukunft daraus zu lernen. Bei freiesMagazin gab es 2014 Kurzgeschichten und einen Programmierwettbewerb und natürlich viele viele Artikel.

Insgesamt 121 Artikel von 40 verschiedenen Autoren auf 527 Seiten konnten wir 2014 in freiesMagazin veröffentlichen. Das sind gut 20 Artikel mehr zum Vorjahr und auch acht Autoren mehr, was sehr erfreulich ist, da es zeigt, dass verschiedene Menschen Interesse daran haben, ihr Wissen anderen frei zur Verfügung zu stellen.

Die Downloadzahlen gingen im Laufe des Jahres etwas zurück. Bei der PDF-Ausgabe lagen die Downloads am Jahresanfang noch bei knapp über 7000, am Jahresende nur noch bei knapp 6000. Die HTML-Ansichten der Mobilausgabe pendeln



freiesMagazin-Downloadzahlen 2014. 🔍

zwischen 2500 und 3200. Und die mobile EPUB-Version hat ca. 1200 bis 1300 im Durchschnitt. In der Summe macht dies also immer noch über 10000 Downloads für PDF, HTML und EPUB (ggf. mit Doppelungen), was nach wie vor ein sehr guter Wert ist.

Das Experiment der Kurzgeschichten haben wir erst einmal unterbrochen, auch wenn laut der Umfrage [1] genau 50% der Leser gerne weitere sehen möchten. Die Resonanz zum Programmierwettbewerb war zwar etwas gering [2], aber diesen wollen wir auch 2015 wieder stattfinden lassen.

Zusätzlich gibt es in dieser Ausgabe auch wieder den *Jahresindex 2014*, bei dem alle Artikel des vergangenen Jahres nach Schlagworten sortiert aufgelistet sind.

Wir wünschen allen Lesern ein tolles Jahr 2015!

Ihre freiesMagazin-Redaktion

### LINKS

- [1] <http://www.freiesmagazin.de/20141207-moechten-sie-weitere-kurzgeschichten-in-freiesmagazin-lesen-2>
- [2] <http://www.freiesmagazin.de/20141222-siebter-programmierwettbewerb-ist-beendet>



Teilen



Kommentieren

### Inhalt

#### Linux allgemein

Der Dezember im Kernelrückblick S. 3

#### Anleitungen

Bildbearbeitung mit GIMP – Teil 1 S. 5

Git Tutorium – Teil 2 S. 11

Spacewalk – Teil 4: Verwaltung von Solaris-Systemen S. 17

Kurztipp: tmux mit Byobu nutzen S. 23

#### Software

Tiny Tiny RSS – Ein web-basierter Feed-Aggregator S. 25

Calibre – Teil 1: Installation und Erst-Konfiguration S. 31

#### Hardware

Fuchs in Flammen: Mozilla Flame im Test S. 35

#### Community

Rezension: Programmieren lernen mit Python S. 44

Rezension: Wireshark 101 - Einführung in die Protokollanalyse S. 46

#### Magazin

Editorial S. 2

Leserbriefe S. 48

Veranstaltungen S. 50

Vorschau S. 50

Konventionen S. 50

freiesMagazin-Index 2014 S. 51

Impressum S. 60

## Der Dezember im Kernelrückblick von Mathias Menzer

**B**asis aller Distributionen ist der Linux-Kernel, der fortwährend weiterentwickelt wird. Welche Geräte in einem halben Jahr unterstützt werden und welche Funktionen neu hinzukommen, erfährt man, wenn man den aktuellen Entwickler-Kernel im Auge behält.

### Linux 3.18

Was haben Linux 3.18 und die Dezember-Ausgabe von [freiesMagazin](#) gemeinsam? Beide wurden am ersten Dezember-Sonntag veröffentlicht [1]. Allerdings stellt die aktuellste Version des Linux-Kernels mit 117 MB an komprimiertem Quelltext definitiv die schwerere Lektüre dar. In den 63 Tagen, die seit Beginn des Merge Window für 3.18 verstrichen sind, haben die Linux-Entwickler nicht nur Fehler behoben, sondern auch neue Funktionen eingebracht.

### Überlagert

Die für viele interessanteste Änderung dürfte das neue OverlayFS sein. Dieses Dateisystem schafft eine zusätzliche, beschreibbare Ebene über einem nicht beschreibbaren Dateisystem wie beispielsweise UDF [2]. Dies dient beispielsweise dazu, Live-Systeme, die von CD, DVD oder aus dem lokalen Netzwerk gestartet werden, vom Anwender anpassen lassen zu können. Die Änderungen werden eben nicht auf das Startmedium geschrieben, sondern auf die durch OverlayFS bereitgestellte zweite Schicht. Diese lässt sich wiederum

auf einem Datenträger wie einem USB-Stick speichern, und beim nächsten Start bleiben Änderungen wie Anpassungen der Desktop-Einstellungen oder gegebenenfalls auch Systemaktualisierungen erhalten. Dabei ist OverlayFS nicht zwingend auf schreibgeschützte Systeme beschränkt, es kann natürlich auch über Dateisysteme gelegt werden, die zwar beschrieben werden können, wo man dies aber nicht möchte. Als Anwendungsbeispiel sei hier die Virtualisierung genannt, wo Anwendungen in einen eigenen Bereich eingesperrt werden sollen, ohne selbst die darunterliegende Dateistruktur manipulieren zu können.

### Eingelagert

Linux 3.16 brachte die Möglichkeit mit, dass Userspace-Anwendungen auf den Speicher von Intel-Grafikkarten zugreifen können. Damit lassen sich beispielsweise das Bereitstellen von Texturen oder Fensterinhalten durch eine Anwendung direkt im Grafikspeicher realisieren. Dadurch werden diese Daten direkt dem Grafikprozessor zur Darstellung zugeführt und müssen nicht noch einmal durch den Grafiktreiber kopiert werden. Dies schont zum Einen die Bandbreite des Speicherbus und nebenbei geht es auch schneller. Mit Linux 3.18 steht diese Funktion nun auch für Radeon-Grafikchips von AMD/ATI zur Verfügung.

### Gefiltert

„eBPF“ basiert auf dem Berkeley Packet Filter [3], einem Filter für Netzwerkpakete. eBPF bietet nun

einen Systemaufruf `bpf()`, mittels dem Anwendungen, die im Kontext des Benutzers laufen, Regelsätze an eBPF übergeben um gezielt Daten aus dem Netzwerkverkehr abzugreifen. Mit Programmen wie `tcpdump` war dies bislang nur mit Superuser-Berechtigung möglich.

### Vernetzt

Speziell im Netzwerk-Umfeld hat sich viel getan. So soll „Data Center TCP“ (DCTCP) Netzwerkverbindungen innerhalb von Datenzentren optimieren. DCTCP ist speziell auf die typischen Belange von Rechenzentren zugeschnitten und sorgt für geringere Latenzen sowie einen höheren Durchsatz, indem die Netzwerk-Puffer möglichst wenig genutzt werden.

Eine weitere Neuerung ist die Einführung von „Generic Network Virtualization Encapsulation“ oder kurz „Geneve“. Dieses Protokoll dient zur Netzwerkvirtualisierung und soll die Trennung der Netzwerk-Anbindungen virtueller Maschinen auf einem Host-System vereinfachen. Es hat zwar derzeit noch den Status als Draft (Entwurf) [4], wird jedoch von Intel, Microsoft, Red Hat und VMware vorangetrieben und dürfte damit auf viel Akzeptanz stoßen und somit einen zügigen Weg zum Internet-Standard finden.

Der Aufwand des Kernels, um Pakete für den Versand über das Netzwerk vorzubereiten ist recht hoch, sodass das Übertragen einer bestimmten

Datenmenge mittels vieler kleiner Pakete erheblich langsamer vonstatten geht als mittels großer Pakete. „Transmission Queue Batching“ ermöglicht es nun, in Verbindung mit einigen Treibern für 10- und 40-GBit-Netzwerkchips, die Übertragung einzelner Pakete aufzuschieben. Ist nun eine Anzahl an Paketen zusammengekommen, werden sie in einem Rutsch an die Netzwerkhardware übermittelt und können so die zur Verfügung stehende Bandbreite besser auslasten.

„Foo-over-UDP“ (FOU) mag sich ersten Moment eher nach Nonsense als einem nutzbringenden technischen Konzept anhören, hat aber dennoch einen zweckdienlichen Hintergrund. Konkret beschreibt FOU die Möglichkeit, IP-Pakete nochmal einzupacken und über UDP-Verbindungen zu versenden. Da die IP-Protokollschicht unterhalb von dem davon abhängigen UDP-Protokoll liegt und damit ein zusätzlicher IP-Header vor das Netzwerkpaket geklebt werden muss, wirkt dies eigentlich immer noch recht unnützlich. UDP liefert einige Mechanismen, die den Betrieb von getunnelten Netzwerkverbindungen erheblich vereinfachen, bleibt dabei jedoch noch unter den bei der Nutzung von TCP auftretenden Aufwänden. So können ab Linux 3.18 die bekannten Tunnelprotokolle GRE [5], IPinIP [6] und dem mit letzterem vergleichbaren SIT auf UDP als Unterbau aufsetzen und damit beispielsweise in den Genuss der Nutzung von UDP-Ports kommen.

Eine vollständige, jedoch englischsprachige Liste der Neuerungen bietet die Seite Kernel New-

bies [7], die auch einen sehr ausführlichen Überblick über die Änderungen der Treiber pflegt [8].

### Linux 3.19

Nicht ganz zwei Wochen nach Veröffentlichung von Linux 3.18 schloss Torvalds mit der Veröffentlichung der ersten Entwicklerversion von 3.19 das Merge Window [9]. Es handelt sich nicht um den größten -rc1 in der Reihe der 3er-Kernel, aber unter den Top 3 ist er in jedem Fall vertreten.

Unter den Neuerungen wird eine neue Architektur sein, die auf den Namen „nios2“ hört und aus dem Hause Altera stammt. Es handelt sich hierbei um ein 32bit-FPGAs also frei programmierbare Integrierte Schaltungen, die dadurch auch im Betrieb sehr genau auf ihre Aufgaben hin konfiguriert werden können. Sie finden im Embedded-Bereich sowie bei der Signalverarbeitung Verwendung.

Außerdem wurde Unterstützung für Intels MPX-Technologie [10] aufgenommen. Prozessoren mit dieser Erweiterung sind zwar noch nicht verfügbar, doch Linux-Anwender werden dann gleich in den Genuss der Vorteile kommen, wie zum Beispiel verbessertem Schutz von Speicherüberläufen.

Erwartungsgemäß fiel die Größe der zweiten Entwicklerversion [11] recht gering aus - 65 Commits. Darunter finden sich dann auch keine aufregenden Änderungen. Die Mehrzahl waren Patches, ein Revert im Bereich der Speicherverwaltung und den Fix für einen Flüchtigkeitsfehler im Code des Dateisystems Lustre [12]. Es bleibt zu erwart-

ten, dass sich die Dynamik der Entwicklung nach den Weihnachtsfeiertagen und dem Jahreswechsel ändern wird.

### LINKS

- [1] <https://lkml.org/lkml/2014/12/7/202> 
- [2] [https://de.wikipedia.org/wiki/Universal\\_Disk\\_Format](https://de.wikipedia.org/wiki/Universal_Disk_Format)
- [3] [https://de.wikipedia.org/wiki/Berkeley\\_Filter](https://de.wikipedia.org/wiki/Berkeley_Filter)
- [4] <http://tools.ietf.org/html/draft-gross-geneve-01> 
- [5] [https://de.wikipedia.org/wiki/Generic\\_Routing\\_Encapsulation\\_Protocol](https://de.wikipedia.org/wiki/Generic_Routing_Encapsulation_Protocol)
- [6] [https://en.wikipedia.org/wiki/IP\\_in\\_IP](https://en.wikipedia.org/wiki/IP_in_IP) 
- [7] [http://kernelnewbies.org/Linux\\_3.18](http://kernelnewbies.org/Linux_3.18) 
- [8] [http://kernelnewbies.org/Linux\\_3.18-DriversArch](http://kernelnewbies.org/Linux_3.18-DriversArch) 
- [9] <https://lkml.org/lkml/2014/12/20/207> 
- [10] [https://en.wikipedia.org/wiki/Intel\\_MPX](https://en.wikipedia.org/wiki/Intel_MPX) 
- [11] <https://lkml.org/lkml/2014/12/28/142> 
- [12] [https://de.wikipedia.org/wiki/Lustre\\_\(Dateisystem\)](https://de.wikipedia.org/wiki/Lustre_(Dateisystem))

**Autoreninformation** 

**Mathias Menzer** ([Webseite](#)) behält die Entwicklung des Linux-Kernels im Blick, um über kommende Funktionen von Linux auf dem Laufenden zu bleiben.

 [Teilen](#)  [Kommentieren](#)

## Bildbearbeitung mit GIMP – Teil 1: Bildteile entfernen von Die GIMPer

**E**ine oft gewünschte Funktion ist, in Bildern einzelne, störende Teile zu entfernen. Diese Aufgabe ist weniger trivial, als sie auf den ersten Blick wirkt, denn: Was soll anstelle der entfernten Bereiche im Bild erscheinen? Es gibt daher eine ganze Reihe von Möglichkeiten, diese Aufgabe zu bewerkstelligen. Der erste Teil der GIMP-Reihe soll zeigen, wie es geht.

**Redaktioneller Hinweis:** Dieser Text ist ein angepasster Auszug aus dem Buch „Bildbearbeitung mit GIMP – Die 101 wichtigsten Tipps: Teil 1 – Grundlegende Funktionen“ [1].

### Einleitung

Einige oft verwendete Möglichkeiten zum Entfernen von Bildteilen sind:

- Klonen oder Heilen: Die unerwünschten Bereiche werden manuell mit Material aus der Umgebung übermalt. Klonen funktioniert am besten in kleinen Bereichen mit gleichmäßig strukturierter Umgebung. Heilen lässt sich auch bei ungleichmäßigen Hintergründen einsetzen.
- Resynthese, Heal-Selection (oder Smart Remove Selection) oder Patch-based-Inpaint: Mehrere Methoden, diese Manipulationen mehr oder weniger automatisiert vorzunehmen. Resynthese funktioniert gut in großen Bereichen mit gleichmäßigen Hintergründen. Resynthese erlaubt weitere Aktionen, wie etwa den

Texture-Transfer. Patch based Inpaint ist derzeit hauptsächlich für das Entfernen kleiner Bereiche in uneinheitlichen Umgebungen geeignet.

Dazu kommen noch viele weitere Methoden, die bei speziellen Situationen Anwendung finden.

### Klonen

„Klonen“ bezeichnet ein Verfahren, bei dem der Anwender mit einem – normalerweise weichen – Pinsel im Bild über unerwünschte Bildteile malt und damit diese verschwinden lässt. Die Besonderheit beim Klonen besteht darin, dass die aufgetragene Farbe automatisch aus der Nähe der zu übermalenden Fläche genommen wird, damit das Übermalen möglichst unauffällig ist. GIMP stellt drei Werkzeuge für diese Arbeiten bereit:

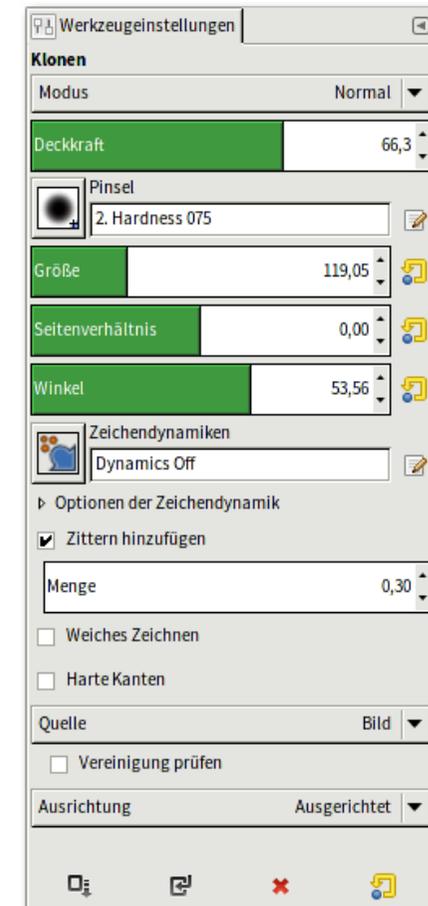
- das klassische Klonwerkzeug
- eine Variante für perspektives Klonen und
- eine als Heilen bezeichnete Variante, die den Farbauftrag fast automatisch an die Umgebung anpasst.

### Tipp 55: Das Klonwerkzeug verstehen

GIMPs Klonwerkzeug (Taste **C**) gehört zu den Malwerkzeugen, mit denen es die meisten der dort typischerweise vorhandenen Optionen teilt.

**Hinweis:** Die effektive Verwendung dieses Werkzeugs benötigt einige Übung. Man sollte sich nicht entmutigen lassen, wenn man zunächst nicht

die gewünschten Ergebnisse erzielt. Mit einiger Übung geht das dann. Generell sollte man sich aber zunächst mit den Malwerkzeugen vertraut machen, bevor man sich am Klonen versucht.



Das Klonwerkzeug arbeitet wie ein Pinsel oder Stift, verwendet aber die Farben aus einem im Bild definierten „Quellbereich“.



Beim Klonen entfernt man Bildteile, indem man sie mit Material aus der Umgebung übermalt. 

Die Verwendung dieses Werkzeugs besteht immer aus zwei Teilen. Nach der Vorbereitung des Werkzeugs durch Einstellen der Optionen wählt man

1. zunächst den Quellbereich im Bild aus. Ein Mausklick bei gedrückter **[Strg]**-Taste macht dies. Aus dem so definierten Bereich holt sich GIMP die Farbe, mit der im nächsten Schritt dann
2. das Malen (durch Mausklicks ohne zusätzliche Taste) erfolgt.

Aus naheliegenden Gründen sollte man

- als Pinselspitze eine weiche Spitze verwenden,
- den Quellbereich so wählen, dass die Farbe möglichst gut zum Zielgebiet passt.

Um sich wiederholende Muster zu vermeiden, kann man mit der Option „Zittern hinzufügen“ für ein gewisses Maß an Zufälligkeit sorgen. Ansonsten muss die Deckkraft des Pinsels auch nicht unbedingt 100% betragen. Manchmal ist es besser, mehrfach mit geringer Deckkraft und unterschiedlichen Quellbereichen zu arbeiten.

**Hinweis:** Wie üblich lassen sich einzelne mit dem Malwerkzeug erzeugte Striche schrittweise durch **[Strg] + [Z]** rückgängig machen. Es ist daher sinnvoll, mit kurzen Strichen zu arbeiten und nicht mit einem langen Strich.

**Tipp 56: Die Ausrichtung bei den Klonwerkzeugen einstellen**

Es gibt eine weitere wichtige Einstellung, die eigentlich ganz einfach zu verstehen ist: „Ausrichtung“.

Diese Option legt fest, auf welche Weise die Quelle dem Pinsel (Ziel) folgt. Vier Möglichkeiten gibt es dabei:

1. „Kein“: Das bedeutet, dass GIMP bei jedem Pinselstrich die Farbe wieder exakt an der gleichen Stelle aufnimmt, die zuvor durch **[Strg]**+Mausklick definiert wurde.
2. „Ausgerichtet“: In diesem Modus folgt die Quelle immer im gleichen Abstand dem Pinsel.
3. „Registriert“: Erlaubt die Quelle auf einer anderen Ebene als das Ziel zu definieren.
4. „Fest“: Die Farbe wird immer vom gleichen, anfangs definierten Punkt genommen.

Jede dieser Einstellungen hat ihre Berechtigung und lässt sich mit den anderen Optionen – etwa „Zittern hinzufügen“ – kombinieren.

Weniger oft werden die unter „Quelle“ zur Verfügung stehenden Möglichkeiten genutzt. Hierüber wird entschieden, ob man – wie voreingestellt – das Bild als (Farb-) Quelle verwendet oder ein zuvor ausgewähltes Muster.

**Tipp 57: Das Heilen verwenden**

Das „Heilen“-Werkzeug (Taste **[H]**) ist eine Variante des Klonwerkzeugs. Es verfügt über fast die gleichen Optionen und wird analog angewendet. Der Unterschied zwischen beiden Werkzeugen besteht in der Wirkung. Während das Klonwerkzeug – symbolisiert durch einen Stempel – aufgenommenes Material unverändert zum Malen verwendet, analysiert das Werkzeug Heilen



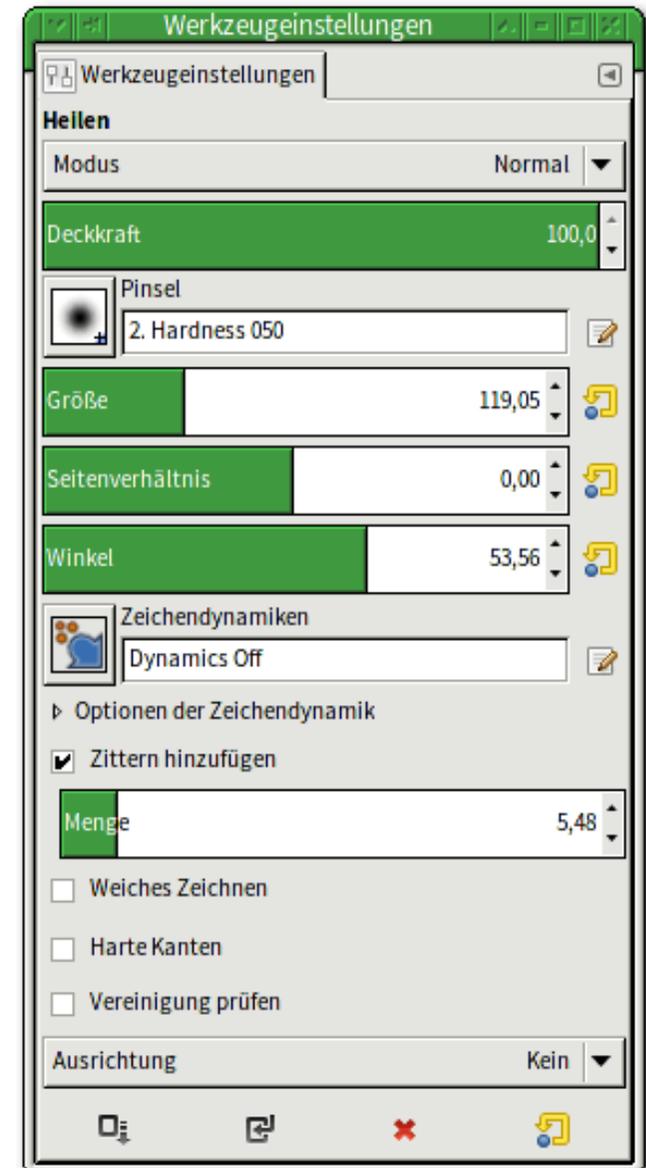
Das Kinn wurde nur mit dem Klon- und dem Heilen-Werkzeug gezeichnet. 🔍

zunächst das Zielgebiet. An die dort herrschenden Bedingungen (Helligkeiten) passt es das aufgenommene Material an, bevor es dieses in das Bild zeichnet. Damit klebt es quasi ein weiches, kleines Pflaster auf die zu reparierende Stelle.

Das Heilen benötigt deutlich mehr Rechenzeit als das Klonen und wird daher meistens für kleinere Korrekturen verwendet. Dann zeigt es allerdings sehr gute Resultate. Als Pinselspitze wird auch hier eine weiche Spitze verwendet, die etwas größer als das zu heilende Gebiet sein sollte. In diesem Fall reichen oft ein oder zwei Mausklicks aus, um den Bereich zu heilen.

Bei großen Bereichen ist diese Methode nicht mehr praktikabel. Zum einen ist die Größe der Pinselspitzen begrenzt, zum anderen reduziert sich die Arbeitsgeschwindigkeit bei großen Pinseln derart, dass ein flüssiges Arbeiten unmöglich ist. Bei solchen Problemen kann man

- zunächst mit dem Klonwerkzeug eine grobe Korrektur vornehmen und anschließend mit dem Heilen versuchen, kleinere Probleme zu beseitigen
- vom Rand nach innen arbeiten,
- oder auf das Resynthesize-Werkzeug ausweichen.



Heilen funktioniert im Wesentlichen wie das Klonen und bietet auch fast die gleichen Optionen. 🔍



Flecken zu entfernen. Hierzu sollte man eine weiche Pinselspitze verwenden, die etwas größer als der störende Fleck ist. Für das Entfernen sind Mausklicks besser als lang gezogene Striche.

### Tipp 58: Das Plugin „Resynthesize“ und „Heal Selection“ nutzen

Ein GIMP-Plug-in mit dem Namen Resynthesize erlaubt, die Arbeit des Heilen-Werkzeugs weitgehend zu automatisieren. Das Plug-in muss zunächst nachgerüstet werden; anschließend wird GIMP neu gestartet. Unter Linux kann man Plug-in direkt aus den Repository der Distribution installieren (z. B. das Paket **gimp-plugin-registry** unter Ubuntu), für Windows und Mac gibt es fertige Programme und weitere Informationen auf der Resynthesize-Webseite [2].

Beim Heilen berücksichtigt GIMP die Helligkeiten im Zielgebiet, was oft besonders unauffällige Korrekturen erlaubt. 🔍

Die besonders wichtigen Optionen dieses Werkzeugs sind:

- die „Deckkraft“, um die Stärke des Effekts zu verringern,
- die Pinselspitze, um die Ränder möglichst unauffällig zu gestalten,
- die Größe der Pinselspitze,
- „Zittern hinzufügen“, um repetierende Muster zu verhindern,
- die „Ausrichtung“.

Alle Optionen stellt man in den Werkzeugeinstellungen ein.

**Hinweis:** Heilen bietet eine gute Möglichkeit, die durch Schmutz auf dem Objektiv erzeugten



Flecken auf dem Objektiv fallen auch bei ausdrucksstarkem Himmel sehr unangenehm auf. Heilen entfernt sie schnell und vollständig. 🔍

Es gibt zwei Möglichkeiten, diese Software zu verwenden:

1. Meistens wird man vermutlich auf das Script „Filter → Verbessern → Heal selection...“ zurückgreifen. Manchmal wird das Script auch als „Smart Remove Selection“ installiert.
2. Alternativ kann man das Plug-in über „Filter → Abbilden → Resynthesize...“ aufrufen.

Das Heal-Selection-Script benötigt nur eine zuvor angelegte Auswahl und präsentiert einen extrem einfachen Dialog.

Die Auswahl wird automatisch mit dem Material aus der Umgebung der Auswahl so gefüllt, dass es möglichst wenig auffällt. „Radius to take texture from“ (manchmal auch „Context sampling width (pixels)“ erlaubt dem Anwender, den Bereich einzugrenzen, aus dem das Material um die Auswahl entnommen werden kann.



Das Script „Heal Selection“ benötigt als einzige Angabe die zulässige Größe des Bereichs um die Auswahl, aus der das Material entnommen werden darf. 

**Hinweis:** In vielen Fällen erzeugt dieses Script wirklich gute Ergebnisse, die oft besser sind als das, was weniger geübte Anwender manuell erzeugen. Falls das Ergebnis nicht gefällt, hat man mehrere Möglichkeiten weiter vorzugehen:

- Man kann den Schritt durch **[Strg] + [Z]** rückgängig machen und es mit einer neuen Auswahl erneut versuchen.
- Das Script mit der bestehenden Auswahl erneut (durch **[Strg] + [F]**) aufrufen.
- Das Script mit einer neuen Auswahl erneut (durch **[Strg] + [F]**) aufrufen. Oft verbessern sich die Ergebnisse, wenn man die Auswahl um einen kleinen Betrag (etwa 20 Pixel) vergrößert.

- Manchmal hilft es, wenn man den zu bearbeitenden Bereich einschränkt. Statt alle vier Ecken eines Panoramabildes in einem Schritt zu bearbeiten, versucht man besser die Ecken einzeln zu füllen.

**Hinweis:** „Heal Selection“ funktioniert auch in besonders „schwierigen“ Bereichen von Bildern, etwa in den Ecken und an Kanten. Bei einem gleichmäßigen Hinter- oder



Das Plug-in füllt die beim Zusammensetzen eines Panoramas entstandenen Ecken auf. 

Untergrund kommt allerdings auch dieses Script an seine Grenzen und hinterlässt manchmal sichtbare Artefakte.

Das Resynthesize-Plug-in selbst verfügt ebenfalls über eine Oberfläche. Man ruft diese über „Filter → Abbilden → Resynthesize...“ auf. Die dann offenbaren Optionen sind wenig intuitiv und erfordern einige Übung, um bessere Ergebnisse als die mit dem Heal-Selection-Script erzeugten zu generieren. Daher wird hier nicht weiter auf das Plug-in eingegangen.



G'MICs „Patch based Inpaint“ wirkt ähnlich wie „Resynthesize“. 

### Tipp 59: G'MICs-Filter „Patch based Inpaint“ kennen

Besonders für das Entfernen kleiner Bildteile, etwa ein Verkehrsschild in einem Landschaftsbild oder einem Auto usw., wurde der G'MIC-Filter „Patch based Inpaint“ entwickelt. G'MIC selbst kann man über die Webseite [3] oder die Linux-Paketquellen (Paket **gimp-gmic**) installieren.

Dieser Filter ähnelt in vielerlei Hinsicht dem Resynthesizer, allerdings weicht die Anwendung ab:

1. Zuerst lädt man das gewünschte Bild.
2. Dann dupliziert man die Ebene und malt auf der neuen mit einem harten Pinsel die zu entfernenden Bereiche an. Hierzu sollte man die im Filterdialog angegebene Farbe verwenden. Voreingestellt ist ein leuchtendes Rot.

3. Danach ruft den G'MIC-Filter über „Filter → G'MIC... → Repair → Inpaint [patch-based]“ auf.
4. Zum Schluss wählt man einen geeigneten „Ausgabemodus“ auf der linken Seite, etwa „Neue Ebene(n)“.

Ein Klick auf „OK“ schließt den Dialog und berechnet das Ergebnis. „Anwenden“ macht dies, ohne den Dialog zu schließen.

**Hinweis:** Es gibt unter „Mask Type“ eine alternative Methode um festzulegen, welche Bereiche ersetzt werden sollen.

Die Ergebnisse bei diesen Filter variieren stark, abhängig vom Bild und den gewählten Einstellungen:

- „Patch size“ definiert die Größe der zum Füllen verwendeten Flicker.
- „Lookup size“ stellt die Größe des Umfeldes ein, aus dem sie stammen.
- „Lookup factor“ beschreibt die Anzahl der Übermalungen.
- Die „Blend“-Parameter steuern das Auftragen der Flicker im Bild.

Insgesamt kann dieser Filter schnell zu guten Ergebnissen führen, sofern eine gewisse Erfahrung mit den Einstellungen besteht.

**Achtung:** Im Unterschied zu den Klonwerkzeugen benötigt dieser Filter eine Maske mit harten Kanten. Als Malwerkzeug empfiehlt sich daher ein Stift mit harter Spitze.

#### LINKS

- [1] [http://www.bookrix.de/\\_ebook-wilber-gimper-bildbearbeitung-mit-gimp-die-101-wichtigsten-tipps/](http://www.bookrix.de/_ebook-wilber-gimper-bildbearbeitung-mit-gimp-die-101-wichtigsten-tipps/)
- [2] <http://registry.gimp.org/node/25219> 
- [3] <http://gmic.sourceforge.net/> 

#### Autoreninformation

Die **GIMPer** sind eine wechselnde Gruppe von GIMP-Freunden und -Nutzern, die das Programm für ihre Kunstprojekte, Arbeit oder Hobby einsetzen.

 [Teilen](#)

 [Kommentieren](#)

## Git Tutorium – Teil 2 von Sujeevan Vijayakumaran

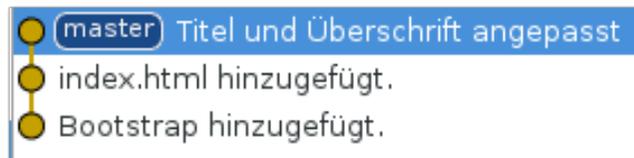
Im ersten Teil des Git Tutoriums (siehe freiesMagazin 12/2014 [1]) wurden die ersten Schritte mit Git getätigt: Zunächst das Anlegen eines Repositories, dann das Hinzufügen und Committen von Dateien und das Anschauen des Logs. Im zweiten Teil wird nur ein Thema behandelt und zwar das Branching-Modell von Git.

### Allgemeines zum Branching

Ein wichtiges Element von Git und auch anderen Versionsverwaltungsprogrammen ist das Branching. Das Wort „Branch“ lässt sich in diesem Fall am Besten mit „Zweig“ übersetzen. Es ist möglich, den aktuellen Entwicklungsstand „abzuzweigen“ und daran weiter zu entwickeln. Konkret bedeutet dies, dass quasi eine Kopie vom aktuellen Arbeitsstand erzeugt wird und man dort weitere Commits tätigen kann, ohne dabei die Hauptentwicklungslinie zu berühren. Die Nutzung von Branches ist eine zentrale Eigenschaft von Git, insbesondere in der Software-Entwicklung. In der Praxis sieht das dann meistens so aus, dass einzelne Features in einzelnen Branches entwickelt werden und dann nach und nach in den Haupt-Entwicklungszweig gemergt werden. Häufig ist es allerdings so, dass man auch noch ältere Versionen pflegt, die etwa noch mit Sicherheitsaktualisierungen versorgt werden müssen. So kann man recht einfach von einem Entwicklungszweig auf einen anderen Branch wechseln und dort noch schnell einen Fehler korrigieren.

Anschließend kann man wieder zurück wechseln und an seinem Feature weiterarbeiten. Das ganze Vorgehen hilft den Programmierern zwischen verschiedenen Versionen und Entwicklungslinien zu springen, ohne großen Aufwand betreiben zu müssen.

Im ersten Teil des Tutoriums wurden bereits drei Commits getätigt. Da kein spezieller Branch angegeben worden ist, geschah dies automatisch auf dem Master-Branch. Der Master-Branch ist der Haupt-Zweig, der in vielen Git-Repositories existiert. Dieser wird automatisch angelegt, wenn man in einem leeren Git-Repository den ersten Commit tätigt.



Die aktuellen Commits auf dem Master-Branch. 🔍

Die ersten drei Commits wurden, wie oben bereits geschrieben, auf den Branch **master** übertragen. Die Entwicklung verlief bislang geradlinig, sodass keine Abzweigung erstellt wurde.

Beim Arbeiten mit Git bietet es sich je nach Entwicklungsart häufig an, für jedes Feature, welches man implementieren möchte, einen eigenen Branch zu erstellen. Insbesondere deshalb, da oft

Features zeitgleich von verschiedenen Entwicklern implementiert werden.

### Branches anlegen

Die Beispiel-Webseite besitzt aktuell lediglich eine simple Überschrift. Was fehlt, wäre zum einen ein Inhalt und zum anderen ein kleines Menü. Für beides sollen eigene Branches erstellt werden.

Um sicherzustellen, dass man auf dem richtigen Branch ist, kann man folgenden Befehl ausführen:

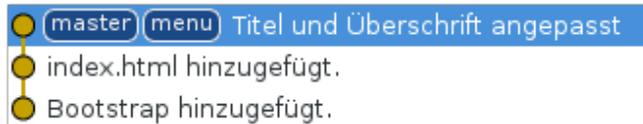
```
$ git branch
* master
```

Da nur ein Branch aktuell vorhanden ist, wird auch nur der Branch **master** angezeigt. Das \* vor dem Branchnamen signalisiert, dass man sich gerade auf dem Branch befindet.

```
$ git branch menu
```

Der oben aufgeführte Befehl erzeugt den neuen Branch **menu**. Wenn man einen Branch mit dem **git branch** Befehl erzeugt, wird der Branch zwar angelegt, aber man wechselt nicht automatisch auf diesen Branch. Dies macht ein erneutes Ausführen von **git branch** deutlich:

```
$ git branch
* master
  menu
```



Der Branch **menu** ist noch identisch mit **master**. 🔍

Jetzt werden beide vorhandenen Branches angezeigt. Man befindet sich allerdings immer noch auf dem **master**-Branch. Zum Wechseln des Branches nutzt man den Befehl **git checkout**.

```
$ git checkout menu
Gewechselt zu Branch 'menu'
```

Beim häufigen Erzeugen und Wechseln zu einem Branch wären die obigen Befehle auf Dauer zu lästig, weil man häufig sofort auf dem neu erstellten Branch wechseln will. Dafür gibt es den kombinierten Befehl:

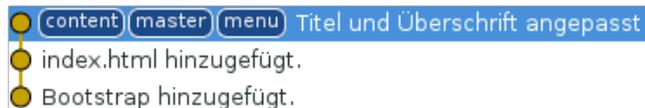
```
$ git checkout -b menu
Gewechselt zu einem neuem Branch 'menu'
```

Dieser Befehl legt nicht nur den Branch **menu** neu an, sondern wechselt auch direkt auf diesen Branch. Es ist wichtig zu wissen, auf welchem Branch man sich befindet, wenn man den neuen Branch anlegt. Dies ist zwar in diesem Beispiel irrelevant, da nur ein Branch existiert, man sollte es aber stets beachten.

Als Basis des neuen Branch wird der aktuelle Commit des aktuellen Branches genommen.

Wenn man sich also auf dem Branch **menu** befindet und den Branch **content** erstellt, dann nimmt er als Basis den aktuellsten Commit von **menu** und nicht **master**. Um das Beispiel fortzuführen, muss daher der Branch **content** erzeugt werden.

```
$ git checkout -b content
Gewechselt zu Branch 'content'
```



Drei Branches existieren, die auf denselben Commit zeigen. 🔍

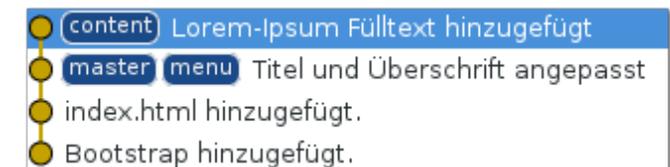
Zum aktuellen Zeitpunkt existieren drei Branches. Alle fußen auf demselben Commit. In diesem Branch wird nun ein kleiner Inhalt hinzugefügt, dazu reicht es, den Lorem-Ipsum Generator [2] zu nutzen, um einen Fülltext zu erzeugen.

Unterhalb der **<h1>** Überschrift in der Datei **index.html** sollte folgendes eingefügt werden:

```
<p>
  Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod ~
  tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At ~
  vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, ~
  no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit ~
  amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut ~
  labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam ~
  et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata ~
  sanctus est Lorem ipsum dolor sit amet.
</p>
```

Diese Änderung muss dann aus dem Arbeitsverzeichnis heraus wie gewohnt übertragen werden:

```
$ git add index.html
$ git commit -m "Lorem-Ipsum Fülltext ~
hinzugefügt"
[content 395dd48] Lorem-Ipsum Fülltext ~
hinzugefügt
1 file changed, 3 insertions(+)
```



Durch den neuen Commit ist **content** ein Commit über **master**. 🔍

Es lohnt sich, das Log mit dem Befehl **git log** anzusehen. Auf dem aktuellen Branch **content** sind vier Commits vorhanden. Es sind sowohl die ersten drei Commits vor dem Abzweigen vorhanden, als auch der zuletzt hinzugefügte Commit.

Wechselt man mit **git checkout master** zurück auf **master** und schaut das Log an, dann sind dort nur drei Commits vorhanden. Dies liegt daran, dass Git den Commit nur auf **content** ausgeführt hat und nicht auf **master**. Die Änderungen aus **content** können in **master** übernommen werden. Dieser Schritt folgt hier jedoch noch nicht.

Es gilt noch die ein oder andere Änderung im Branch **menu** durchzuführen. Dazu muss man wieder auf den Branch **menu** wechseln:

```
$ git checkout menu
Gewechselt zu Branch 'menu'
```

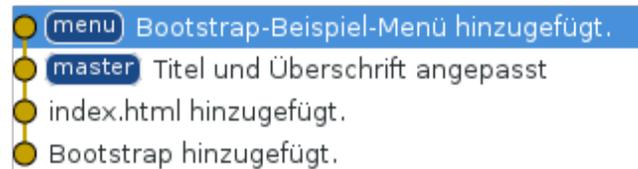
Wenn man nun **index.html** zum Bearbeiten öffnet, sind die Änderungen nicht enthalten. Das macht auch Sinn, da die Änderungen auf dem Branch **content** durchgeführt wurden.

Die **index.html**-Datei bekommt nun ein Menü spendiert. Hierfür muss folgender Code vor der **<h1>**-Überschrift hinzugefügt werden:

```
<nav class="navbar navbar-default" role="navigation">
  <div class="container-fluid">
    <div class="collapse navbar-collapse" id="bs-example-
      navbar-collapse-1">
      <ul class="nav navbar-nav">
        <li class="active"><a href="#">Link</a></li>
        <li><a href="#">Link</a></li>
      </ul>
    </div>
  </div>
</nav>
```

Diese Änderung kann dann ebenfalls wie gewohnt übertragen werden:

```
$ git add index.html
$ git commit -m "Bootstrap-Beispiel-Menü
hinzugefügt"
```



Auch **menu** ist ein Commit über **master**. 🔍

Jetzt fällt aber auf, dass zwar ein Menü vorhanden ist, in beiden Menüpunkten steht allerdings nur „Link“. Der Einfachheit halber reicht es, wenn man an dieser Stelle den ersten „Link“ mit „Home“ und den zweiten „Link“ mit „About“ ersetzt. Diese Änderung muss dann ebenfalls übertragen werden.

## Branches mergen

Bis jetzt wurden einige Arbeiten am Repository durchgeführt. Dieser Abschnitt soll noch kurz zusammenfassen, was alles geschah. Zunächst wurden zwei neue Branches mit den Namen **content** und **menu** erstellt. Beide basieren auf dem Branch **master**. Im An-

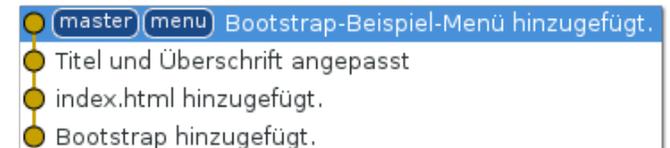
schluss wurde dann ein Commit in **content** und zwei Commits in **menu** erzeugt.

Diese Änderungen können nun zusammengeführt werden. Dafür existiert der Befehl **git merge**. Dieser Befehl muss dort ausgeführt werden, wohin die Änderungen aus dem anderen Branch eingefügt werden sollen. In diesem Beispiel sollen die Änderungen aus den Branches **content** und **menu** in **master** übernommen werden. Dazu muss man auf den Branch "master" wechseln:

```
$ git checkout master
```

Anschließend kann der erste Branch gemerged werden.

```
$ git merge menu
Aktualisiere 24e65af..c3cf413
Fast-forward
 index.html | 10 ++++++++
 1 file changed, 10 insertions(+)
```



Nach dem Merge sind die beiden Branches **menu** und **master** identisch. 🔍

Git führt an dieser Stelle einen sogenannten „fast-forward merge“ durch. Dies geschieht immer genau dann, wenn seit dem Abzweigen des Branches auf dem ursprünglichen Branch keine Ände-

rungen geschehen sind. Das ist genau bei diesem Merge der Fall. Anders sieht es hingegen aus, wenn man den Branch **content** nach **master** mergen möchte.

```
$ git merge content
```

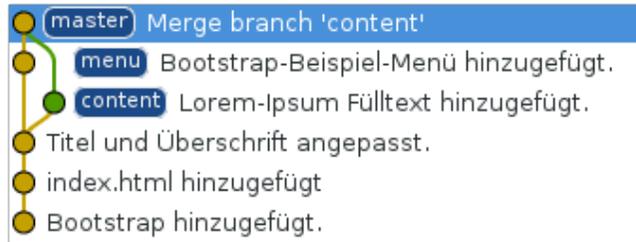
Der Befehl öffnet den in Git konfigurierten Editor, etwa vim, mit folgendem Inhalt:

```
Merge branch 'content'

# Bitte geben Sie eine Commit-
# Beschreibung ein um zu erklären, warum
# dieser
# Merge erforderlich ist, insbesondere
# wenn es einen aktualisierten
# Upstream-Branch mit einem Thema-Branch
# zusammenführt.
#
# Zeilen beginnend mit '#' werden
# ignoriert, und eine leere Beschreibung
# bricht den Commit ab.
```

In der Regel belässt man den Commit-Text bei dem vorgegebenen Inhalt. Gegebenenfalls kann man allerdings, wie die Nachricht bereits aussagt, einen Grund angeben, warum der Merge nötig war. Als Ausgabe erscheint nach dem Abspeichern dann folgendes:

```
automatischer Merge von index.html
Merge made by the 'recursive' strategy.
index.html | 3 +++
1 file changed, 3 insertions(+)
```



*Der Recursive-Merge benötigt einen Merge-Commit.*

Im Gegensatz zum ersten Merge war hier ein „recursive merge“ notwendig. Dies geschieht zwar in diesem Fall auch vollkommen automatisch, die Commit-Historie sieht allerdings anders aus. Dies hängt damit zusammen, dass durch das Mergen vom Branch **menu** nun Änderungen auf dem Branch **master** passiert sind, seitdem der Branch **content** abgezweigt wurde. Die beiden Branches sind dadurch divergiert. Das heißt, die beiden Commits auf dem Branch **content** fußen nicht direkt auf dem neuen Commit aus **content**, welches in **master** überführt worden ist.

Wenn man nun das Git Log anschaut, dann sind mittlerweile alle Commits aus allen Branches in **master** enthalten. Zusätzlich wurde durch den letzten Merge ein weiterer Merge-Commit hinzugefügt.

### Merge-Konflikte

Mergen von Branches ist nicht immer ganz einfach. Git selbst verfolgt verschiedenen Strategien, um Branches zu mergen. Das klappt bei einigen kleineren Änderungen zwar ohne Proble-

me. Wenn allerdings größere Änderungen in den Branches stattgefunden haben, passiert es häufig, dass dann Merge-Konflikte auftreten. Merge-Konflikte sind Probleme, die auftreten, wenn der gleiche Code-Abschnitt von beiden Branches verändert wurde. Darunter fällt auch, wenn Zeilen auf einem Branch gelöscht worden sind, aber auf dem anderen noch vorhanden sind.

Das Verhalten lässt sich auch ganz einfach nachbilden. Zunächst wechselt man zurück auf den Branch **master**, falls man sich noch nicht drauf befindet.

```
$ git checkout master
Bereits auf 'master'
```

Anschließend erzeugt man einen neuen Branch:

```
$ git checkout -b titel
```

Auf diesem Branch ändert man anschließend den Titel in der **<h1>**-Überschrift von „Hallo Git!“ in „Hallo Merge-Konflikt!“. Nach dem Abspeichern kann man die Datei wieder wie gewohnt zum Index hinzufügen und schlussendlich committen:

```
$ git add index.html
$ git commit -m "Titel für den Merge-
Konflikt"
[titel 420e0ae] Titel für den Merge-
Konflikt
1 file changed, 1 insertion(+), 1
deletion(-)
```

Anschließend geht es zurück auf **master**.

```
$ git checkout master
```

Dort ändert man den Titel in der **<h1>**-Überschrift von „Hallo Git!“ auf „Hallo!“. Auch hier überträgt man die Änderungen.

```
$ git add index.html
$ git commit -m "Neuer Titel"
[master 9cb085b] Neuer Titel
1 file changed, 1 insertion(+), 1
deletion(-)
```

Die Voraussetzung für einen simplen Merge-Konflikt wurden somit geschaffen. Wenn man nun die beiden Branches **master** und **titel** mergen möchte, geschieht folgendes:

```
$ git merge titel
automatischer Merge von index.html
KONFLIKT (Inhalt): Merge-Konflikt in
index.html
Automatischer Merge fehlgeschlagen;
beheben Sie die Konflikte und committen
Sie dann das Ergebnis.
```

Wie gewünscht trat der Merge-Konflikt auf. Bevor man hinget und versucht den Konflikt zu beheben, lohnt sich ein Blick auf die Ausgabe von **git status**:

```
$ git status
Auf Branch master
Sie haben nicht zusammengeführte Pfade.
```

```
(beheben Sie die Konflikte und führen
Sie "git commit" aus)
```

Nicht zusammengeführte Pfade:

```
(benutzen Sie "git add/rm <Datei>..."
um die Auflösung zu markieren)
```

von beiden geändert: index.html

```
keine Änderungen zum Commit vorgemerkt
benutzen Sie "git add" und/oder "git
commit -a")
```

Der Befehl **git status** gibt bei fehlgeschlagenen automatischen Merges immer die Information aus, dass Dateien vorhanden sind, die noch zusammengeführt werden müssen.

Schaut man sich nun die Datei **index.html** an, dann findet man dort folgende Zeilen:

```
<<<<<<< HEAD
  <h1>Hallo!</h1>
=====
  <h1>Hallo Merge-Konflikt!</h1>
>>>>>>> titel
```

Der Merge-Konflikt wird direkt in der Quell-Datei eingefügt. Git nutzt Marker um aufzuzeigen, welcher Teil des Codes aus welchem Branch bzw. Commit kommt. In der ersten Zeile des Konflikt ist der Marker folgender: **<<<<<<< HEAD**. **HEAD** ist ein Zeiger auf den aktuellen Commit auf dem Branch, auf dem man sich vor dem Merge be-

fand. **HEAD** gibt es nicht nur bei Merge-Konflikten, sondern auch an allen anderen Stellen in einem Git-Repository. In diesem Fall ist das der letzte Commit im Branch **master**. Getrennt wird dies durch den weiteren Marker **=====**. Alles was zwischen **<<<<<<< HEAD** und **=====** befindet, stammt vom aktuellen Branch ab. Der zweite Teil nutzt ebenfalls **=====** als Trennzeichen und endet mit **>>>>>>> titel**. In diesem Teil sind alle Änderungen aus dem Branch **titel** enthalten.

Der Konflikt kann nun relativ einfach aufgelöst werden. Es müssen alle Marker entfernt und nur der gewünschte Teil eingefügt werden. In diesem Falle ist gewollt die Änderungen aus dem **HEAD** beizubehalten, weshalb man Zeile 1 und Zeile 3-5 löschen kann. Im Anschluss muss man **index.html** wieder dem Index hinzufügen.

```
$ git add index.html
```

Wenn man nun erneut **git status** ausführt, dann meldet Git, dass die Konflikte behoben worden sind.

```
$ git status
Auf Branch master
Alle Konflikte sind behoben, aber Sie
sind immer noch beim Merge.
(benutzen Sie "git commit" um den
Merge abzuschließen)

nichts zu committen, Arbeitsverzeichnis
unverändert
```

Beim Ausführen von **git commit** öffnet sich der Editor mit folgender Commit-Nachricht:

```
$ git commit
Merge branch 'titel'

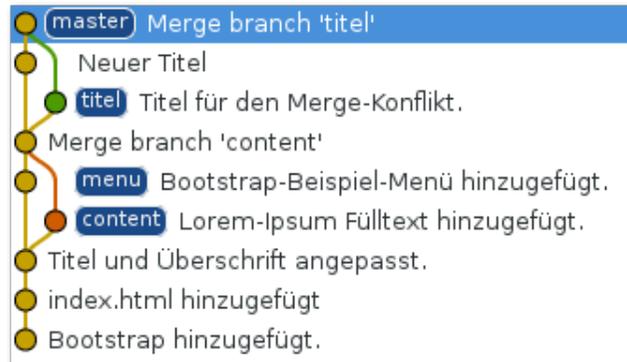
Conflicts:
  index.html
#
# Es sieht so aus, als committen Sie ~
# einen Merge.
# Falls das nicht korrekt ist, löschen ~
# Sie bitte die Datei
#   .git/MERGE_HEAD
# und versuchen Sie es erneut.

# Bitte geben Sie eine Commit-~
# Beschreibung für Ihre Änderungen ein. ~
# Zeilen,
# die mit '#' beginnen, werden ignoriert~
# , und eine leere Beschreibung
# bricht den Commit ab.
# Auf Branch master
# Alle Konflikte sind behoben, aber Sie ~
# sind immer noch beim Merge.
```

Hier kann man die Commit-Nachricht verändern. Dann ist der Merge-Konflikt erfolgreich behoben.

Zum Schluss können nicht mehr benötigten Branches aufgeräumt werden. Der Befehl **git branch** kennt hierfür den Parameter **-d** für „delete“:

```
$ git branch -d titel
Branch titel entfernt (war 420e0ae).
```



Der Graph nach dem letzten Merge. 🔍

### Ausblick

Der nächste Teil rundet den Einstieg in Git ab. Thematisiert wird zum einen wie man mit Remote-Repositories arbeitet und zum anderen wie man Branches „rebased“.

### LINKS

- [1] <http://www.freiesmagazin.de/freiesMagazin-2014-12>
- [2] <http://www.loremipsum.de/>

### Autoreninformation

**Sujeewan Vijayakumaran** ([Webseite](#)) setzt seit drei Jahren Git zur Versionsverwaltung ein. Dabei nutzt er es nicht nur zur Software-Entwicklung, sondern auch für das Schreiben von Artikeln.



Teilen



Kommentieren



## Spacewalk – Teil 4: Verwaltung von Solaris-Systemen von Christian Stankowic

**F**okus des dritten Teils dieser Artikel-Serie war die Automatisierung von Administrationsaufgaben und das Provisionieren neuer Systeme (freiesMagazin 11/2014 [1]). Dieser Teil beschäftigt sich ganz mit der Unterstützung von Solaris-basierenden Systemen.

### Spacewalk „UNIX-Support“

Spacewalk verfügt über einen optionalen „UNIX-Support“, mit dessen Hilfe UNIX-Systeme ähnlich wie Linux-Systeme verwaltet werden können. Insbesondere für Migrationsprojekte konzipiert, wurde dieses Feature in Spacewalk und den kommerziellen Red Hat Satellite-Server aufgenommen – SUSE hat dieses Feature in seinem Alternativprodukt SUSE Manager vollständig entfernt. In der Red Hat Satellite-Dokumentation [2] ist zwar von generellem „UNIX-Support“ die Rede, unterstützt wird jedoch lediglich Solaris. Andere bekannte proprietäre UNIX-Derivate, wie z. B. IBM AIX oder HP-UX, wurden nie unterstützt.

Gemäß Dokumentation werden die SUN/Oracle Solaris-Releases 8 bis 10 (Architekturen x86 und SPARC) unterstützt. Inoffiziell funktionieren in der Regel darüber hinaus jedoch auch:

1. Oracle Solaris 11
2. OpenSolaris/OpenIndiana
3. theoretisch auch alle anderen auf Illumos basierenden Derivate

Seit Spacewalk 2.2 ist diese Funktion jedoch „deprecated“ – das Projekt behält sich somit die Option vor, sie in zukünftigen Versionen zu entfernen. Die Funktion wird in letzter Zeit immer weniger gepflegt, was auch mit der allgemein eher stagnierenden Solaris-Marktentwicklung zusammenhängt.

### Einschränkungen

Gegenüber verwalteten Linux-Systemen müssen auch einige Einschränkungen beachtet werden. Die auffälligste Einschränkung ist, dass Solaris-Pakete nicht über Netzwerk-Spiegel importiert werden können. Die Architektur der Solaris Netzwerk-Spiegel unterscheidet sich stark von der gängiger Linux-Distributionen – eine entsprechende Integration ist in Spacewalk derzeit nicht gegeben.

Bevor Software-Pakete für Solaris verteilt werden können, müssen sie zunächst in MPM-Dateien konvertiert und auf den Spacewalk-Server hochgeladen werden. Auch hinsichtlich der Aufgabenplanung müssen Abstriche gemacht werden – unter Solaris steht das Programm osad nicht zur Verfügung. Somit können Aufgaben nicht in Echtzeit eingeplant werden, eine periodische Abfrage seitens der registrierten Systeme ist notwendig. Remote-Kommandos und im Spacewalk-Profil hinterlegte Hardware-Informationen sind bei manchen Architekturen und Solaris-Versionen leider fehlerhaft. Solaris-Systeme können nicht interak-

tiv registriert werden – es ist notwendig, einen Aktivierungsschlüssel zu erstellen und das System mittels `rhnreg_ks` zu registrieren.

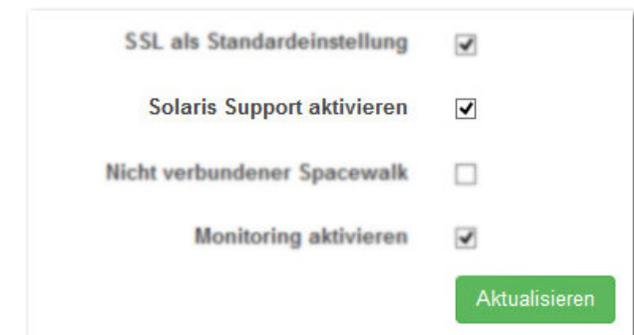
Solaris-Systeme lassen sich demnach lediglich rudimentär zentral verwalten, was für Migrationszwecke jedoch ausreichen dürfte.

### Verwaltung von Solaris-Systemen

#### Vorbereitung von Spacewalk

Bevor Solaris-Systeme mit Spacewalk verwaltet werden können, müssen der UNIX-Support aktiviert und die Spacewalk-Dienste neu gestartet werden. Die Unterstützung kann in der Web-Oberfläche unterhalb des Menüs „Admin → Spacewalk Konfiguration“ aktiviert werden. Nach dem Übernehmen der Änderungen erfolgt der Neustart von Spacewalk über die Registerkarte „Neustart“ oder die Kommandozeile:

```
# spacewalk-service restart
```



Einstellung „Solaris Support aktivieren“.

Anschließend muss ein Software-Basiskanal für Solaris erstellt werden; für zusätzliche (Drittanbieter-)Anwendungen können im Anschluss weitere Unterkanäle erstellt werden. Die Erstellung der Kanäle erfolgt über die Web-Oberfläche unterhalb des Menüs „Channels → Software-Channels verwalten → Neuen Channel erstellen“. Im daraufhin folgenden Dialog werden die folgenden Einstellungen vorgenommen:

1. Channel-Name: z. B. „Solaris 11“
2. Channel-Label: z. B. „solaris-11“
3. Parent-Channel: keine
4. Architektur: „i386 Solaris“ oder „Sparc Solaris“
5. Channel-Zusammenfassung: z. B. „Solaris 11-Pakete“

Da Solaris-Systeme nicht interaktiv registriert werden können, muss ein Aktivierungsschlüssel über „Systeme → Aktivierungs-Schlüssel → Neuen Schlüssel erstellen“ erstellt werden.

1. Beschreibung: z. B. „Solaris11-Key“
2. Basis-Channels: zuvor erstellter Kanal

Es ist auch wichtig, im Formular die Provisioning-Zusatzberechtigung auszuwählen. Somit ist sichergestellt, dass registrierte Solaris-Systeme über das zentrale Konfigurationsmanagement verwaltet werden können.

### Vorbereitung der Solaris-Systeme

Bevor Solaris-Systeme mit Spacewalk verwaltet werden können, müssen die für das verwenden-

de Release konzipierten OpenSSL- und ZIP-Bibliotheken und die GCC-Runtime installiert werden. Oftmals sind diese Bibliotheken bereits vorinstalliert – andernfalls hilft ein Blick in die offiziellen Installationsmedien:

```
# pkginfo | egrep -i "zlib|openssl|gccruntime"
system    SUNWgccruntime    GCC Runtime libraries
system    SUNWopenssl       OpenSSL Libraries (Root)
system    SUNWzlib          The Zip compression library
```

Unter OpenIndiana-Derivaten lautet das Paket der GCC-Runtime **gcc-libstdc** und kann komfortabel über das pkg-Kommando wie folgt installiert werden:

```
# pkg install gcc-libstdc
```

Für ältere SUN Solaris-Releases stehen die benötigten Pakete unter Umständen nicht auf den offiziellen Installationsmedien zur Verfügung. In diesem Fall hilft ein Blick auf die Webseite des unabhängigen OpenCSW-Spiegels [3] – hier gibt es auch für alte Solaris-Releases noch zahlreiche Software-Pakete.

Zur Systemverwaltung werden noch einige Python-Tools benötigt, die auf der Spacewalk-Webseite [4] heruntergeladen werden können. Die sogenannten Bootstrap-Pakete sind nach Solaris-Release und Architektur gegliedert; so existieren Pakete für Solaris 8 bis 10, jeweils für die Architekturen SPARC und x86. Für Solaris 11 existiert kein offizieller Tarball, die Verwendung

des Solaris 10-Tarballs funktioniert hier jedoch ebenfalls.

Der Tarball muss auf dem Solaris-System entpackt und die Software-Pakete installiert werden:

```
# gzip -d rhn-solaris-bootstrap*.tar.gz
# tar xf rhn-solaris-bootstrap*.tar
# cd rhn-solaris-bootstrap-*
# for i in *.pkg ; do pkgadd -d $i all; \r
done
```

Anschließend müssen die Pfade für gemeinsame Programmbibliotheken angepasst werden. Dieser Schritt unterscheidet sich zwischen Solaris 10, 11 und OpenIndiana. Unter Solaris 10 benutzt man:

```
# crle -l /lib -l /usr/lib -l /usr/local\r
/lib -l /opt/redhat/rhn/solaris/lib
```

Unter Solaris 11 lautet der Befehl leicht abgewandelt:

```
solaris11 # crle -l /lib -l /usr/lib -l \r
/usr/local/lib -l /usr/srw/lib -l /opt/\r
redhat/rhn/solaris/lib
```

Und schließlich unter OpenIndiana benutzt man den Befehl:

```
# crle -l /lib -l /usr/lib -l /opt/redhat/rhn/solaris/lib
```

In diesem Beispiel wird die bereits definierte Pfadliste um den Pfad **/opt/redhat/rhn/solaris/lib** erweitert. In diesem Ordner befinden sich SSL- und Python-Bibliotheken. Werden die Pfade nicht korrekt angepasst, können die installierten Python-Anwendungen nicht gestartet werden.

Es empfiehlt sich, das eigene oder systemweite Benutzerprofil anzupassen, damit die neu hinzugekommenen RHN-Kommandos auch zur Verfügung stehen. Diese Anpassung ist notwendig, da sich die Programme in einer eigenen Ordnerstruktur befinden – dieser Pfad gehört nicht zu den Standard-Suchpfaden.

```
PATH=$PATH:/opt/redhat/rhn/solaris/bin:/opt/redhat/rhn/solaris/usr/bin:/opt/redhat/
rhn/solaris/usr/sbin
MANPATH=$MANPATH:/opt/redhat/rhn/solaris/man
export PATH
export MANPATH
```

Sollen für das Solaris-System auch Remote-Befehle und das Konfigurationsmanagement aktiviert werden, müssen die entsprechenden Berechtigungen vergeben werden:

```
# rhn-actions-control --enable-run
# rhn-actions-control --enable-deploy
```

Bei älteren Solaris-Versionen steht das Programm **rhn-actions-control** unter Umständen nicht zur Verfügung. Hier müssen ersatzweise Ordner-

und Dateistrukturen erstellt werden, um die benötigten Berechtigungen zu erteilen:

```
# mkdir -p /opt/redhat/rhn/solaris/etc/
sysconfig/rhn/allowed-actions/{script,
configfiles}
# touch /opt/redhat/rhn/solaris/etc/
sysconfig/rhn/allowed-actions/script/run
# touch /opt/redhat/rhn/solaris/etc/
sysconfig/rhn/allowed-actions/
configfiles/all
```

### Registrierung

Vor der Registrierung erfolgt, wie auch bei Linux-Systemen, die Anpassung der

up2date-Konfiguration. Im Wesentlichen muss hier die URL des Spacewalk-Systems und der Pfad zum SSL-Zertifikat angegeben werden. Das SSL-Zertifikat befindet sich im **pub**-Ordner des Management-Systems und muss noch auf das zu verwaltende System übertragen werden – beispielsweise mittels wget:

```
# wget --no-check-certificate https://fqdn-satellite.domain.loc/pub/RHN-ORG-TRUSTED-
SSL-CERT -O /opt/redhat/rhn/solaris/usr/share/rhn/RHN-ORG-TRUSTED-SSL-CERT
# cd /opt/redhat/rhn/solaris/etc/sysconfig/rhn/
```

Im Wesentlichen müssen drei Zeilen der up2date-Konfiguration angepasst werden:

```
noSSLServerURL=http://fqdn-satellite.
domain.loc/XMLRPC
...
serverURL=https://fqdn-satellite.domain.
loc/XMLRPC
...
sslCACert=/opt/redhat/rhn/solaris/usr/
share/rhn/RHN-ORG-TRUSTED-SSL-CERT
```

Die Registrierung des Systems muss über einen Aktivierungsschlüssel erfolgen. Eine interaktive Registrierung ist nicht möglich, da das Programm **rhn\_register** nicht zur Verfügung steht. Zur Registrierung wird das Kommando **rhnreg\_ks** unter Angabe des vorher erstellten Aktivierungsschlüssels ausgeführt:

```
# rhnreg_ks --activationkey=x-xxx
Doing checkNeedUpdate
Updating cache...
#####
[100%]
Updating cache...
#####
[100%]
Package list refresh successful
```

Wie bereits erwähnt, stehen osad und somit Echtzeit-Systemverwaltung unter Systemen, die

Ungruppierte Systeme

0 1 2 3 4 5 6 7 8 9 A B C D E F G H I J K L M N O P Q R S T U V W X Y Z 1 - 2 von 2

Filtern nach Systemname:   25 Elemente pro Seite anzeigen

<input type="checkbox"/> System	Updates	Gesundheit	Errata	Pakete	Konfigurationen	Abstürze	Basis-Channel	Berechtigung
<input type="checkbox"/> tvm-sol10	✓	✓	0	0	0	(keine)	Solaris 10	Management, Provisioning
<input type="checkbox"/> tvm-sol11.localdomain.loc	✓	✓	0	0	0	(keine)	Solaris 11	Management, Provisioning

Alle auswählen

Ausgewählte zum SSM hinzufügen  Vorhandene Server vom SSM entfernen [CSV herunterladen](#) 1 - 2 von 2

Zwei registrierte Solaris-Systeme. 🔍

auf Solaris basieren, nicht zur Verfügung, weswegen rhnspd verwendet werden muss. Das Tool kann mit folgendem Aufruf gestartet werden:

```
# /opt/redhat/rhn/solaris/usr/sbin/rhnspd
--foreground -interval=10 &
```

Dieser Aufruf überprüft alle 10 Minuten, ob Aufgaben anstehen. Je nach Größe der Systemlandschaft ist es ratsam, diesen Wert anzupassen, um unnötigen Netzwerktraffik zu vermeiden.

Damit **rhnspd** automatisch mit dem System startet, muss je nach Solaris-Version ein Init-Skript (bis Solaris 9) oder ein SMF-Manifest (Service Management Facility, ab Solaris 10) erstellt werden. Gegenüber herkömmlichen Init-Skripten stellen

SMF-Manifeste XML-Dokumente dar. Dieses System bietet gegenüber klassischen Init-Systemen einige Vorteile:

1. Parallelisierung von Startprozessen, schnelleres Booten
2. einfachere Definition von Abhängigkeiten zu anderen Diensten
3. automatischer Neustart nach eingetretenen Fehlern

Ein SMF-Manifest kann entweder manuell oder mithilfe eines Python-Tools Manifest [5] erstellt werden. Dieser Assistent erfragt alle wichtigen Parameter und erstellt daraus ein entsprechendes XML-Dokument, das dann zum Beispiel wie folgt aussehen könnte:

```
# manifold rhnsd.xml

The service category (example: 'site' or
'/application/database') [site]

The name of the service, which follows
the service category
(example: 'myapp') [] rhnsd
...
The human readable name of the service
(example: 'My service.') [] Red Hat
Network Daemon

Can this service run multiple instances
(yes/no) [no] ?
...
Manifest written to rhnsd.xml
You can validate the XML file with "
svccfg validate rhnsd.xml"
And create the SMF service with "svccfg
import rhnsd.xml"
```

Zur einfacheren Installation des Python-Moduls existiert ein Framework namens „setuptools“ – dieses kann komfortabel über folgenden Aufruf installiert werden:

```
# wget https://bitbucket.org/pypa/
setuptools/raw/bootstrap/ez_setup.py -O
- | python
# easy_install Manifest
```

Wer das Installationsskript nicht ungeprüft ausführen möchte, kann dafür folgende Aufrufe verwenden:

```
# wget --no-check-certificate https://pypi.python.org/packages/source/M/Manifold/Manifold-0.2.0.tar.gz
# tar xzf Manifold-0.2.0.tar.gz ; cd Manifold-0.2.0
# python setup.py install
```

Für die manuelle Erstellung eines SMF-Manifests existiert auf der Oracle-Webseite ein Whitepaper [6], welches das Vorgehen exemplarisch für einen PostgreSQL-Server beschreibt.

Auf GitHub [7] befindet sich darüber hinaus eine Manifest-Vorlage, die ebenfalls verwendet werden kann. Um diese zu benutzen, muss die Datei heruntergeladen und importiert werden. Anschließend kann der Dienst gestartet werden:

```
# wget https://raw.githubusercontent.com/stdevl/rhnsd-solman/master/rhnsd.xml
# svccfg validate rhnsd.xml
# svccfg import rhnsd.xml
# svcadm enable rhnsd
```

Sobald **rhnsd** gestartet wurde, kann das System über Spacewalk verwaltet werden:

```
# ps -ef|grep -i rhn
root 6306 11 0 17:19:32 ?
0:00 /opt/redhat/rhn/solaris/usr/
sbin/rhnsd --foreground --interval
=10 -v
```

### Upload von Software-Paketen

Solaris-Programmpakete verfügen über ein spezielles Datenformat. Damit diese mit Spacewalk verteilt werden können, muss eine Konvertierung

in das MPM-Format durchgeführt werden. MPM-Pakete beinhalten neben dem eigentlichen Binärinhalt zusätzliche Metainformationen. Die Konvertierung muss über das Solaris-System mithilfe des Kommandos **solaris2mpm** erfolgen:

```
# solaris2mpm --select-arch=i386 webmin-1.680.pkg
Opening archive, this may take a while
Writing WWebmin-1.680-1
_PSTAMP_Jamie_Cameron.i386-solaris.mpm
```

Bei Software-Paketen, die für mehrere Architekturen konzipiert wurden, ist es notwendig, den Schalter **--select-arch** unter Angabe der verwendeten Architektur (i386 bzw. sparc) zu benutzen.

Nachdem das Paket konvertiert wurde, kann es mittels des Befehls **rhnpush** auf den Spacewalk-Server hochgeladen werden. Dabei müssen der Hostname, gültige Login-Informationen und der Name des Solaris-Software-Kanals mit

*Ein hochgeladenes MPM-Paket.*

angegeben werden. Darüber hinaus muss der Benutzer für die Ausführung über die „Channel Administrator“-Rolle verfügen:

```
# rhnpush -v --server fqdn-spacewalk.
domain.loc --username admin -c solaris
-11 *.mpm
Connecting to http://fqdn-spacewalk.
domain.loc/APP
Red Hat Network password:
Package WSwebmin-1.680-1
_PSTAMP_Jamie_Cameron.i386-solaris.mpm
Not Found on RHN Server -- Uploading
Uploading package WSwebmin-1.680-1
_PSTAMP_Jamie_Cameron.i386-solaris.mpm
Using POST request
```

Anschließend steht das Paket auf dem Spacewalk-Server zur Verfügung und kann verteilt werden.

Eine eingeplante Paket-Installation wird bei der nächsten rhnsd-Kommunikation vorgenommen. Alternativ kann auch das **rhnc**-Kommando auf dem betroffenen System ausgeführt werden, um die Installation sofort vorzunehmen:

```
# rhnc -v
Installing packages [['WSwebmin', '
1.680', '1_PSTAMP_Jamie_Cameron', 'i386-
solaris', 'solaris-11'], {}]]
Updating cache...

Computing transaction...
Fetching packages...
...
```

```
Committing transaction...
pkgadd -a /opt/redhat/rhn/solaris/var/
lib/smart/adminfile -n -d /opt/redhat/
rhn/solaris/var/lib/smart/packages/
WSwebmin-1.680-1_PSTAMP_Jamie_Cameron.
i386-solaris.pkg WSwebmin
Installing WSwebmin

Updating cache...

Package list refresh successful
Doing checkNeedUpdate
Updating cache...

Package list refresh successful
```

Der nächste und letzte Teil dieser Artikel-Serie wird sich der Fehlersuche und der Zukunft von Spacewalk widmen.

## LINKS

- [1] <http://www.freiesmagazin.de/freiesMagazin-2014-11>
- [2] [https://access.redhat.com/documentation/en-US/Red\\_Hat\\_Satellite/5.6/html-single/Reference\\_Guide/index.html#sect-Reference\\_Guide-UNIX\\_Support\\_Guide](https://access.redhat.com/documentation/en-US/Red_Hat_Satellite/5.6/html-single/Reference_Guide/index.html#sect-Reference_Guide-UNIX_Support_Guide)
- [3] <http://mirror.opencsw.org/opencsw/>
- [4] <http://www.spacewalkproject.org/solaris/>
- [5] <https://code.google.com/p/manifold/>
- [6] <http://www.oracle.com/technetwork/server-storage/solaris10/solaris-smf-manifest-wp-167902.pdf>
- [7] <https://github.com/stdevel/rhnsd-solman>

## Autoreninformation

**Christian Stankowic** ([Webseite](#)) beschäftigt sich seit 2006 mit Linux und Virtualisierung. Nachdem er privat Erfahrungen mit Debian, CRUX und ArchLinux sammeln konnte, widmet er sich seit seiner Ausbildung zum Fachinformatiker insbesondere RHEL, CentOS, Spacewalk und Icinga.

 [Teilen](#)

 [Kommentieren](#)



OCTOBER 8<sup>TH</sup>, 2004:

A CHILD SWALLOWS AN 'OPERATION' BUZZER, LEADING TO THE SINGLE MOST DIFFICULT SURGERY EVER PERFORMED.

“Incision” © by Randall Munroe  
(CC-BY-NC-2.5), <http://xkcd.com/738/>

## Kurztipp: tmux mit Byobu nutzen von Sujeevan Vijayakumaran

**M**it dem Programm tmux ist es möglich, laufende Terminal-Programme von der aktuellen Sitzung zu trennen und wieder anzuhängen. Die darin laufenden Programme laufen im Hintergrund weiter. Byobu [1] erweitert einige Funktionen von tmux und erleichtert den Einsatz und den Umgang mit tmux.

Das Programm tmux wurde in der Mai-Ausgabe von freiesMagazin [2] bereits ausführlich von Wolfgang Hennerbichler beschrieben. Byobu selbst ist mehr ein Erweiterungs-Script bzw. ein Hilfsprogramm, welches auf tmux aufbaut. Byobu bietet unter anderem eine einfach zu konfigurierende Statusbar, bei der man nicht viel Zeit und Aufwand in die Erstellung von Konfigurationsdateien stecken muss.

Byobu lässt sich einfach über die Paketverwaltung installieren, sofern es in den Paketquellen der Distribution vorhanden ist. Ubuntu beispielsweise bietet eine Version in den Paketquellen an.

### Der erste Start

Byobu kann im Terminal mit dem eigenen Namen gestartet werden:

```
$ byobu
```

Nach dem Starten erscheint am unteren Fenster- rand eine Status-Leiste, wie man es von Screen

oder tmux gewohnt ist. Im Gegensatz zu eben diesen Programmen werden in Byobu im Standard schon einige Informationen über die laufende Sitzung und über das System angezeigt. Darunter fällt in etwa die aktuelle Uhrzeit, die Anzahl der verfügbaren Aktualisierungen, die Uptime oder auch der aktuelle RAM-Verbrauch.



Die Standard-Statusbar von Byobu unter Ubuntu 14.04. 🔍

Byobu lässt sich über zwei Arten von Tastatur-Kürzeln bedienen. Wenn man etwa neue Sitzungen öffnen oder zwischen den bereits geöffneten Sitzungen wechseln möchte, dann kann man dies über die Funktionstasten (F-Tasten) tun. Durch das Drücken der Taste **F2** öffnet sich etwa eine neue Terminal-Sitzung. Häufig werden einzelne dieser Sitzungen auch Tabs genannt, wie man sie aus Browsern oder aus anderen gängigen Terminal-Emulatoren kennt. Wenn man hingegen zwischen den Sitzungen wechseln möchte, kann man mit **F3** zur vorherigen Sitzung nach links oder mit **F4** zur nächsten Sitzung nach rechts wechseln.

Die Steuerung von Byobu über die Funktions-tasten ist nicht die einzige vorhandene Möglichkeit. Nutzer von Screen oder tmux sind etwa die Steuerung über die Tastenkombination **Strg + A**

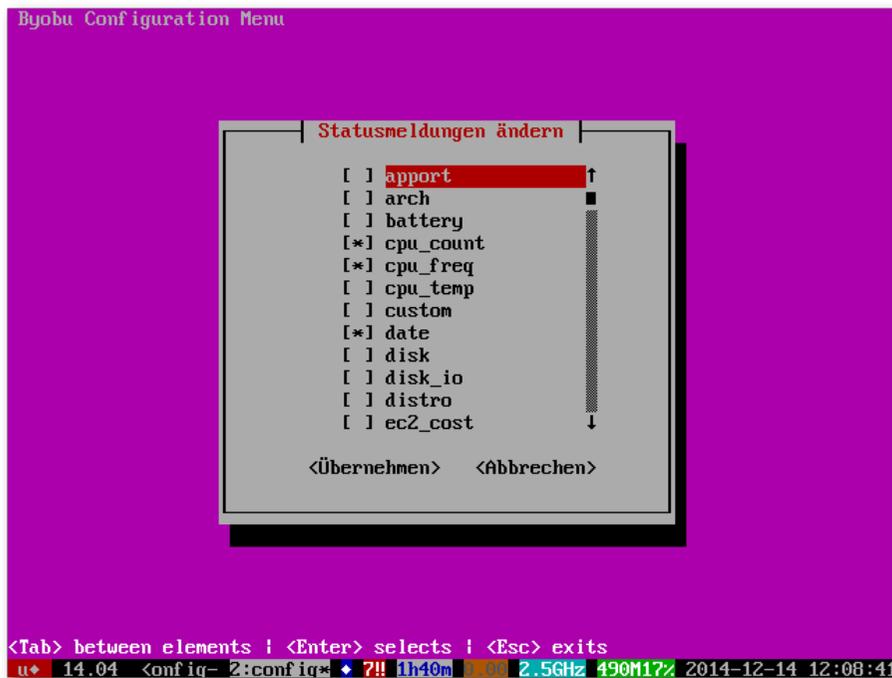
gewohnt. Sobald man diese Tastenkombination auf einer frischen Byobu-Installation durchführt, öffnet sich eine Abfrage in der Konsole, welches Verhalten man für die Tastenkombination wünscht. Angeboten werden hier zwei Varianten: Die erste Variante ist die Nutzung des Screen-Modus, alternativ kann man den Emacs-Modus verwenden.

Der Emacs-Modus ist das Standard-Verhalten von Konsolen. In einer normalen Konsole verschiebt sich der Cursor durch **Strg + A** auf den Anfang der Zeile.

### Konfiguration

Wie bereits erwähnt, lassen sich in der Statusbar diverse Informationen darstellen. Hierfür muss man keine Konfigurationsdateien anpassen. Durch das Drücken von **F9** öffnet sich zunächst das Konfigurationsmenü von Byobu.

Der erste Punkt ist die Hilfe, in der alle nötigen und vorhanden Tastenkombinationen aufgeführt und kurz erläutert werden. Darunter fallen sowohl das simple Wechseln zwischen den Sitzungen, als auch etwas komplexere Dinge wie das gleichzeitige Anzeigen zweier Sitzungen auf dem Bildschirm.



Im Konfigurationsmenü kann man diverse Informationen an- und ausschalten. 🔍

Im zweiten Menüpunkt kann man die Statusmeldungen in der unteren Statusbar bearbeiten. Dort lassen sich über 30 Informationen aktivieren oder deaktivieren. Je nach Einsatzzweck und persönlichen Vorlieben kann man so die eine oder andere Option aktivieren. So kann man etwa den Hostnamen, die IP oder auch Informationen über die CPU, Festplatte, RAM oder Netzwerk darstellen lassen.

Die dritte Option ist die Änderung der Escape-Folge. Dies ist, wenn man sich im Screen-Modus befindet, die Tastenkombination `Strg` + `A`.

Sofern gewünscht, kann man die Escape-Folge auch auf eine andere Tastenkombination mappen.

Die letzte Konfigurationsoption ist das automatische Starten von Byobu beim Login. Wenn man diese Funktion aktiviert, muss man hier allerdings beachten, dass sich dies in der Regel nur auf das Einloggen in der Konsole ohne graphische Oberfläche bezieht. Es wirkt sich allerdings auch auf das Einloggen auf einen Rechner per SSH aus. In beiden Varianten startet beim Einloggen automatisch Byobu. Wenn man beispielsweise auf einem Server Byobu

installiert hat, bietet es sich gegebenenfalls an, diese Option zu aktivieren, da man somit beim erneuten Verbinden im selben Zustand landet, wie es beim Schließen der vorherigen Verbindung der Fall war.

### Sonstiges

Byobu lässt sich nicht nur mit tmux nutzen. Wenn man möchte, kann man auch auf Screen zurückgreifen. Dazu reicht es, folgenden Befehl auszuführen:

```
$ byobu-select-backend
```

Anschließend kann man zwischen tmux oder Screen wechseln. Der Standard ist jedoch tmux.

Es ist weiterhin möglich, auf einem System mehrere Byobu-Sessions zu starten. Bei einem Ausführen von `byobu` im Terminal öffnet sich, sofern keine Session vorhanden ist, automatisch eine neue Session. Wenn man allerdings explizit eine weitere Session starten will, dann ist dies mit dem folgenden Befehl möglich:

```
$ byobu new-session
```

### Fazit

Byobu bietet eine einfache Möglichkeit, tmux oder Screen zu nutzen, ohne viel Aufwand, um praktische Informationen in der Statusbar darstellen zu können. Zudem bietet es einen guten Einstieg in die Nutzung von Terminal-Multiplexer.

### LINKS

- [1] <http://byobu.co/> 🇬🇧
- [2] <http://www.freiesmagazin.de/20140504-maiausgabe-erschienen>

### Autoreninformation

**Sujeevan Vijayakumaran** ([Webseite](#))  
wechselte vor geraumer Zeit von Screen zu Byobu und ist dank der einfachen Einrichtung sehr zufrieden.

✉ [Teilen](#)

💬 [Kommentieren](#)

## Tiny Tiny RSS – Ein web-basierter Feed-Aggregator von Matthias Sitte

**D**as Internet – unendliche Weiten. Wir schreiben das Jahr 2015. Dies sind die Abenteuer des Autors, der mit zwei Workstations, einem Laptop und je einem Smartphone und Tablet unterwegs ist und News-Feeds liest. Viel in der Welt unterwegs, dringt er in ungeahnte Problem-Welten vor, die sicherlich einigen Lesern bekannt vorkommen.

Das ist sicherlich arg zugespitzt, aber so oder so ähnlich fühlt sich wahrscheinlich jeder, der News-Feeds auf mehr als einem Endgerät abonniert hat und sich eine Synchronisation der News-Feeds wünscht. Und das, obwohl das World Wide Web nun mittlerweile mehr als 25 Jahre alt ist und der RSS 2.0 Standard auch schon gut 11 Jahre auf dem Buckel hat. Das Hauptproblem ist nämlich, dass man News-Feeds nicht oder nur auf sehr beschwerlichem Wege zwischen Endgeräten verschiedenster Natur synchronisieren kann. Dieser Artikel stellt Tiny Tiny RSS [1] vor, einen web-basierten Feed-Aggregator [2], den man auf seinem eigenen Server betreibt.

### Warum Tiny Tiny RSS?

Bei der Suche nach einem geeigneten Feed-Aggregator wird schnell klar, dass man nicht alle Ziele gleichzeitig erfüllen kann: Einerseits möchte man sich nicht allzu sehr mit den technischen Details befassen müssen, greift also am einfachsten zu einem Feed-Service eines großen Anbieters;

andererseits möchte man aber auch seine Daten in Sicherheit wissen, da man über die Lesegeohnheiten ja ein sehr gutes Profil über Zielpersonen anlegen kann. Proprietäre Software hat für viele Anwender den Vorteil, dass Bugs meistens schnell vom Hersteller behoben werden; sie kann aber nicht an eigene Bedürfnisse angepasst werden. Wehe demjenigen, der sich zu sehr auf diesen Service verlässt, sollte er eingestellt werden, wie beispielsweise der vormals sehr beliebte Google Reader [3] ...

Gesucht ist also eine Open-Source-Software, die möglichst unabhängig vom Betriebssystem der Endgeräte ist, gleichzeitig aber eine möglichst harmonische und ansehnliche Übersicht der abonnierten News-Feeds liefert. Eine absolute Standard-Software, die auf allen Endgeräten in der Regel vorinstalliert ist, ist der Webbrowser. Daher bietet sich die web-basierte Open-Source-Software Tiny Tiny RSS [1] als Feed-Aggregator an. Diese unter der GNU GPLv3 stehende Software ist mittlerweile in Version 1.15.3 erschienen (Stand: 9. Dezember 2014) und bietet einen stattlichen Funktionsumfang. Dazu zählen unter anderem:

- Unterstützung von RSS-Feeds [4] und Atom-Feeds [5]
- Keyboard-Shortcuts zur einfachen Navigation
- Import/Export von OPML-Dateien [6]
- flexibel konfigurierbare Filter

- Detektion und Filterung von News-Duplikaten
- Unterstützung verschiedener Plug-ins und Themes

### Installation und Konfiguration

Während auf Seiten der Endgeräte nur ein Web-Browser benötigt wird, braucht man auf Server-Seite einen funktionierenden LAMP-Stack [7] bestehend aus einem Web-Server mit PHP-Unterstützung und einem Datenbank-Server. Die nachfolgende Installation wurde unter Debian [8] Wheezy (Point Release 7.7) getestet, sollte allerdings auf den meisten anderen Linux-Betriebssystemen ähnlich ablaufen.

#### LAMP-Stack auf dem Server

Um einen minimalen LAMP-Stack, bestehend aus Apache (2.2.22), PHP (5.4.35) und MySQL (5.5.40), unter Debian Wheezy einzurichten, installiert man einfach die entsprechenden Pakete **apache2** und **libapache2-mod-php5** sowie **mysql-server** und **php-mysql**. PostgreSQL wird zwar als Datenbank-Server empfohlen, weil Tiny Tiny RSS performanter und schneller laufen soll, aber Tiny Tiny RSS läuft auch mit einem MySQL-Server, wobei InnoDB [9] als Backend vorausgesetzt wird (MyISAM [10] wird nicht unterstützt). Gerade auf kleineren Systemen mit begrenzten Ressourcen oder Tiny Tiny RSS als einziger PostgreSQL-Anwendung bietet es sich daher an, sich auf einen Datenbank-Server zu beschränken.

Optional, aber empfohlen sind des Weiteren das PHP-CURL-Modul (**php5-curl**), damit Tiny Tiny RSS schneller Feeds herunterladen kann, und das PHP-GD-Modul (**php5-gd**), das Tiny Tiny RSS benutzt, um QR-Codes zu erstellen. Außerdem wird das Zusatzmodul APC (Alternative PHP Cache [11]) aus dem Paket **php-apc** empfohlen, das den Zugriff und die Ausführung des PHP-Codes beschleunigen soll.

### MySQL-Datenbank einrichten

Vor der eigentlichen Installation von Tiny Tiny RSS legt man zunächst einen dedizierten MySQL-Account **ttrss** und eine gleichnamige Datenbank für Tiny Tiny RSS an. Das im Folgenden verwendete Standard-Passwort **ttrss-passwd** sollte auf keinen Fall übernommen werden und muss entsprechend abgeändert werden – ein generisches Passwort kann man beispielsweise mittels **pwgen** [12] generieren.

```
$ mysql -u root -p
mysql> CREATE USER 'ttrss'@'localhost'
IDENTIFIED BY 'ttrss-passwd';
Query OK, 0 rows affected (0.00 sec)

mysql> CREATE DATABASE ttrss;
Query OK, 1 row affected (0.00 sec)

mysql> GRANT ALL PRIVILEGES ON ttrss.*
TO 'ttrss'@'localhost';
Query OK, 0 rows affected (0.00 sec)
```

Mit **quit** oder dem Shortcut **[Strg]+[D]** verlässt man MySQL erst einmal, da man für alles wei-

tere keine Root-Rechte unter MySQL benötigt. Anschließend loggt man sich in MySQL testweise mit dem Account **ttrss** ein, um zu testen, dass sich keine Tippfehler eingeschlichen haben:

```
$ mysql -u ttrss -p ttrss
Enter password:
mysql>
```

### Tiny Tiny RSS installieren

Nachdem man den LAMP-Stack mitsamt Datenbank eingerichtet hat, kann man nun Tiny Tiny RSS installieren. Tiny Tiny RSS steht in Version 1.15 auch als eigenes Paket im sog. Unstable-Zweig (sid) von Debian zur Verfügung [13]. Dazu muss man unter Debian zuerst den Unstable-Zweig zur Paket-Resource-List **/etc/apt/sources.list** von APT hinzufügen und anschließend das Paket **tt-rss** installieren. Dabei werden aber gleich eine ganze Reihe Pakete aktualisiert, um die Paket-Abhängigkeiten zu erfüllen.

Um diese Probleme zu vermeiden, wird Tiny Tiny RSS im Folgenden manuell installiert. Zunächst lädt man die aktuelle Version herunter – den aktuellen Download-Link findet man auf der Wiki-Seite [14] – und entpackt dann das gzip-te tar-Archiv:

```
$ cd /var/www/
$ wget https://github.com/gothfox/Tiny-Tiny-RSS/archive/1.15.tar.gz
$ tar -zxvf 1.15.tar.gz
```

Um den Pfadnamen, den man ja später in den Webbrowser eingeben muss, möglichst kurz zu halten, bietet es sich an, den Ordner umzubenennen oder alternativ einen Symlink [15] auf die jeweils aktuelle Version einzurichten:

```
$ rm 1.15.tar.gz
$ mv Tiny-Tiny-RSS-1.15 tt-rss
```

Gegebenenfalls passt man die Datei- und Zugriffsberechtigungen im Installationsverzeichnis an, damit Tiny Tiny RSS Schreibrechte im **cache**-Ordner hat:

```
# chgrp -R www-data cache/ feed-icons/ lock
# chmod -R g+w cache/ feed-icons/ lock
# chmod -R g+s cache/
```

Anschließend öffnet man im Webbrowser die Installationsseite **http://www.example.com/tt-rss/install/** und gibt dort die üblichen Datenbank-Einstellungen (Typ, Account, Passwort, Datenbankname, Host und Port) sowie die URL, unter der Tiny Tiny RSS erreichbar ist, ein. Sind alle Tests erfolgreich verlaufen, initialisiert man im nächsten Schritt die Datenbank durch einen Klick auf „*Initialize database*“. Anschließend versucht das Installations-Skript, die Konfiguration in der Datei **config.php** im Installationsverzeichnis abzulegen. Schlägt dies – beispielsweise auf Grund mangelnder Schreibrechte des Webrowsers – fehl, kopiert man den angezeigten Inhalt manuell in die **config.php**-Datei.

Der Installer von Tiny Tiny RSS. 🔍

Damit ist die Installation von Tiny Tiny RSS eigentlich schon beendet, und man kann sich mit dem Standard-Account **admin** und Passwort **password** auf der Startseite <http://www.example.com/tt-rss/> einloggen. Vorher sollte man sich allerdings noch einige erweiterte Einstellungen in der Konfigurationsdatei **config.php** anschauen.

### Erweiterte Konfiguration

Es ist in jedem Fall ratsam, sich die Konfigurationsdatei **config.php** aufmerksam durchzulesen und die Einstellungen an die eigenen Bedürfnisse anzupassen. Hervorzuheben sind hier die folgenden Einstellungen:

- **FEED\_CRYPT\_KEY** definiert einen 24-stelligen String, der für die Verschlüsselung von Passwort-geschützten Feeds genutzt wird. Einen geeigneten Schlüssel kann man beispielsweise via **pwgen 24 1** erzeugen.

- Die Multi-User-Unterstützung kann man durch Setzen von **SINGLE\_USER\_MODE** deaktivieren; in diesem Fall sollte man Tiny Tiny RSS allerdings anderweitig (z. B. mittels HTTP-Authentifizierung [16]) absichern.
- Standardmäßig deaktiviert ist **ENABLE\_REGISTRATION**, was Nutzern erlaubt, sich selbst Accounts anzulegen. Diese Einstellung sollte deaktiviert bleiben, es sei denn, man möchte einen offenen Feed-Dienst anbieten.
- Mit **ENABLE\_GZIP\_OUTPUT** kann man die Ausgabe von Tiny Tiny RSS zusätzlich mittels Gzip komprimieren; hierzu muss die PHP-Zlib-Erweiterung installiert sein. In der Standard-Installation von Apache ist aber bereits das Apache-Modul **deflate\_module** aktiviert, so dass eine zusätzliche Kompression nicht notwendig sein sollte.

Weitere Einstellungen bzgl. E-Mails, die Tiny Tiny RSS versendet, haben das Präfix **SMTP**; diese sollte man ebenfalls anpassen.

### Automatische Aktualisierung der News-Feeds

Bevor man Tiny Tiny RSS sinnvoll nutzen kann, muss eine Update-Methode eingerichtet werden, denn sonst werden die News-Feeds nicht aktualisiert. Die Feeds können dabei sowohl sequentiell (**update.php**) oder parallel mit mehreren Prozessen (**update\_daemon2.php**) aktualisiert werden, wobei man darauf ach-

ten sollte, dass man die PHP-Prozesse nicht als Root starten sollte, sondern am besten mit der User-ID des Webservers, um Konflikte in den Dateiberechtigungen zu vermeiden (unter Debian ist dies standardmäßig **www-data**). Außerdem benötigt man das PHP Command Line Interface (CLI), um diese Skripte außerhalb des Webservers starten zu können.

### Update-Dämon

Die empfohlene Methode, um News-Feeds in Tiny Tiny RSS zu aktualisieren, verwendet einen Update-Dämon, der im Hintergrund auf dem Server läuft. Hierzu benötigt man allerdings vollen Zugang, d. h., Root-Rechte auf dem Server, um den Dämon einzurichten. Das sequentielle Update der News-Feeds startet man mit dem Befehl

```
$ php ./update.php --daemon
```

Schneller geht's mit mehreren, parallelen PHP-Prozessen, die mit dem folgenden PHP-Skript gestartet werden:

Startseite mit Login. 🔍

```
$ php ./update_daemon2.php
```

Diese Skripte forken aber nicht, das heißt, sie erzeugen keinen Prozess, der mit der Shell nicht mehr verbunden und damit im Hintergrund (detached) läuft. Allerdings kann man beispielsweise mit Hilfe von **start-stop-daemon** ein entsprechendes Init-Skript schreiben, das im Ordner `/etc/init.d/` gespeichert werden kann. Das Debian-Paket **tt-rss** im Unstable-Zweig [13] enthält ein SysV-Init-Skript, das man aber noch an die eigene Installation anpassen muss. Man sollte aber auf jeden Fall darauf achten, dass in der Konfigurationsdatei die Variable **PHP\_EXECUTABLE** den richtigen Pfad zum PHP CLI enthält, da ansonsten keine Tasks vom Update-Skript **update\_daemon2.php** gestartet werden können.

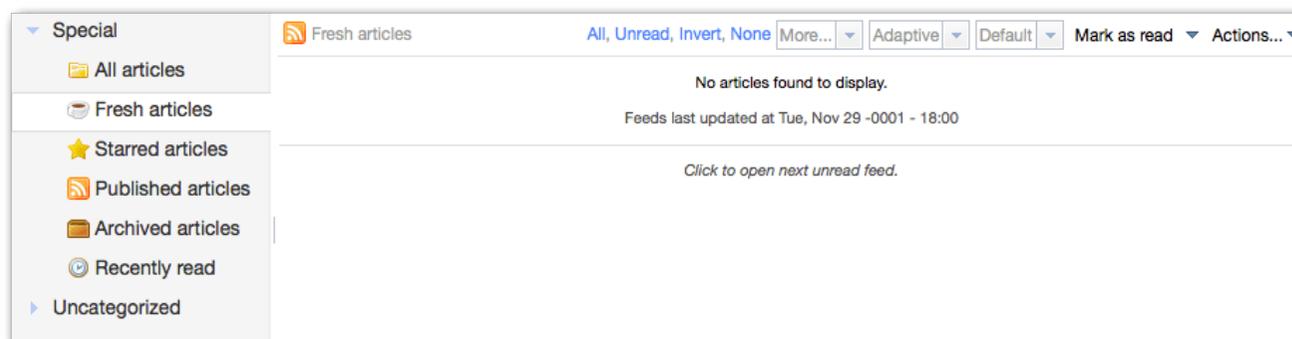
### Periodische Updates mittels Crontab

Wenn das PHP CLI installiert ist, man aber keinen Zugang zu den Server-Diensten hat, kann man die News-Feeds periodisch mittels Crontab [17] aktualisieren. Dazu bearbeitet man mit dem Befehl

```
$ crontab -u www-data
```

die Crontab-Datei des System-Accounts, unter dem der Webserver läuft, und fügt folgende Zeile hinzu:

```
*/30 * * * * /usr/bin/php /var/www/tt-rss/update.php --feeds --quiet
```



Web-Interface von Tiny Tiny RSS. 🔍

Dies ruft alle 30 Minuten das PHP-Skript **update.php** auf und aktualisiert alle Feeds in sequentieller Reihenfolge. Hier sollte man darauf achten, dass der Pfad `/usr/bin/php` auch auf das PHP CLI zeigt. `/var/www/tt-rss/` ist der Installationspfad, in dem Tiny Tiny RSS wie oben beschrieben installiert wurde und der ggf. an die eigene Installation anzupassen ist.

### Manuelle Updates

Wenn alle Stricke reißen und kein PHP CLI zur Verfügung steht, kann man den „einfachen Update-Modus“ von Tiny Tiny RSS aktivieren, indem man in der Konfigurationsdatei **config.php** die Variable **SIMPLE\_UPDATE\_MODE** auf **true** setzt. Solange Tiny Tiny RSS dann im Webbrowser geöffnet ist, werden die News-Feeds periodisch aktualisiert. Dies funktioniert aber nicht mit Dritt-Anwendungen, beispielsweise Apps auf Smartphones, welche auf die API und nicht direkt auf das Web-Interface von Tiny Tiny RSS zurückgreifen.

### Das Web-Interface von Tiny Tiny RSS

Nachdem man Tiny Tiny RSS erfolgreich entpackt, installiert und konfiguriert hat, kann man Tiny Tiny RSS nun im Browser unter der Adresse <http://www.example.com/tt-rss/> öffnen. Auf der Login-Seite kann man sich zunächst nur mit dem Standard-Account **admin** und Passwort **password** einloggen.

Das Web-Interface ist im Wesentlichen zweigeteilt und sehr übersichtlich gehalten: In der linken Spalte findet man eine Übersicht über die abonnierten Feeds, sortiert nach Kategorien und Labels. Der übrige Teil des Fensters wird von Tiny Tiny RSS zur Darstellung der News-Feeds verwendet. In der oberen Zeile sind zudem noch Buttons bzw. Dropdown-Menüs zu finden, mit deren Hilfe man die Artikel schnell und einfach verwalten kann. Besonders hervorzuheben ist das Menü „*Aktionen...*“ (bzw. „*Actions...*“ in der voreingestellten englischen Lokalisation), das rechts oben im Web-Interface zu finden ist. Hier kann man direkt Feeds

abonnieren, bearbeiten und löschen sowie Artikel als gelesen markieren. Hilfreich sind auch die vielen Keyboard-Shortcuts, die eine schnelle Navigation zwischen den Artikeln, aber auch zwischen den News-Feeds erlauben. Dies funktioniert natürlich am besten, wenn man eine Tastatur zur Verfügung hat, also am Desktop.

## Einstellungen

Das Passwort des Standard-Accounts **admin** sollte man nach dem Login sofort ändern, da dieser Account Admin-Rechte hat. Dazu öffnet man im Reiter „*Einstellungen*“ das Untermenü „*Persönliche Daten/Authentifizierung*“ und gibt dort das neue Passwort ein. Nach einem weiteren Klick auf „*Speichern*“ muss man sich ggf. erneut einloggen, da die alten Session-Credentials durch die Passwort-Änderung ungültig geworden sind.

## Allgemeines

Im gleichnamigen Untermenü „*Einstellungen*“ legt man das generelle Verhalten von Tiny Tiny RSS fest, beispielsweise Sprache, Zeitzone, Standard-Intervalle für Feed-Updates und neue Artikel, etc. Im Untermenü „*Plugins*“ kann man weitere System- bzw. Benutzer-Plug-ins nach Belieben aktivieren. Zu guter Letzt bietet das Untermenü „*Tiny Tiny RSS updaten*“ eine kurze Übersicht über die installierte und die derzeit aktuelle Version an. Weichen diese voneinander ab, kann man versuchen, ein Update von Tiny Tiny RSS über das Web-Interface anzustoßen – dies setzt allerdings Schreibrechte des Webservers (unter Debian läuft Apache als Nutzer **www-data**) im In-

stallationsverzeichnis (hier `/var/www/tt-rss/`) voraus. In jedem Fall ist vor der Aktualisierung ein Backup der Daten empfohlen.

## Feeds

Im Reiter „*Feeds*“ kann man übersichtlich die abonnierten Feeds verwalten und bearbeiten beziehungsweise neue News-Feeds hinzufügen. Außerdem kann man die Feeds in sogenannte „*Kategorien*“ einordnen, die im Web-Interface sortiert angezeigt werden. Im Untermenü „*OPML*“ können Feeds, Filter, Label und Tiny-Tiny-RSS-Einstellungen als OPML-Datei [6] sowohl im- als auch exportiert werden. Wenn man Artikel „teilt“, werden diese Artikel als öffentlicher RSS-Feed exportiert, der von jedem abonniert werden kann, sofern man die entsprechende URL kennt. Diese kann man im entsprechenden Untermenü anzeigen.

## Filter

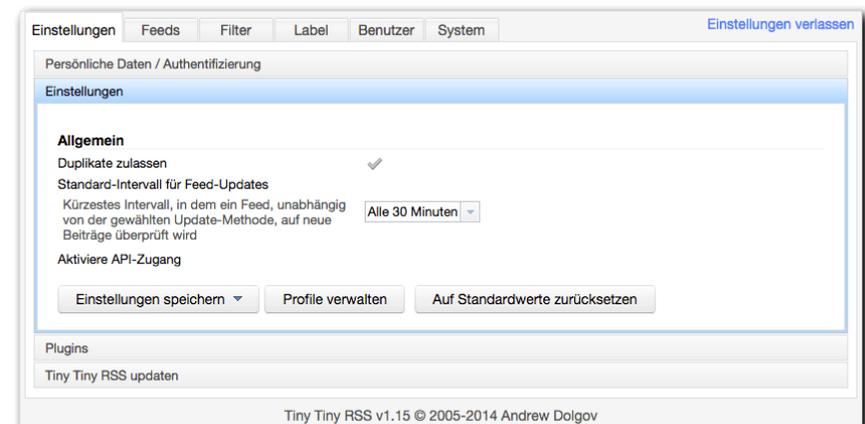
Hinter dem Reiter „*Filter*“ verbirgt sich eine der Hauptfunktionen von Tiny Tiny RSS: Hier kann man Filter definieren, die automatisch beim Feed-Update ausgeführt werden. Dabei kann man durchaus mehrere Kriterien, die aus regulären Ausdrücken [18] bestehen dürfen, miteinander verknüpfen, um Artikel beispielsweise als gelesen zu markieren oder direkt zu löschen.

## Label

Gleichermaßen interessant ist der Reiter „*Label*“, in dem man selbst eine Vielzahl Labels erstellen kann. Durch entsprechende Filter kann man einzelnen Artikeln dann die zuvor erstellten Labels zuweisen. Durch Auswahl eines Labels in der linken Spalte im Web-Interface lassen sich dann alle Artikel mit gleichem Label anzeigen, ähnlich der Schlagwortwolke [19] (engl. „tag cloud“), die unter anderem auch **freiesMagazin** verwendet [20].

## Benutzer

Der Reiter „*Benutzer*“ stellt die Benutzerverwaltung von Tiny Tiny RSS dar. Der Klick auf eine Zeile öffnet einen kleinen Benutzereditor, in dem man schnell die Zugriffsberechtigung („Benutzer“, „Erfahrener Benutzer“ oder „Administrator“), das Passwort sowie die E-Mail-Adresse des Benutzers ändern kann. Weitere Details über einen Benutzer wie zum Beispiel die Anzahl



Konfigurationsseite des Web-Interface. 🔍

abonnierter Feeds erfährt man über den Button „Details“, nachdem man einen Benutzer durch Klick auf das Häkchen in der ersten Spalte markiert hat.

### System-Protokoll

Im letzten Reiter „System“ kann man das Systemprotokoll von Tiny Tiny RSS einsehen. Hier kann man nach der Installation verfolgen, ob beispielsweise die Zugriffsrechte für den Webserver im Installationsverzeichnis zu restriktiv sind, was sich oftmals in PHP- Fehlermeldungen des Typs **E\_USER\_WARNING** widerspiegelt.

### Fazit

Tiny Tiny RSS erlaubt es dem Nutzer, News-Feeds auf mehreren Endgeräten zu lesen und synchron zu halten, sofern man einen Webbrowser zur Hand hat. Das AJAX-unterstützte Web-Interface wirkt sehr aufgeräumt und bietet viel Platz für Artikel aus verschiedensten News-Feeds. Tiny Tiny RSS bietet auch interessante Features wie zum Beispiel tägliche Digests (Nachrichtensammlungen), Scoring (Bewertungen) von Artikeln, etc. Das Layout kann quasi nach Belieben an den eigenen Geschmack angepasst werden, sofern man sich mit Cascading Style Sheets (CSS [21]) auskennt und gewillt ist, dort Hand anzulegen.

Für Android gibt es bereits mehrere Reader [22]. Unter Mac OS X kann man bislang nur auf die (kostenpflichtige) App MicroRSS [23] zurückgrei-

fen, während für iOS-basierte Endgeräte (iPhone und iPad) eine Suche im US-basierten App Store bislang noch keine native Apps liefert (in anderen App Stores soll man beispielsweise auch YATTRSSC herunterladen können).

Einen kleinen Wermutstropfen gibt es dennoch: Tiny Tiny RSS ist nicht unbedingt die schnellste (Web-)Anwendung. Es dauert mitunter schon ein paar Sekunden, wenn man zwischen Web-Interface und Einstellungen hin- und herwechselt oder wenn man einen anderen News-Feed öffnet, selbst wenn man Tiny Tiny RSS nur testweise lokal betreibt. Teilweise ist dies dem Umstand geschuldet, dass man eben nicht die Hardware und Internetanbindung zur Verfügung hat wie zum Beispiel Google. Tiny Tiny RSS aktualisiert nämlich den Feed und lädt neue Artikel in dem Moment herunter, in dem man auf den Feed klickt, während manche andere App das mitunter automatisch im Hintergrund erledigt.

### LINKS

- [1] <http://tt-rss.org/> 
- [2] <http://de.wikipedia.org/wiki/Feedreader>
- [3] <http://googleblog.blogspot.com/2013/03/a-second-spring-of-cleaning.html> 
- [4] <https://de.wikipedia.org/wiki/RSS>
- [5] [https://de.wikipedia.org/wiki/Atom\\_\(Format\)](https://de.wikipedia.org/wiki/Atom_(Format))
- [6] [https://de.wikipedia.org/wiki/Outline\\_Processor\\_Markup\\_Language](https://de.wikipedia.org/wiki/Outline_Processor_Markup_Language)
- [7] [https://de.wikipedia.org/wiki/LAMP\\_\(Softwarepaket\)](https://de.wikipedia.org/wiki/LAMP_(Softwarepaket))

- [8] <https://www.debian.org/> 
- [9] <https://de.wikipedia.org/wiki/InnoDB>
- [10] <https://de.wikipedia.org/wiki/MyISAM>
- [11] [https://de.wikipedia.org/wiki/Alternative\\_PHP\\_Cache](https://de.wikipedia.org/wiki/Alternative_PHP_Cache)
- [12] <http://linux.die.net/man/1/pwgen> 
- [13] <https://packages.debian.org/sid/tt-rss> 
- [14] <http://tt-rss.org/redmine/projects/tt-rss/wiki> 
- [15] [https://de.wikipedia.org/wiki/Symbolische\\_Verknüpfung](https://de.wikipedia.org/wiki/Symbolische_Verknüpfung)
- [16] <https://de.wikipedia.org/wiki/HTTP-Authentifizierung>
- [17] <https://de.wikipedia.org/wiki/Cron>
- [18] [https://de.wikipedia.org/wiki/Regulärer\\_Ausdruck](https://de.wikipedia.org/wiki/Regulärer_Ausdruck)
- [19] <https://de.wikipedia.org/wiki/Schlagwortwolke>
- [20] <http://www.freiesmagazin.de/wortwolke>
- [21] [https://de.wikipedia.org/wiki/Cascading\\_Style\\_Sheets](https://de.wikipedia.org/wiki/Cascading_Style_Sheets)
- [22] [https://play.google.com/store/search?q=tiny\\_tiny\\_rss](https://play.google.com/store/search?q=tiny_tiny_rss) 
- [23] <https://itunes.apple.com/us/app/microrss/id709491925?mt=12>

**Autoreninformation** 

**Matthias Sitte** (Webseite) nutzt Tiny Tiny RSS, um den Überblick über seine gesammelten News-Feeds zu behalten.

 [Teilen](#)  [Kommentieren](#)

## Calibre – Teil 1: Installation und Erst-Konfiguration von Maria Seliger

**C**alibre ist eine Open-Source-Verwaltungssoftware für E-Books, die unter Linux, MacOS und Windows läuft. Dabei unterstützt das Programm die E-Book-Verwaltung auf E-Readern, das Herunterladen von E-Books aus dem Internet, die Bearbeitung und die Konvertierung von E-Books.

Dieser Artikel wird in mehreren Teilen erscheinen:

- Installation und Erst-Konfiguration von Calibre
- Lesen und Konvertieren von E-Books
- Die Gerätesynchronisation mit Calibre
- Die Erweiterung des Buchbestands
- Konfiguration von Calibre und Erweiterung der Funktionalitäten durch Plug-ins
- E-Book-Bearbeitung mit Calibre
- Tipps & Tricks für Calibre

### Installation von Calibre

Für die Installation von Calibre unter Linux erstellt man sich am besten ein Bash-Skript, da man mit diesem dann auch eine bereits bestehende Installation aktualisieren kann. Normalerweise gibt es jeden Freitag eine neue, verbesserte Version des Programms (Agile Programmierung), so dass Updates relativ häufig eingespielt werden können. Dazu wechselt man auf die Calibre-Webseite [1] und kopiert den Inhalt der Programmzeile unter „Binary install“ in einen Editor, wie zum Beispiel Leafpad:

```
#!/bin/bash
sudo -v && wget -nv -O- https://raw.githubusercontent.com/kovidgoyal/calibre/master/setup/linux-installer.py | sudo python -c "import sys; main=lambda:sys.stderr.write('Download failed\n'); exec(sys.stdin.read()); main()"
```

Anschließend speichert man diese Datei z. B. als **CalibreUpdate.sh**. Für die Installation bzw. für ein Update startet man das Skript dann über die Kommandozeile mit

```
$ sh CalibreUpdate.sh
```

Das Skript fragt nach dem Administratorenkennwort, lädt dann Calibre von der Webseite herunter und installiert das Programm bzw. macht ein Update. Dabei bleiben die alten Einstellungen erhalten.

Nach der Installation bzw. dem Update kann man Calibre entweder über die Kommandozeile starten oder z. B. über ein Startmenü wie in Ubuntu, wo die Applikation unter Office-Anwendungen eingetragen wird.

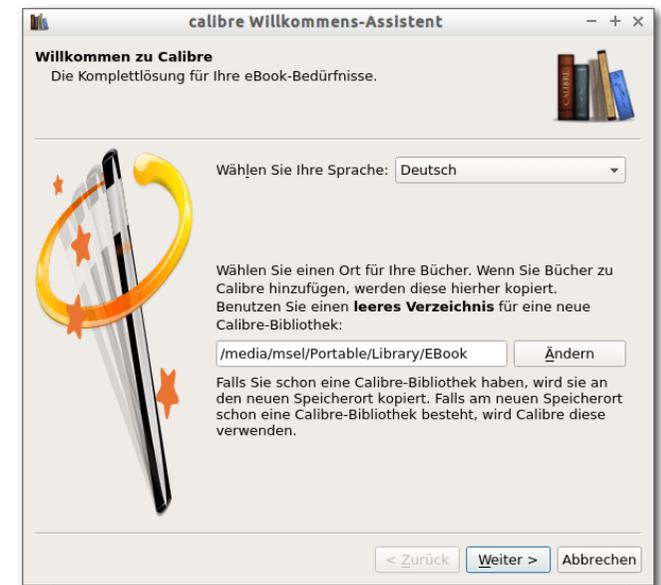
### Erst-Konfiguration von Calibre

Beim ersten Start von Calibre wird ein „Welcome Wizard“ (Willkommens-Assistent) gestartet. Hierbei handelt es sich um einen geführten Dialog.

**Hinweis:** Wenn sich das Lesegerät später einmal ändern sollte, so kann es sinnvoll sein, den Welcome-Wizard wieder aufzurufen. Das ist

jederzeit über die Calibre-Einstellungen im Programm möglich!

Über die Schaltfläche „Weiter“ kommt man zum nächsten Schritt, über „Zurück“ zum vorherigen Schritt und über „Abbrechen“ kann der Welcome Wizard abgebrochen werden und die alten Einstellungen (sofern vorhanden) werden verwendet.



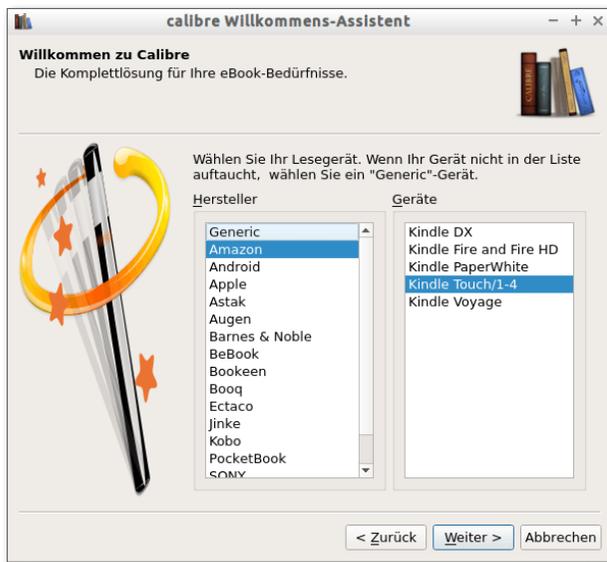
*Calibre Welcome Wizard, 1. Schritt: Auswahl der Sprache, hier Deutsch. *

## Auswahl der Sprache

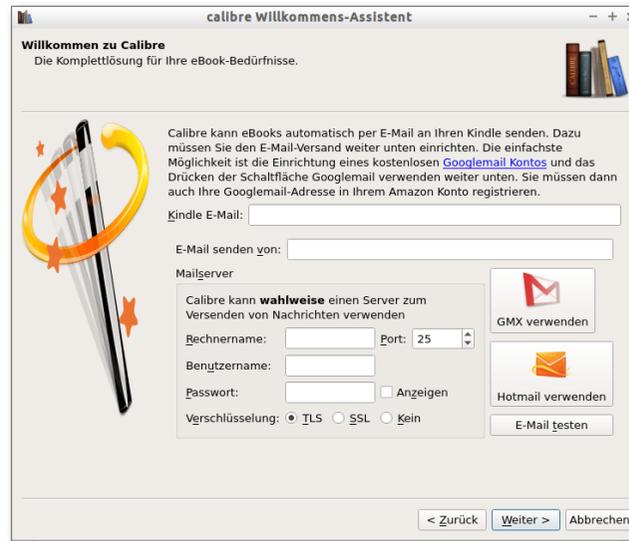
Im ersten Schritt wählt man die Sprache der Oberfläche für Calibre aus. Sobald man eine Sprache auswählt, ändert sich entsprechend die Oberfläche und es erscheinen die Menüpunkte in der ausgewählten Sprache. Über „Weiter“ gelangt man zum zweiten Schritt.

## Auswahl des Lesegeräts

Dieser Schritt ist der wichtigste im Welcome Wizard, da hier bestimmt wird, wie später Calibre und Lesegerät synchronisiert werden. Calibre unterstützt nativ diverse E-Book-Reader sowie Smartphone-Modellen und Tablets. Hier wird beispielhaft die Einrichtung eines Amazon Kindle Modells (Kindle 3 Keyboard 3G) beschrieben.



Calibre Welcome Wizard, 2. Schritt: Geräteauswahl.



Calibre Welcome Wizard, 3. Schritt: Einrichtung von Kindle E-Mail (optional).

In der linken Spalte befinden sich die Hersteller und in der rechten die Modelle. Zunächst wählt man den Hersteller in der linken Spalte aus, in diesem Fall „Amazon“. Hat man ein Modell, das nicht nativ von Calibre unterstützt wird, so gibt es oft Plug-ins, die diese Unterstützung hinzufügen, oder es bietet sich an, unter der Gerätetreiber-Schnittstelle, die später noch erklärt wird, einen USB-Treiber selbst zu definieren. In diesem Fall wählt man „Generic“ aus. Anschließend wählt man aus der rechten Liste das Modell, in diesem Fall „Kindle Touch/1-4“.

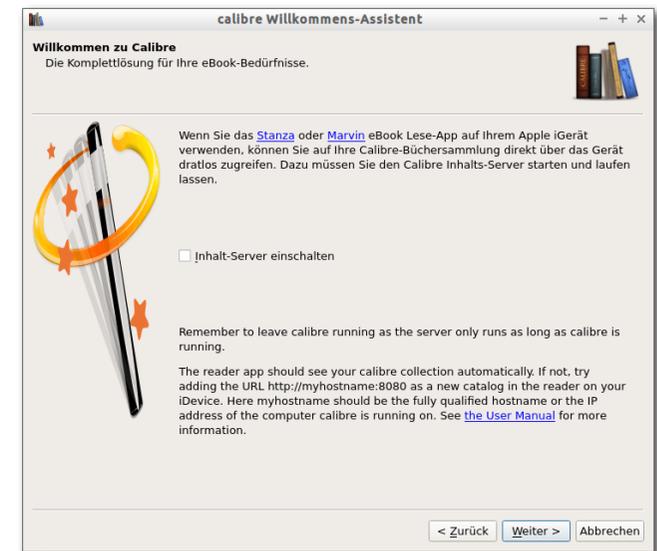
Über „Weiter“ kommt man zum nächsten Schritt des Wizards. Bei den meisten Modellen ist keine weitere Konfiguration erforderlich, wohl aber

zum Beispiel beim Kindle oder bei einem Apple iPhone.

## Kindle E-Mail einrichten (optional)

Wenn man ein Kindle-Modell hat, so kann man jetzt, sofern dies denn gewünscht ist, eine E-Mail-Adresse für den Kindle einrichten, über die E-Books automatisch per Mail an den Kindle gesendet werden.

**Hinweis:** Je nach Konfiguration des Kindles bei Amazon können dadurch jedoch Kosten für den E-Mail-Empfang entstehen, wenn diese per 3G-Verbindung über Amazon und nicht über WLAN zustande kommen.



Calibre Welcome Wizard, 4. Schritt: Einrichtung des Inhalt-Servers bei Apple Devices (optional).



Calibre Welcome-Wizard, 5. Schritt: Beenden des Welcome-Wizards. 🔍

Der Wizard bietet einige vorkonfigurierte Konten, z. B. für GMX oder Hotmail. Wie die einzelnen Einträge auszusehen haben, erfährt man von seinem E-Mail-Provider, falls man diese Funktionalität nutzen möchte. Wenn man Bücher an den Kindle nur über USB, aber nicht per WLAN senden will, braucht man hier keine weitere Einrichtung zu machen.

Über die Schaltfläche „E-Mail testen“ wird eine Test-E-Mail an den E-Book-Reader gesendet.

### Apple-Devices konfigurieren (optional)

Hat man ein Apple-Gerät und nutzt die App Marvin [2], so wird diese über den Calibre Inhalt-Server unterstützt. In diesem Fall aktiviert man das entsprechende Kästchen.

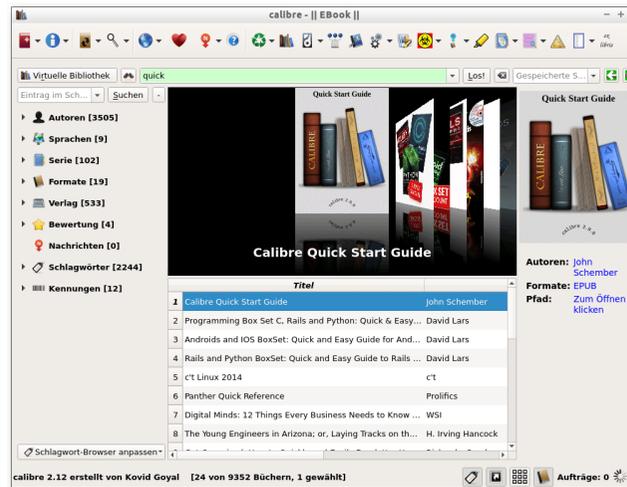
### Abschluss des Welcome-Wizards

Im letzten Schritt finden sich Hinweise zu Videoanleitungen und zum Benutzerhandbuch zu Calibre. Über „Fertigstellen“ wird der Wizard beendet und eine neue Calibre-Bibliothek erstellt. Calibre wird dann gestartet.

### Erster Start von Calibre

Wenn man das erste Mal Calibre startet, so wird eine neue Bibliothek angelegt. Anschließend öffnet sich Calibre. Das Fenster ist dabei folgendermaßen aufgeteilt:

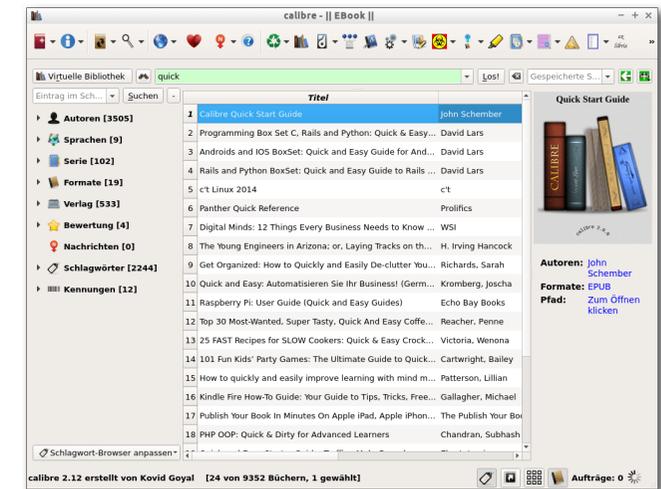
Oben befindet sich die Menüleiste mit den einzelnen Funktionen, die Calibre unterstützt. Darunter befinden sich verschiedene Funktionen, z. B. die Möglichkeit, eine „Virtuelle Bibliothek“ zu erstellen.



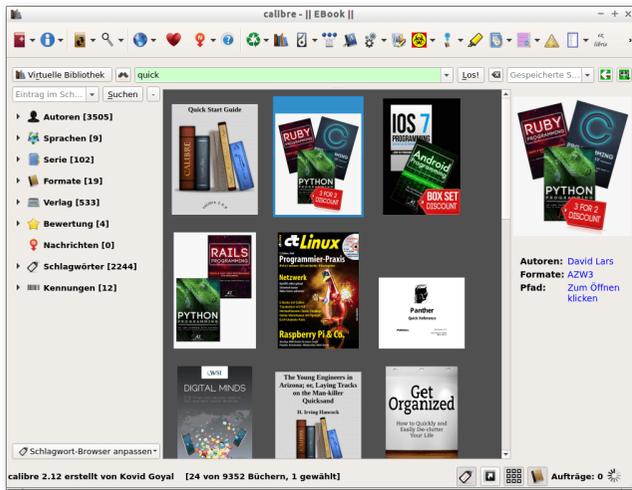
Calibre mit ausgewähltem Buch und Coverbild-Ansicht. 🔍

Was genau sich dahinter verbirgt, wird später noch genauer erklärt. Außerdem befindet sich eine Suchleiste dort, die unter anderem eine Volltextsuche über alle Metadaten der Bücher der Bibliothek ermöglicht.

Links unter der Menüleiste befindet sich der Schlagwort-Browser (Tag-Browser), Darüber können einzelne Bücher ausgewählt werden. Dabei kann auch eine Auswahl nach Autoren, Sprachen, Serien, Formaten, Verlagen, Bewertung, Nachrichten, Schlagwörter und Kennungen (zum Beispiel ISBN) zur Einschränkung erfolgen. In der Mitte befinden sich schließlich die eigentlichen Bücher, hier als Bücherliste. Im rechten Bereich erhält man eine Vorschau auf das zuvor ausgewählte Buch.



Das Calibre-Fenster mit Menüleiste, Schlagwort-Browser, Listenansicht der Bücher sowie Buchvorschau. 🔍



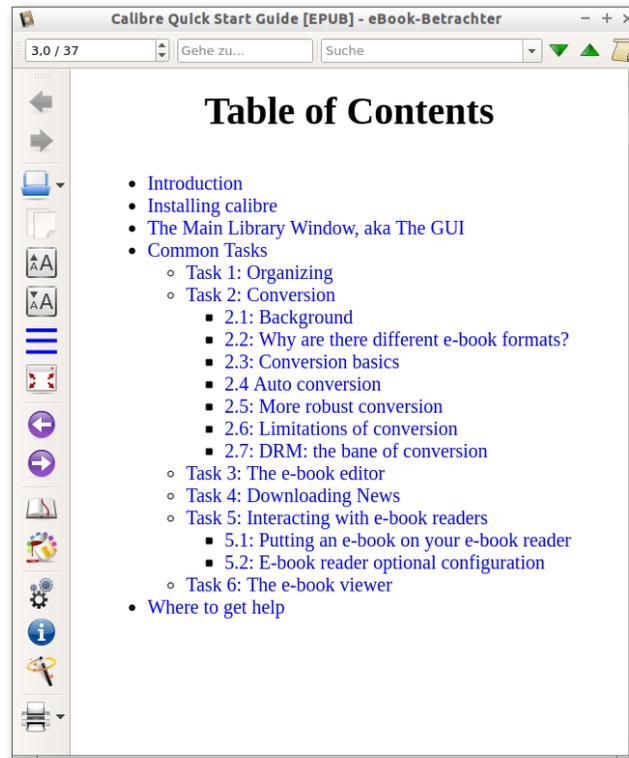
Calibre mit Cover-Raster, das eine Bibliotheks- oder Regalansicht der Bücher ermöglicht. 🔍

Im unteren Bereich gibt es vier Symbole, um die Ansicht anzupassen. Das erste Symbol blendet den Schlagwort-Browser links ein bzw. aus. Das rechte Symbol daneben blendet den Cover-Browser wahlweise ein und aus.

Die dritte Schaltfläche blendet ein Cover-Raster ein bzw. aus und ermöglicht eine Regalansicht der Bücher.

Die rechte Schaltfläche blendet die Buch-Detail-Ansicht ein und aus. Daneben gibt es einen Informationsdialog „Aufträge“, der aktiv ist, sobald z. B. eine Konvertierung von Büchern anfällt.

In jedem Fall befindet sich in der Bibliothek, falls diese neu angelegt wurde, das Buch „Calibre Quick Start Guide“ von John Sember, das in



Calibre mit geöffnetem Buch „Calibre Quick Start Guide“. 🔍

englischer Sprache verfasst ist (es gibt allerdings auch eine deutsche Version [3]). Das Buch ist für alle empfehlenswert, die das erste Mal mit Calibre arbeiten, da hier alle grundlegende Schritte, die mit Calibre gemacht werden, beschrieben werden. Zum Öffnen des Buches klickt man einfach doppelt auf das Buch. Dieses wird dann im internen E-Book-Betrachter geöffnet.

Oberhalb des Buches befindet sich eine Suchleiste, ferner kann ein fortlaufender Modus für die

Ansicht des Buches an- und ausgeschaltet werden. Schließen kann man den Viewer über die rechte obere Ecke.

Außerdem gibt es ein ausführliches Benutzerhandbuch in verschiedenen Sprachen und Formaten (azw3 (Kindle), epub, PDF) und auch in deutscher Sprache [4].

Im nächsten Teil wird gezeigt, wie man mit Calibre Bücher lesen, konvertieren und auf das Lesegerät übertragen kann.

## LINKS

- [1] [http://calibre-ebook.com/download\\_linux](http://calibre-ebook.com/download_linux) 
- [2] <http://www.appstafarian.com/marvin.html> 
- [3] [https://github.com/kovidgoyal/calibre/tree/master/resources/quick\\_start](https://github.com/kovidgoyal/calibre/tree/master/resources/quick_start)
- [4] <http://manual.calibre-ebook.com/de/index.htm>

## Autoreninformation

**Maria Seliger** ([Webseite](#)) ist bereits vor über einem Jahr von Windows 7 auf Ubuntu umgestiegen, was trotz anderer Erwartungen schnell und problemlos ging, da sich für die meisten Programme unter Windows eine gute Alternative unter Linux fand.



## Fuchs in Flammen: Mozilla Flame im Test von Dominik Wagenführ

**D**er Markt für mobile Betriebssysteme ist 2014 fest in einer Hand [1]: Google's Android dominiert weltweit den ganzen Markt. Einzig Apples iOS kann teilweise noch seine Stellung im zweistelligen Prozentbereich behaupten. Es gibt aber auch noch zahlreiche andere Betriebssystemhersteller, die einen Fuß in die Tür bekommen wollen. Einer davon ist Mozilla mit FirefoxOS. Dieser Artikel stellt das Betriebssystem zusammen mit dem Referenz-Entwickler-Gerät namens Flame vor.

### Mobile Betriebssysteme

Es gibt viele mobile Betriebssysteme. Wie in der Einleitung beschrieben dominiert Googles Android [2] ganz klar den Markt. Neben den proprietären Alternativen wie iOS oder Windows Mobile gibt es aber auch einige Betriebssystemalternativen auf Basis Freier Software (aber nicht alle Projekte sind komplett frei):

- Firefox OS [3]
- Sailfish OS [4]
- Tizen [5]
- Ubuntu Touch [6]

Von diesen vier Systemen gibt es aber nur eines, was aktuell tatsächlich etwas breiter am Markt angeboten wird: Firefox OS [7]. Neben dem Geeksphone [8] und einigen Modellen der Firma ZTE [9], die auch in Deutschland vertrieben werden [10],

hat Mozilla ein Referenz-Smartphone namens Flame auf den Markt gebracht [11]. Das Telefon kann man für knapp 170 Euro (Gerät inkl. Versand und Zoll) aus den USA auch in Deutschland bestellen [12]. Diese Version liegt auch zum Test vor.



Das Mozilla Flame von vorne. 

Der Artikel behandelt hauptsächlich das vorinstallierte Firefox OS 1.3 und stellt dabei gleichzeitig ein Update des Artikels „Firefox OS“ aus freiesMagazin 08/2013 [13] dar.

### Hardware

Mit einer Höhe von 132 mm und einer Breite von 68 mm ist das Flame ähnlich groß wie andere, aktuelle Vertreter auf dem Smartphone-Markt. Da das Flame als Ersatz für ein älteres Handy gedacht war, macht sich die Größe aber schon bemerkbar. Nicht in jeder Hosentasche findet das

Flame Platz bzw. man kann es zwar einstecken, sollte danach aber hektische Bewegungen vermeiden.

Das Gewicht ist mit ca. 120 Gramm ganz okay und das Gerät liegt gut in der Hand, was auch an der aufgerauhten Rückschale liegt. Diese lässt sich abnehmen, wobei man hierbei etwas Vorsicht walten lassen sollte, da sie aus sehr dünnem Plastik ist und leicht brechen könnte.

Unter der Haube, also der Rückschale, befindet sich der herausnehmbare Akku, ein Slot für eine Micro-SD-Karte und gleich zwei SIM-Karteneinschübe. Das Flame ist also auch gut für Leute geeignet, die sich oft in zwei verschiedenen Netzen aufhalten und nicht ständig die SIM-Karte tauschen wollen. Der Micro-SD-Slot ist glücklicherweise so angebracht, dass man eine Karte auch einsetzen kann, ohne den Akku entfernen zu müssen (was nicht zwingend zum Standard gehört).



Das Innenleben des Mozilla Flame. 

An der Rückseite des Gerätes befindet sich eine 5-Megapixel-Kamera mit Flash. Auf der Vorderseite ist eine kleinere 2-Megapixel-Kamera angebracht. Ansonsten hat das Flame nur noch zwei Knöpfe: den Ein-/Ausschalter an der Oberseite und den Lautstärkeregler an der rechten Seite. Als Anschlüsse gibt es einen Kopfhöreranschluss (3,5 mm Klinke) an der Oberseite und einen Micro-USB-Anschluss Typ B an der Unterseite. Der USB-Anschluss ist auch zum Laden des Flames gedacht, da kein separates Netzteil mitgeliefert wird.

Die Vorderseite des Smartphones nimmt hauptsächlich der 4,5 Zoll große Touchscreen ein. Darüber ist die Frontkamera und der orangefarbene Lautsprecher, darunter der Hauptmenü-Knopf.

Im Inneren des Flame ist eine MSM8210-Dual-Core-CPU mit 1,2 GHz. 1 GB RAM Speicher dient für Anwendungen. Zum Speichern von Daten sind 8 GB bereits vorinstalliert, wobei man durch eine Micro-SD-Karte den Speicher erweitern kann. Bluetooth, WLAN und GPS gehören ebenfalls zum Standard.

## Software im Überblick

Wenn man das Mozilla Flame einschaltet, begrüßt einen ein Fuchs, dessen Schwanz wie eine Flamme geformt ist und dem Smartphone auch seinen Namen gibt. Beim erstmaligen Einschalten wird nach der Sprache gefragt, wobei die meisten europäischen Sprachen sowie traditionelles Chinesisch angeboten wird. Als Standardsystem des Flame kommt Firefox OS 1.3 zum Einsatz.



Das Mozilla Flame von hinten. 🔍

Es gibt effektiv zwei Hauptmenü-Bildschirme, zwischen denen man durch seitliches Wischen wechseln kann. Auf dem ersten Bildschirm ist eine Art Suchfenster für installierte Anwendungen zu finden sowie die Kategorien „Sozial“, „Spiel“, „Musik“ und „Werkzeuge“. Die Kategorien dienen zum leichteren Wiederfinden von installierten Apps sowie zum direkten Zugriff auf installierbare Apps aus dem Marketplace, wenn man mit dem Internet verbunden ist.

Auf der zweiten Bildschirmseite befinden sich die Hauptanwendungen, die vorinstalliert sind, darunter Musik, Galerie, Radio, Kalender, Uhr, Musik, Video, E-Mail und die Einstellungen. Der Browser fehlt in der Liste, da es am unteren Rand noch einmal vier Symbole für den Schnellstart gibt. Dort sind die Telefon-App, Kontakte, der Browser und die SMS-App zu finden.

Per Standard öffnet sich eine App oder eine Kategorie durch das einzelne Antippen. Wenn man ein Symbol länger gedrückt hält, kann man es verschieben oder vom Bildschirm entfernen. Das Entfernen geht dabei aber nur für Kategorien und nachinstallierte Apps. Die vorinstallierten Hauptanwendungen lassen sich nicht entfernen. Durch Drag-and-drop lässt sich ein bestehendes Symbol auch eine Kategorie hinzufügen.

Hält man den Finger länger auf den Hintergrund, kann man das Hintergrundbild ändern oder neue Kategorien hinzufügen.

Etwas ungewohnt mag einem die Navigation zwischen den Apps vorkommen. Die meisten Anwendungen kann man nicht beenden, sondern mittels des Hauptmenü-Knopfs unter dem Bildschirm kommt man immer ins Hauptmenü zurück, die Applikation bleibt im Hintergrund geöffnet. Anwendungen schließen und zu anderen Anwendungen wechseln kann man, indem man längere Zeit den Hauptmenüknopf gedrückt hält. Dann kann man

durch die offenen Applikationen wischen bzw. diese dort auch ganz schließen.



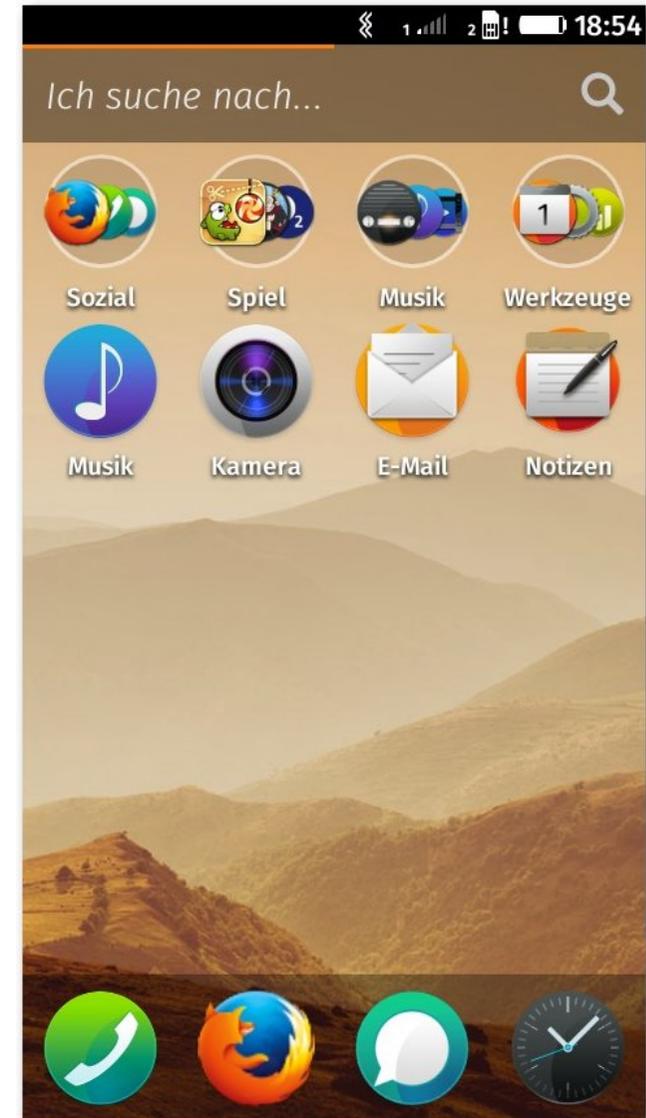
Start von Firefox OS. 

In Firefox OS 1.3 gibt es im Entwicklermenü (siehe unten) eine Option, durch seitliches Wischen zwischen den offenen Apps zu wechseln. In Firefox OS 2.0 ist diese Option sogar Standard.

### Kommunikation

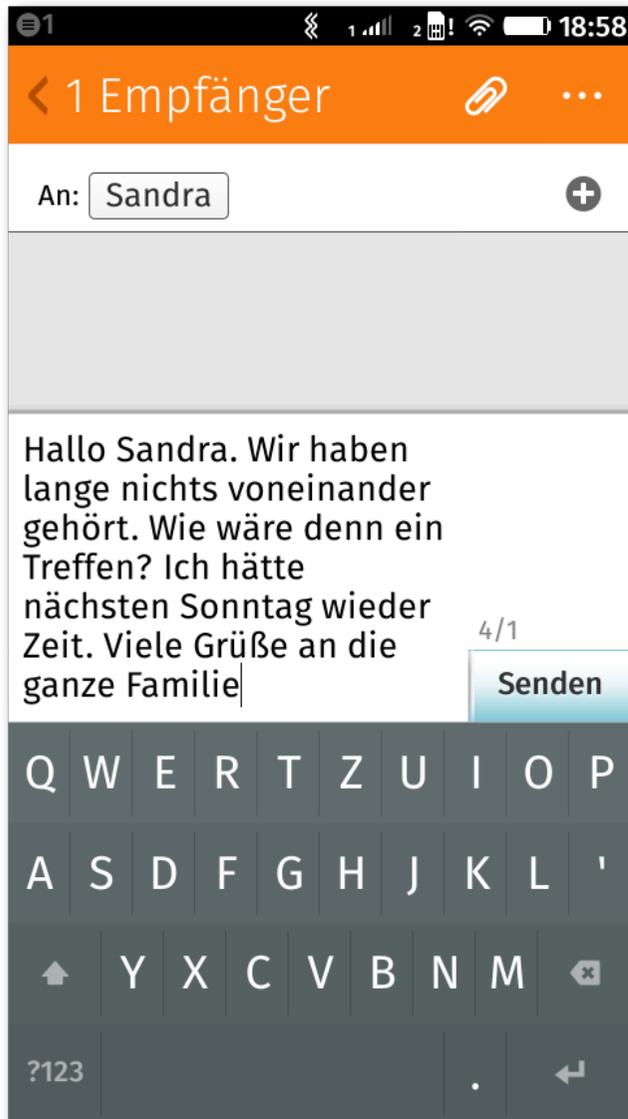
Das Wichtigste bei einem Smartphone sind sicherlich die Kommunikationsfähigkeiten. Auch wenn heute schon Witze darüber gemacht werden, dass viele ein Smartphone für alles nutzen, nur nicht zum Telefonieren, kann das Flame natürlich auch das. Das Wählen per Kontaktliste geht intuitiv und funktioniert wie gewünscht. Etwas schlecht ist, dass man beim Halten des Telefons am Ohr sehr leicht mit der Wange auf das Display und die Bildschirmtastatur kommt, wodurch beim Gegenüber ein nerviges Tonkonzert entsteht. Von der Sprachqualität ist das Telefon okay, es gab im Test aber ab und zu Tonaussetzer, wobei nicht klar auszumachen war, auf welcher Seite der Verbindung das Problem lag.

Die E-Mail-Verwaltung unter Firefox OS funktioniert auch mit mehreren E-Mail-Konten per IMAP ganz gut. Um Speicher zu sparen, lädt die E-Mail-App allerdings nicht alle Daten in einem IMAP-Ordner herunter. Konkret bedeutet das, dass, wenn man eine alte E-Mail sucht, man erst manuell bis ganz ans Ende der Liste scrollen und dort dann „*Weitere Nachrichten vom Server*“ auswählen muss. Das lädt dann die nächsten 20 E-Mails herunter. Nutzt man allein das Smartphone für E-Mail-Kommunikation, ist das aber gegebenenfalls kein Problem.



Das Hauptmenü. 

Für weitere Kommunikationswege stehen Kurznachrichten per SMS oder MMS zur Verfügung.



Kommunikation per SMS. 🔍

Diese funktionieren wie gewünscht, einzig die Kommunikationsanzeige ist nicht optimal, da zwar

der Kommunikationsstrang angezeigt wird, aber leider ohne Uhrzeit. Nur bei einer SMS, die an einem anderen Tag ankommt, wird Datum und Uhrzeit eingeblendet. Dadurch ist nicht mehr sichtbar, wann genau man eine Nachricht erhalten oder gesendet hat. Neue Nachrichten und Ereignisse werden in der oberen linken Seite angezeigt bzw. wenn das Telefon gesperrt ist, direkt auf dem Sperrbildschirm, was sehr praktisch ist.

Für alles Weitere, wie z. B. Facebook, Google+, Twitter und Co. oder auch Whatsapp stehen im Marketplace passende Apps zur Verfügung.

### Surfen

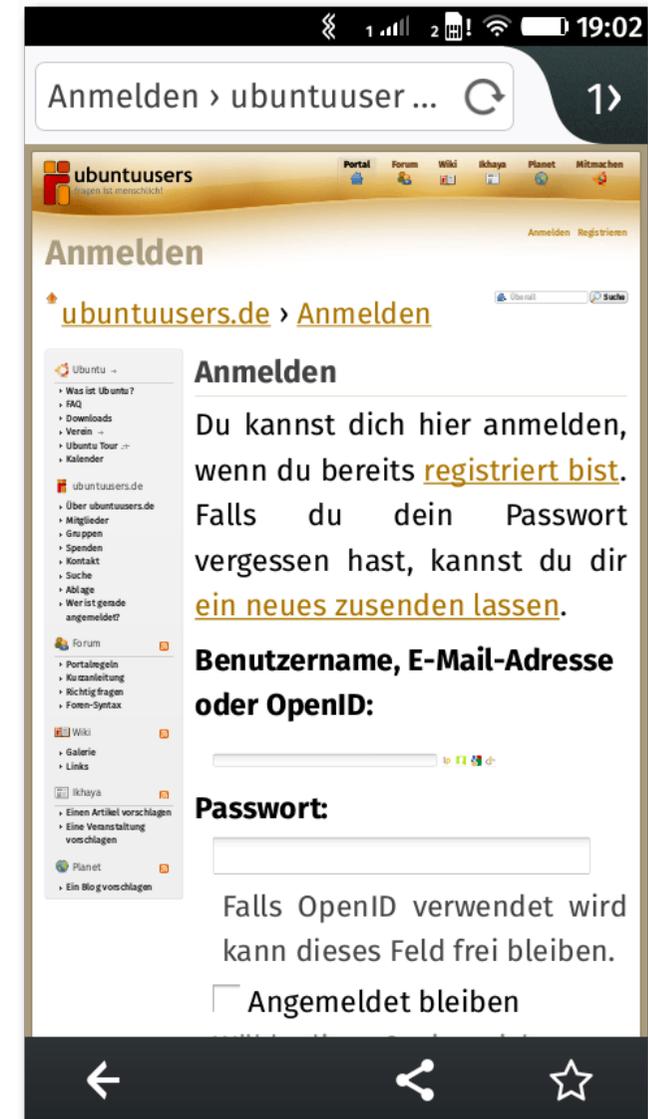
Das Zweitwichtigste an einem Smartphone ist sicherlich der Weg ins Internet. Und natürlich wäre es seltsam, wenn Mozilla für sein Firefox OS einen anderen Browser als Firefox benutzt hätte.

Größtenteils funktioniert der Browser intuitiv. Etwas ungewohnt ist, dass die URL-Eingabeliste verschwindet, wenn man auf einer Seite nach unten scrollt. Die Liste erscheint auch erst wieder, wenn man an den Beginn der Seite geht.

Unklar ist auch, wieso Webseiten (ohne Browserweiche) doch sehr anders als auf dem Desktop aussehen. V. a. Eingabefelder sind oft winzig im Vergleich zum restlichen Text, sodass das Antippen dieser ohne Zoom fast unmöglich ist.

Neben Lesezeichen und Tabs bietet der Browser nichts. Einstellungen gibt es keine, so kann

man z. B. nicht regulieren, welche Seiten Cookies setzen dürfen oder wo JavaScript aktiv sein soll.



Im Browser wird nicht alles passend angezeigt. 🔍

## Weitere Anwendungen

Eine weitere wichtige Anwendung ist sicherlich der Wecker in der Uhr-App, mit dem man mehrere Alarme definieren kann, um beispielsweise am Wochenende zu einer anderen Uhrzeit geweckt zu werden. Daneben enthält die App auch noch eine Stoppuhr und einen Countdown-Zähler.

Natürlich kann man mit dem Flame auch Musik hören. Neben dem UKW-Radio, für das zwingend als Empfänger ein Kopfhörer angeschlossen sein muss, kann die Musik-App verschiedene Musiktypen abspielen. MP3 ist selbstverständlich Standard, aber auch der freie Standard OGG [14] wird unterstützt. Die Musiktitel kann man entweder als Übersicht oder nach Titel, Interpret oder Album sortiert anzeigen lassen. Es gibt zusätzlich verschiedene Wiedergabelisten, die beispielsweise die am wenigsten gehörten oder die neuesten Titel abspielt. Die Klangqualität über Kopfhörer ist recht gut, einen Equalizer gibt es leider nicht. Die Ausgabe über den Telefonlautsprecher zur Raumbeschallung sollte man aber sein lassen.

Ein Problem beim Musik hören ist die Lautstärke-Einstellung. Das Flame hat für verschiedene Szenarien verschiedene Lautstärkeprofile. So lässt sich die Lautstärke des Klingelns bei einem Anruf separat zur Lautstärke beim Telefonieren einstellen. Ebenso kann man auch die Musiklautstärke separat einstellen. Hierbei gibt es aber einen Bug, sodass die Musik-App beim ersten Start die Lautstärke des Klingelns übernimmt. Wenn dieser aber – wie im Test – auf lautlos steht und nur Vi-

bration an ist, hört man nichts und muss manuell erst die Lautstärke nach oben drehen.



Die Musik-App mit der Übersicht aller Alben. 

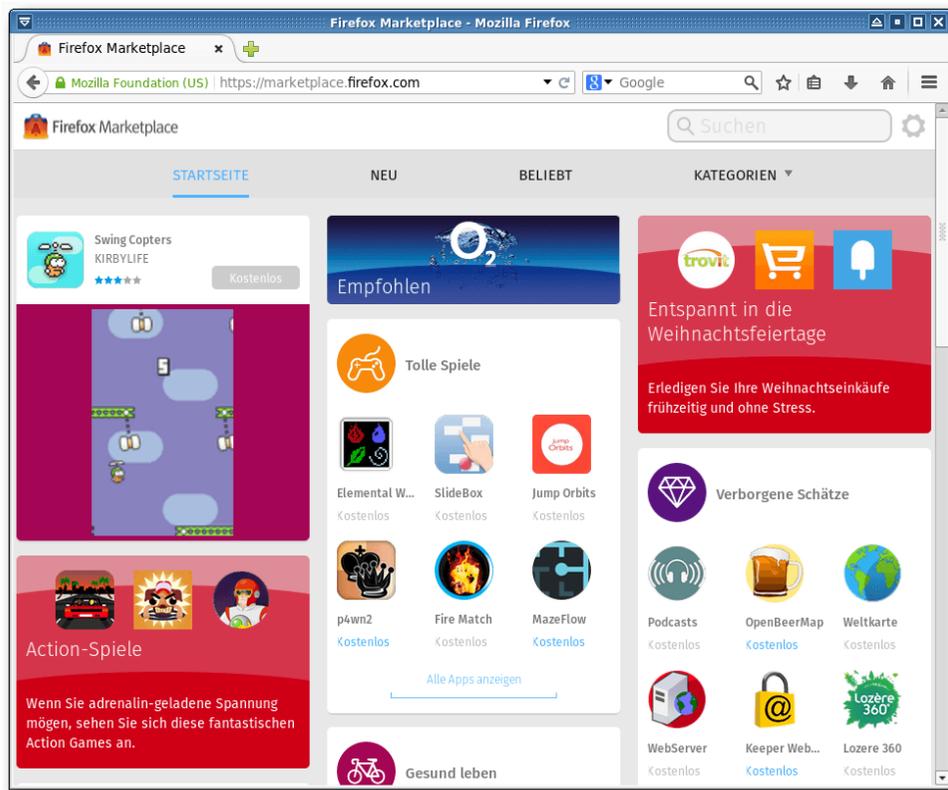
Weiterhin oft genutzt ist die Bildschirmtastatur. Diese funktioniert ganz normal, stellt aber per Standard keine Umlaute dar. Wer es nicht weiß, sucht ggf. etwas länger, aber durch das lange Festhalten eines Buchstabens wird ein Overlay angezeigt, worüber man Sonderzeichen auswählen kann. Hiervon stehen zahlreiche zur Verfügung. Eine Wisch-Technik wie Swype [15] steht per Standard nicht zur Verfügung.

Wie bei der Hardware schon erwähnt, hat das Mozilla Flame zwei Kameras. Beide liefern für ihre Verhältnisse gute Bilder. Sehr praktisch ist, dass die Kamera-App auch gleich noch einige Bildbearbeitungswerkzeuge mitbringt. So kann man Bilder beschneiden, die Helligkeit ändern, Farbräume wählen oder eine Auto-Verbesserung aktivieren.

## Entwickleroptionen

Da es sich beim Mozilla-Flame um ein Entwicklerhandy handelt und auch Firefox OS noch nicht fertig entwickelt ist, gibt es etwas versteckt in den Einstellungen unter „Geräteinformationen → Weitere Informationen → Entwickler“ ein spezielles Entwicklermenü, in dem man verschiedene Hilfoptionen wählen kann.

So ist in Firefox 1.3 dort eine Option „Randgesetzen“ zu finden, mit der man nach der Aktivierung zwischen offenen Apps per seitlicher Wischgeste wechseln kann. Daneben gibt es auch noch die wichtige Option „Externes Debugging“, mithilfe derer es möglich ist, das Telefon manuell zu aktualisieren.



Der Firefox Marketplace im Browser. 🔍

## Apps und Marketplace

Was wäre ein Smartphone ohne Apps? Richtig, ein normales Telefon, das man aber ebenso normal nutzen kann, um die notwendigsten Dinge des Alltags damit zu erledigen. Dennoch reichen die vorinstallierten Anwendungen sicherlich nicht allen Nutzern aus, weswegen Firefox einen eigenen App-Marketplace hat [16].

Eine Vielzahl der Anwendungen sind – wie wohl bei anderen Smartphones auch – Spiele. Aber es

Sehr gut ist, dass man für jede App die Berechtigungen einzeln einstellen kann. Zumindest im Test gab es als Einstellung aber nur, ob eine Anwendung den Standort erfahren darf. Die meisten Apps benötigen glücklicherweise keinen besonderen Zugriff.

Was für viele Flame-Käufer beziehungsweise Firefox-OS-Nutzer ein großes Problem darstellen könnte, ist die Inkompatibilität zu Android-Apps.

gibt auch einige sinnvolle Anwendungen wie beispielsweise Whatsapp-Clients, Übersetzungen, Navigation, Taschenrechner etc. Aber der Großteil sind wirklich Spiele.

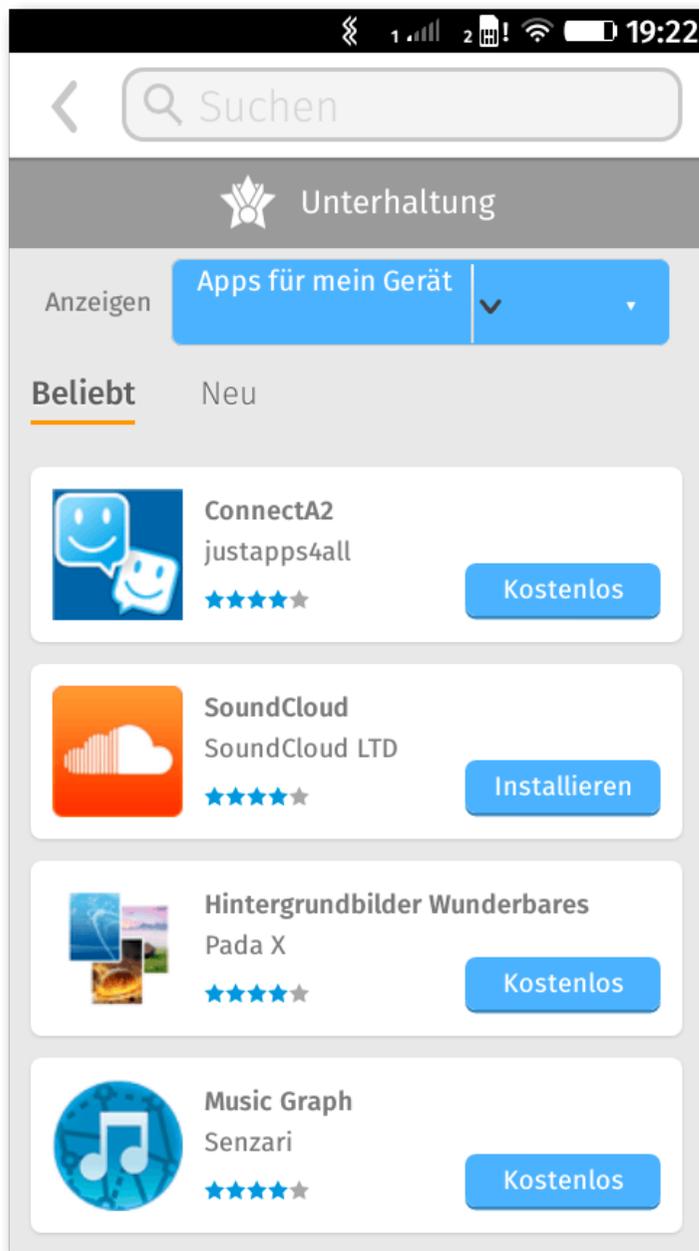
Bei vielen Apps wird angegeben, dass diese offline funktionieren, was leider so gut wie nie den Tatsachen entspricht. Von willkürlich 10 getesteten Apps, die angeblich offline funktionieren sollten, meldeten 8 nach der Installation beim Start, dass sie ohne eine WLAN-Verbindung nicht starten können.

Viele, vor allem proprietäre Anwendungen gibt es aufgrund der Marktführerschaft lediglich für Android und für iOS. Firefox-OS-Nutzer sind dann außen vor und können die Anwendungen überhaupt nicht nutzen. Jolla hat es hier mit seinem Sailfish OS cleverer gemacht und kann auch Android-Apps ausführen [17]. Was das für die Zukunft heißt, ist aktuell noch unklar. Einen schwereren Stand hat Firefox OS damit aber natürlich, weil sich viele App-Entwickler überlegen werden, ob sie auch noch für eine dritte Plattform entwickeln wollen.

Eine Anekdote zum Schluss: Dass es nicht alles in App-Form gibt, zeigt eine Suche nach einer Screenshot-App im Firefox Marketplace. Leider findet man dort keine und für den Artikel wurde die Digitalkamera schon bereit gelegt. Eine Suche im Netz war dann aber erfolgreich [18], wobei das gleichzeitige Drücken von Hauptmenü- und Ausschalt-Knopf nicht zu den intuitivsten Dingen gehört, die sich die Firefox-OS-Entwickler ausgedacht haben.

## Firefox-OS-Updates

Viele Android-Anwender kennen das Problem mit den ausbleibenden Updates [19] nur zu gut. Sehr oft stehen von den Handy-Herstellern keine zur Verfügung. Das Flame scheint da leider keine Ausnahme zu sein, obwohl man mit Mozilla doch direkt an der Quelle sitzt. Auch eine manuelle Suche nach Updates teilte einem bis Anfang November mit, dass es keine neuen Updates für Firefox 1.3 gibt.



Die Firefox Marketplace App. 

Am 10. November 2014 konnten Flame-Nutzer sich aber freuen, da Mozilla das erste FOTA-Update (Firmware over the air) veröffentlichte [20]. Damit wurde in zwei Update-Paketen Version 2.0 von Firefox OS installiert (Erfahrungsbericht siehe weiter unten).

Auf der Mozilla-Flame-Seite [11] findet man ausführliche Informationen (auf Englisch), wie man das Entwicklerhandy auf einen neuen Stand heben kann. Dies ist im Prinzip sehr simpel und innerhalb von 10 Minuten abgeschlossen inkl. dem Sichern und Zurückspielen der eigenen Daten wie Adressdaten, Kommunikationsprotokollen oder E-Mail-Einstellungen.

Insgesamt gibt es aber für Firefox 1.3 und auch für die Version 2.1 und 2.2 keine weiteren automatischen Systemupdates. Das heißt, wer Firefox OS 2.2 testen will, muss sich dieses manuell installieren, was etwas komplizierter ist, weil es hierfür kein fertiges Image gibt. Die manuelle Erstellung eines eigenen aktuellen Images wurde aufgrund von Zeitmangel (der Download der benötigten Dateien war auch nach 4 Stunden noch nicht abgeschlossen) abgebrochen.

### Erfahrungsbericht

Nach vier Monaten im täglichen Einsatz hat sich das Mozilla Flame zum größten Teil bewährt. Telefonieren, SMS schreiben, Musik

hören und ab und zu ein Foto machen funktioniert ohne Probleme. Sicherlich klappt mitunter etwas nicht auf Anhieb, was mit Software-Version 1.3 aber auch zu erwarten ist. So scrollt beim Schreiben einer SMS das Eingabefenster bei einem Zeilenumbruch nicht immer mit, was man manuell justieren muss. Das nervt zwar etwas, ist aber auszuhalten.

Etwas negativer sind die Neustarts, die Firefox OS des Öfteren plagen. Dies ist dabei ein bekanntes Problem bzw. ein Feature [21]. Damit man das Gerät aus jeder Notsituation neu starten kann, kann man den PowerOn-Knopf an der Oberseite für längere Zeit (5 Sekunden) drücken und das Flame startet automatisch neu. Hat man das Smartphone in einer engeren Hose, passiert es leider sehr häufig, dass der Knopf im Sitzen für die Zeit gedrückt wird. Immerhin wird man durch ein Vibrationssignal „informiert“, dass man seine PIN eingeben sollte, um wieder erreichbar zu sein. Noch blöder ist, dass sich bei dem Neustart manchmal Datum und Uhrzeit auf den Zeitpunkt der letzten Einstellung zurückstellt. Wenn man das nicht mitbekommt, kann es sein, dass der Wecker nicht mehr ganz pünktlich klingelt.

Ab und an spinnt die Software auch komplett. So kam es auch schon vor, dass wie von Geisterhand irgendwelche Tasten beim Wählen einer Nummer erkannt wurden. Selbst als man nicht mehr getippt hat, erschienen neue Zahlen. Ebenso scheint die Tastatursperre nicht immer zu funktionieren, sodass im Telefonbuch bei den ausgehenden Ge-

sprächen auch mal Rufnummern wie 004484 oder 114 (was glücklicherweise knapp vorbei war) erscheinen.

Sehr gut bzw. nicht so schlimm wie befürchtet ist die Akku-Leistung. Man hört aus dem Netz und aus dem Bekanntenkreis, dass deren Smartphones jeden Tag ans Netz zum Laden müssen. Würde man das Flame „normal“ benutzen, träfe das sicherlich auch hier zu. Wenn man das Gerät aber als Telefon nutzt, d. h. ab und an damit telefoniert, eine SMS schreibt oder Musik hört, hält der Akku zwischen fünf und sieben Tagen, was eine gute Leistung ist. Natürlich geht dies auch anders: Wenn man ständig das WLAN oder eine Internet-Verbindung aktiv hat, irgendwelche Spiele damit spielt oder 24 Stunden lang chattet, hält der Akku auch keine zwei Tage durch.

### Erfahrungen mit Firefox OS 2.0

Wie oben berichtet gab es Anfang November ein offizielles Update auf die neue Firefox-OS-Version 2.0. Dabei handelt es sich aber nicht um die neueste Entwicklerversion, die die Zahl 2.2 trägt. Das Update in Form von Base-Image v1.88 stand dabei aber bereits einige Wochen zuvor auf der Update-Webseite [11] zur Verfügung und konnte manuell eingespielt werden.

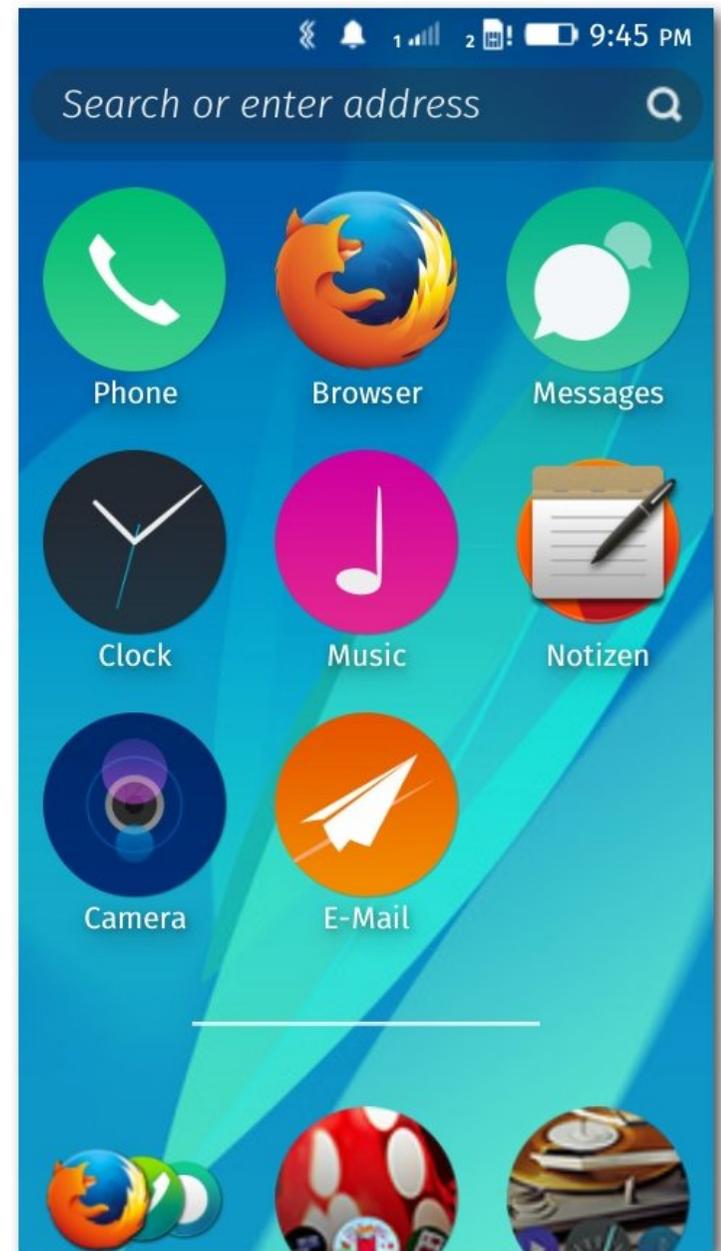
Beide Updates, sowohl das manuelle Flashen als auch über die Systemupdates, klappten recht reibungslos. Das Flash-Skript im Base-Image v1.88 hat nur leider einen kleinen Fehler, sodass es ggf. zu einer Fehlermeldung beim Updaten kommt,

was aber nicht weiter tragisch ist. Man muss es eben nur wissen.

Die Hoffnung bei einem Update ist natürlich, dass einige der Fehler, die Version 1.3 noch hat, ausgemerzt sind. Leider ist dem nicht so und man hat eher das Gefühl, dass nur mehr neue Bugs hinzugefügt wurden.

Zuerst fällt auf, dass nach dem Update die Sprache des Systems plötzlich Englisch ist bzw. kann man bei einem manuellen Update und dem Neueinrichten nur Englisch, Französisch, traditionelles Chinesisch und eine arabische Sprache auswählen. Es ist sehr ärgerlich, dass die Mozilla-Entwickler viele europäische Tester vor den Kopf stoßen.

Das Hauptmenü wurde in Firefox OS 2.0 komplett überarbeitet. Natürlich ist so etwas sehr oft Geschmackssache, aber aus funktionaler Sicht ist es keine Verbesserung. Die untere Schnellstartleiste mit den wichtigsten Funktionen ist verschwunden. Es gibt auch keine Mehrfach-Menübildschirme mehr, sondern diese sind untereinander angeordnet und durch Striche getrennt. Vorher konnte man also durch ein einzelnes seitliches Wischen zu allen vorinstallierten Apps kommen. Mit dem Update auf 2.0 muss man sehr weit nach unten scrollen, um alle Apps zu erreichen, was u. a. auch daran liegt, dass die Symbole größer geworden sind und mehr Platz wegnehmen.



Das Hauptmenü in Firefox 2.0. 

Von den oben genannten Fehler wurde keiner behoben. Der Browser zeigt zwar nun viele Seiten korrekt an, aber die Textgröße variiert immer noch sehr stark. Daneben sind neue Bugs hinzugekommen:

1. Die Musik-App war nach einem Update unbrauchbar, weil sie ewig nach Musik suchte, obwohl gar keine auf Gerät gespeichert war. Nur durch ein erneutes Update konnte das behoben werden.
2. Die Benachrichtigungstexte verschwinden nicht immer, wenn man auf sie klickt, wie das zuvor der Fall war. Man muss manuell noch auf „Alle löschen“ klicken.
3. Der SMS-Versand hängt leider etwas. Nach dem Senden wird die Nachricht nicht im Dialog angezeigt, sodass man denkt, dass sie nicht gesendet wurde. Verlässt man aber den Gesprächsverlauf und betritt ihn neu, sieht man die Nachricht.

Das größte Ärgernis ist aber die Akku-Leistung. Wo man das Flame in Version 1.3 noch loben kann, weil der Akku bei wenig Gebrauch fünf Tage hält, schafft der Akku in Version 2.0 keine 20 Stunden mehr. Das ist nicht hinnehmbar, weil man das Smartphone so nicht einmal länger als 24 Stunden vom Netz getrennt halten kann.

## Fazit

Alles in allem ist das Gerät sehr brauchbar, auch wenn es hie und da noch etwas hängt. Neben den oben genannten Fehlern funktioniert wie erwähnt

auch die Lautstärkeregelung nicht wie erwartet, einmal sind nach Erstellen eines Screenshots alle Hintergrundbilder verschwunden und die WLAN-Aktivierung hängt manchmal. Davon abgesehen gab es aber bisher keinen einzigen Datenverlust.

Sehr negativ ist die Update-Politik von Mozilla. Auch wenn Firefox OS noch nicht ausgereift ist, kann es nicht sein, dass nach einem Update die Spracheinstellungen weg sind und das Smartphone keinen Tag mehr ohne Stromanschluss leben kann. Dies ist auch der Grund, wieso nach dem Update auf Version 2.0 nach einem Tag Test sofort wieder Version 1.3 in Form des Base-Image v1.23 eingespielt wurde, was glücklicherweise auf der Update-Webseite [11] noch zur Verfügung steht. Version 2.0 ist für den täglichen Gebrauch, vor allem für deutschsprachige Nutzer, unbrauchbar.

Mit Version 1.3 kann man aber gut leben und das Smartphone funktioniert größtenteils so, wie es soll. Der Kauf wurde jedenfalls noch nicht bereut.

## LINKS

- [1] <http://www.7mobile.de/handy-news/marktanteile-der-hersteller-und-betriebs-systeme-samsung-und-android-topform.htm>
- [2] <http://www.android.com/> 
- [3] <https://www.mozilla.org/de/firefox/os/>
- [4] <https://sailfishos.org/> 
- [5] <https://www.tizen.org/> 
- [6] <http://www.ubuntu.com/phone> 
- [7] [https://developer.mozilla.org/en-US/Firefox\\_OS/Developer\\_phone\\_guide](https://developer.mozilla.org/en-US/Firefox_OS/Developer_phone_guide) 

- [8] <http://www.geekspone.com/> 
- [9] [http://www.ztedevices.com/product/smart\\_phone/index\\_1.html](http://www.ztedevices.com/product/smart_phone/index_1.html) 
- [10] <https://www.mozilla.org/de/firefox/os/devices/>
- [11] [https://developer.mozilla.org/en-US/Firefox\\_OS/Developer\\_phone\\_guide/Flame](https://developer.mozilla.org/en-US/Firefox_OS/Developer_phone_guide/Flame) 
- [12] <http://www.everbuying.com/product549652.html> 
- [13] <http://www.freiesmagazin.de/freiesMagazin-2013-08>
- [14] <https://de.wikipedia.org/wiki/Ogg>
- [15] <https://de.wikipedia.org/wiki/Swipe>
- [16] <https://marketplace.firefox.com/> 
- [17] <http://www.heise.de/open/meldung/Jolla-Sailfish-OS-ist-nun-Android-kompatibel-1958589.html>
- [18] <https://support.mozilla.org/en-US/kb/how-to-take-screenshots-firefox-os> 
- [19] <https://www.soeren-hentzschel.at/mozilla/firefox-os/2014/11/10/mozilla-verteilt-firefox-os-2-0-fota-fuer-t2mobile-flame/>
- [20] [https://bugzilla.mozilla.org/show\\_bug.cgi?id=1080386](https://bugzilla.mozilla.org/show_bug.cgi?id=1080386) 

### Autoreninformation

**Dominik Wagenführ** (Webseite) hat sich das Flame gekauft, weil das System darauf nicht Android ist und er volle Kontrolle über die Software haben kann.



Teilen



Kommentieren

## Rezension: Programmieren lernen mit Python von Jochen Schnelle

**P**ython [1] gilt als besonders einsteigerfreundliche Programmiersprache. Das vorliegende Buch „Programmieren lernen mit Python“ [2] macht im Titel schon klar, was das Anliegen ist: Den Einsteiger in die Programmierung mit Python begleiten und erklären.

**Redaktioneller Hinweis:** Wir danken O'Reilly für die Bereitstellung eines Rezensionsexemplars.

Der Autor Allen B. Downey ist auch als Dozent für (Einsteiger-) Programmierkurse tätig. Und so wird in der Einleitung erklärt, ist auch dieses Buch entstanden: aus den Vorlagen für und den Erfahrungen aus diesen Kursen. Diese hat der Autor auf knapp 300 Seiten im Buch festgehalten.

Das Buch liegt seit Sommer 2014 in der zweiten, aktualisierten Auflage vor, auf die sich auch diese Rezension bezieht. Das Buch behandelt primär Python 3, auf Unterschiede zu Python 2.7 wird wo immer nötig hingewiesen.

### Fehlstart

Nach ein paar allgemeinen erklärenden Seiten zum Thema Programmierung geht es mit den ersten Python-Beispielen los. Nur: eine Erklärung, wie man Python überhaupt installiert, fehlt komplett. Es ist noch nicht einmal ein Hinweis auf die Downloadseite für Python zu finden. Für ein

Einsteigerbuch ist das unverständlich, ein echter Fehlstart.

### Schrittweise

Danach geht es dann schrittweise durch die Grundlagen der Programmiersprache: Variablen, Ausdrücke, Funktionen, Iteration, Strings, Listen, Dictionaries sowie Klassen und Objektorientierung. Alle Themen werden recht ausführlich anhand von Beispielen erklärt, ohne jedoch ins Langatmige abzudriften.

Das Buch enthält sehr viele Beispiele und kurze, vollständig abgedruckte Listings, die das jeweilige Thema anhand von Python-Code zeigen und erklären.

### Selbst ist der Programmierer!

Ganz am Ende jedes Kapitels findet man eine Reihe von Aufgaben, die das zuvor Erlernte vertiefen und zum Selbststudium animieren sollen. Der Zeitbedarf und die Schwierigkeit der Aufgaben variiert dabei von kurz und einfach bis schwieriger und zeitaufwendiger. Für die komplexeren Aufgaben kann man die Lösung herunterladen.

### Grafisches

Das 19. und letzte reguläre Kapitel beschäftigt sich dann noch mit der Erstellung von grafischen Oberflächen mittels TKinter. Da dieses bei einer Python-Installation in der Regel direkt mit instal-

liert wird, ist dies der übliche Weg, den auch andere Python-Bücher gehen.

Unüblich ist hingegen, dass der Autor nicht das in Python enthaltene Modul für TKinter nutzt, sondern ein von ihm erstelltes mit dem (wenig kreativen) Namen „Gui“. Das vereinfacht die Sache insofern nicht, als das die Methoden dieses Moduls recht kryptische Namen haben, was der Verständlichkeit des Codes nicht zugute kommt.

Des Weiteren zieht der Autor in diesem Kapitel das Tempo drastisch an. Während die anderen Kapitel, wie bereits weiter oben erwähnt, recht gemächlich sind, geht es hier deutlich schneller voran. Einsteiger, die sich zum ersten Mal mit der Programmierung von grafischen Oberflächen beschäftigen, dürften hier Probleme haben, Schritt zu halten.

### Debugging

Viele Kapitel des Buchs haben auch einen Abschnitt namens „Debugging“. In diesem werden Tipps und Tricks rund um die Fehlersuche gegeben. Dies ist gerade für Einsteiger durchweg sehr hilfreich.

Unverständlich ist hingegen jedoch, weshalb an keiner Stelle im Buch zumindest ein kleiner Hinweis auf die in Python enthaltenen Module „doctest“, „unittest“ und „pdb“ (= Python Debugger)

gegeben wird, welche eben zum Debugging und Schreiben von Tests genutzt werden können.

## Auffälligkeiten

Das Buch besitzt aber noch mehr Auffälligkeiten: Zum Thema Schleifen fokussiert der Autor sehr stark auf die while-Schleife. Die eigentlich wesentlich häufiger anzutreffende for-Schleife wird eher nebenbei erwähnt. Was auch insofern ungewöhnlich ist, als Python stark auf Iteration setzt und hier eben oft for-Schleifen genutzt werden.

Apropos Iteration: Das Buch zeigt auch mehrere ausführlichen Beispiele zum Thema Rekursion. Letzteres gehört aber nicht zu den Stärken von Python, sondern eben die Iteration. Warum das Thema Rekursion trotzdem mehrfach aufgegriffen wird ist unklar.

Während das Debugging, wie oben erwähnt, gut und ausführlich erklärt wird, kommt das Thema Fehler abfangen (mittels „try ... except“) sehr kurz. Zwar wird es erwähnt, aber besonders das gezielte Abfangen von bestimmten Fehlern, welches in der Programmier-Praxis durchaus gängig ist, fristet im Buch kaum mehr als ein Schattendasein.

## Was fehlt?

Python folgt bekanntlich der „batteries included“ Philosophie, was so viel heißt, dass eine Python-Installation nicht nur den Sprachkern an sich enthält, sondern eine Vielzahl von weiteren Modulen mitbringt. Auch darauf findet man im Buch

keinen einzigen Hinweis, obwohl bei weiterführenden Programmen die Nutzung dieser Module durchaus normale und gängige Praxis ist.

## Fazit

Das Buch „Programmieren lernen mit Python“ bietet dem geeigneten Programmierneinsteiger einen Einstieg in den Sprachkern von Python. Didaktisch gesehen ist der Einstieg jedoch stark akademisch geprägt und von teilweise geringer Praxisnähe. Außerdem fehlen viele Themen oder zumindest Hinweise darauf, die andere Python-Bücher hingegen bieten.

Des Weiteren stellt auch das kostenlose, offizielle Python Tutorial [3] (eine deutsche Übersetzung gibt es im Internet auch) eine direkte, kostenlose und „pythonischere“ Alternative zu dem Buch dar.

Vor dem Kauf des Buchs sollte man auf jeden Fall beim Buchhändler seiner Wahl einen kritischen Blick in das Buch werfen, ob Umfang und Stil den eigenen Ansprüchen und Vorlieben entsprechen.

**Redaktioneller Hinweis:** Da es schade wäre, wenn das Buch bei Jochen Schnelle im Regal verstaubt, wird es verlost. Die Gewinnfrage lautet:

„Verbesserungen und Zukunftspläne von Python werden im „Python Enhancement Program“ (kurz: PEP) gesammelt. Jeder Vorschlag/Plan hat dabei eine Nummer. Welche PEP-Nummer trägt das Release-Schedule für Python 2.8?“

Die Antwort kann bis zum **11. Januar 2015, 23:59 Uhr** über die Kommentarfunktion oder per E-Mail an [redaktion@freiesMagazin.de](mailto:redaktion@freiesMagazin.de) geschickt werden. Die Kommentare werden bis zum Ende der Verlosung nicht freigeschaltet. Das Buch wird unter allen Einsendern, die die Frage richtig beantwortet konnten, verlost.

### Buchinformationen

<b>Titel</b>	Programmieren lernen mit Python [2]
<b>Autor</b>	Allen B. Downey
<b>Verlag</b>	O'Reilly
<b>Umfang</b>	297 Seiten
<b>ISBN</b>	978-3-95561-806-3
<b>Preis</b>	24,90 € (broschiert), 20,00 € (E-Book)

### LINKS

- [1] <http://www.python.org>   
 [2] <http://www.oreilly.de/catalog/thinkpython2ger/index.html>  
 [3] <https://docs.python.org/3.4/tutorial/index.html> 

### Autoreninformation

**Jochen Schnelle** ([Webseite](#)) programmiert seit vielen Jahren bevorzugt in der Programmiersprache Python.



Teilen



Kommentieren

## Rezension: Wireshark 101 - Einführung in die Protokollanalyse von Stefan Wichmann

**D**er Begriff Protokollanalyse hört sich trocken an. Wer sich jedoch im Netzwerkumfeld bewegt oder gar als Forensiker arbeitet, reibt sich bei dem Thema die Hände. Die kostenlos erhältliche Anwendung mit dem Namen Wireshark [1] unterstützt die Arbeit in herausragender Weise. Das Buch zum Programm ist Programm!

**Redaktioneller Hinweis:** *Wir danken dem mitp-Verlag für die Bereitstellung eines Rezensionsexemplares.*

### Inhalt und Aufteilung des Buches

Nach der Übersicht zu Wireshark und dem grundlegenden Verständnis der Arbeitsweise des Programms stellt Autorin Laura Chapell frühzeitig eine Beispielsitzung vor, um zügig in die Thematik einzutauchen. Entsprechend findet sich im Kapitel ein Einschub über die Verarbeitung von Datenverkehr, während Anpassungen erst in späteren Kapiteln beleuchtet werden. Themen zum vernünftigen Umgang, zur Konfiguration, Einfärbungen bedeutsamer Ergebnisse und die Aufbereitung der analysierten Daten sind sinnvoll dargestellt.

Wer sich um IT-Forensik kümmert, wird das Kapitel über die Rekonstruktion von Datenverkehr und über Kommandozeilenoptionen lieben und wer noch nicht so tief in dieses Thema eingetaucht ist, schlägt im gut gemachten Glossar nach.

Durch den Index ist schnell benötigtes Fachwissen auffindbar, sodass die jeweilige Aufgabe eines jeden Kapitels durch Nachlesen lösbar ist. Die Lösungen zu den Aufgaben werden mit Hardcopies verdeutlicht. Auch verweist die Autorin auf andere Analysetools.

### Zielgruppe

Das Buch richtet sich an angehende Netzwerkanalytiker und ist für alle interessant, die Netzwerke nicht nur einrichten, sondern auch verstehen wollen. Von daher ist es auch ein Buch für alle, die sich um Sicherheitsbelange oder gar für kriminaltechnische Untersuchungen (IT-Forensik) kümmern.

### Stil

Passend zum Thema wird der Leser gesiezt und durch Hardcopies und gekennzeichnete Elemente auch in schwierige Themen geleitet, sodass das Studieren des Buches Spaß macht.

### Verständlichkeit

Hardcopies sind gut erkennbar und mit eindeutigen Hervorhebungen versehen.

Kompakte Tipps und Überschriften selbst zu kurzen Abschnitten zeigen schnell und ohne Ausschweifungen die benötigten Informationen. Positiv zu vermerken ist, das Kapitel nicht aufgebläht werden, wenn es zu einem Thema nicht mehr zu sagen gibt.

Fachbegriffe werden sofort erläutert, zusätzlich gibt es ein Glossar am Ende des Buches.

### Umfang

Eindeutig wird die auf dem Buchrücken benannte Zielgruppe „Einsteiger der Netzwerkanalyse“ erreicht. Dies umfasst auch den Nutzen für Forensiker im Netzwerkumfeld. Entsprechend dem Ziel, die Handhabung von Wireshark vorzustellen, geht das Buch auf Besonderheiten, nämlich den Einsatz von Wireshark zur gerichtstauglichen Dokumentation, nicht ein. Von daher ist die Abgrenzung des Buches zu angrenzenden Themen an dieser Stelle zu scharf.

Durch den Index ist das Buch als Nachschlagewerk zu den vielfältigen Themen nutzbar.

### Qualität

Weder bei der Rechtschreibung noch beim Druck fielen Fehler auf. Die vielen Hardcopies sind zielführend, überladen das Buch jedoch nicht, sondern unterstützen den Leser. Die vielfältigen Hinweise und Tipps sind gut lesbar, obwohl sie nicht barrierefrei dargestellt sind. Sie zeigen eine wertvolle Querverbindung zu angrenzender Software auf. Kleine Überschriften auf jeder Buchseite helfen, schnell gesuchte Informationen zu finden.

Das Cover mit dem Titel „Haie im Licht“ wirkt eher wie ein Jugendroman und täuscht etwas über die Tiefe des Buches hinweg.

## Buchinformationen

<b>Titel</b>	Wireshark 101 - Einführung in die Protokollanalyse [2]
<b>Autor</b>	Laura Chapell
<b>Verlag</b>	mitp
<b>Umfang</b>	352 Seiten
<b>ISBN</b>	978-3-8266-9713-5
<b>Preis</b>	44,99 Euro

## Fazit

Die Zielgruppe ist eindeutig erreicht. Hier ist die sinnvolle Aufteilung des Buches, vielfältige Erläuterungen im Text und schlussendlich das Glossar als wertvolles Gimmick zu nennen. Das Buch ist komplett durchzuarbeiten, um Wireshark zu verstehen. Wer eine Lösung für eine Analyse benötigt, kommt naturgemäß um das ausführliche

Studium des verständlich geschriebenen Buches und um die zeitaufwändige Einarbeitung in das Programm nicht herum.

**Redaktioneller Hinweis:** Da es schade wäre, wenn das Buch bei Stefan Wichmann im Regal verstaubt, wird es verlost. Die Gewinnfrage lautet:

„Wie heißt der Vorläufer von Wireshark?“

Die Antwort kann bis zum **11. Januar 2014, 23:59 Uhr** über die Kommentarfunktion oder per E-Mail an [redaktion@freiesMagazin.de](mailto:redaktion@freiesMagazin.de) geschickt werden. Die Kommentare werden bis zum Ende der Verlosung nicht freigeschaltet. Das Buch wird unter allen Einsendern, die die Frage richtig beantworten konnten, verlost.

## LINKS

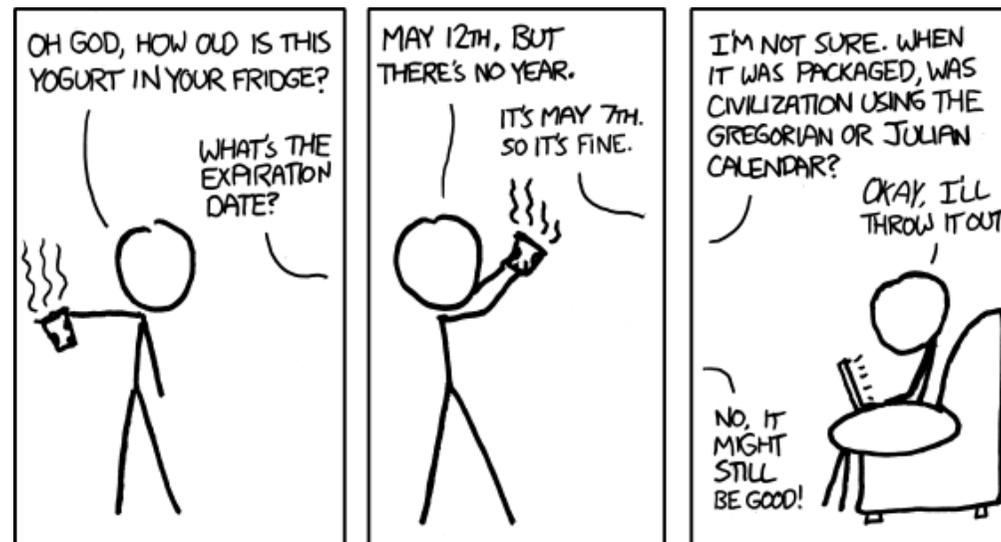
- [1] <https://www.wireshark.org/>   
 [2] <https://www.it-fachportal.de/shop/buch/Wireshark%C2%AE%20101/detail.html,b191208>

## Autoreninformation

**Stefan Wichmann** ([Webseite](#)) (Wic) interessiert sich für das Thema forensische Untersuchungsmethoden im IT-Umfeld und vertiefte seine Kenntnisse an der Fachhochschule der Polizei. Für Anregungen ist er im Blog erreichbar.

 [Teilen](#)

 [Kommentieren](#)



“Yogurt” © by Randall Munroe (CC-BY-NC-2.5), <http://xkcd.com/737/>

## Leserbriefe

**Für Leserbriefe steht unsere E-Mailadresse [redaktion@freiesMagazin.de](mailto:redaktion@freiesMagazin.de) zur Verfügung – wir freuen uns über Lob, Kritik und Anregungen zum Magazin.**

An dieser Stelle möchten wir alle Leser ausdrücklich ermuntern, uns auch zu schreiben, was nicht so gut gefällt. Wir bekommen sehr viel Lob (was uns natürlich freut), aber vor allem durch Kritik und neue Ideen können wir uns verbessern.

### Leserbriefe und Anmerkungen

#### Git-Tutorium – Teil 1

❖ Danke für das schöne Tutorial. Es hat mir großen Spaß gemacht, dem zu folgen und ich warte mit Spannung auf die nächste Folge.

**Maik** (Kommentar)

❖ Danke für den sehr klaren, anschaulichen Artikel. Ich würde gerne jetzt schon die Fortsetzung(en) lesen.

**Uwe** (Kommentar)

❖ Ich veröffentliche die fertigen Teile zuerst bei mir im Blog. Der zweite Teil ist auch fertig und veröffentlicht [1].

Schön, dass es dir gefällt!

**Sujeevan Vijayakumaran**

❖ Leider sind die Zeilenumbrueche in den Codeblöcken in diesem Artikel mehrmals ungluecklich. [...] Drei dieser vier Umbrueche sind unnoetiger-

weise schlecht. Die letzten zwei haetten ein Zeichen frueher sein sollen, der erste ein Zeichen spaeter.

Ist das denn ein technisches Problem im  $\LaTeX$ -Setup oder war der Setzer da nicht ganz aufmerksam?  
**meillo** (Kommentar)

❖ Die Codeblöcke werden automatisch von  $\LaTeX$  umgebrochen. Wir wollen da mit Absicht wenig eingreifen, weil das ein hoher Aufwand ist. Vor allem bei reinen Terminalausgaben achten wir so gut wir gar nicht auf die Umbrüche.

**Dominik Wagenführ**

#### Ubuntu und Kubuntu 14.10

❖ Danke für den interessanten Artikel. Wie doch die Zeit vergeht: Da habe ich noch auf die LTS-Version 14.04 gewartet und musste feststellen, dass diese in der LTS-Version 12.04 erst mit 14.04.1 automatisch als Upgrade angeboten wird. Was war das, etwa Juni.

Ich wollte nicht warten und habe eine meiner Kisten manuell zu einem Upgrade gebracht – es scheiterte und ich konnte das Problem nicht lösen, daher bin ich zurück auf die LTS 12.04: problemlos und es läuft einwandfrei.

Die 14.04.1 Version war gekommen und ich versuchte es mit einer anderen Kiste, von LTS 12.04 auf 14.04 zu upgraden – es scheiterte, die Kiste

konnte am Schluss nicht neu starten, sie läuft nun wieder unter 12.04.

Da gab es noch eine dritte Gelegenheit, noch einmal auf einer anderen Hardware, wo ebenfalls 12.04 klaglos läuft – 14.04 nicht.

Was mache ich falsch? Woran liegt es, dass Hardware die unter 12.04 einwandfrei erkannt wird und problemlos läuft mit 14.04 weder bei kompletter Neuinstallation noch bei Upgrades ein funktionierendes System ergeben? Ob das nicht auch einmal einen Bericht wert wäre?

Gerne würde ich auch einmal ein System als virtueller Desktop aufsetzen, so mit XEN oder VMWARE drunter. So dass man leicht mal von einem Server in Image starten kann, je nachdem was man da gerade so braucht (Office, Multimedia unter Windows oder Linux). Die privaten Daten sollten den Wechsel von Image zu Image weitgehend überstehen, ich möchte ja nicht immer wieder von Null beginnen. Eine Anleitung dazu wäre sicherlich noch interessant.

**Ice** (Kommentar)

❖ Ein Artikel hierzu ist schwer machbar. Es gibt keine allgemeinen Gründe, wieso etwas nicht mehr geht. Das hängt sehr speziell vom Einzelproblem ab. Zum Beispiel kann es an veralteten Treibern liegen, die nicht mehr angeboten werden. Aber es gibt da leider kein Allheilmittel, dass man

in einen Artikel gießen könnte, und dann funktioniert alles wieder. Hier ist eine Support-Anfrage bei [ubuntusers.de](http://ubuntusers.de) [2] eher angebracht.

Auf der Artikelwunschliste befindet sich bereits der Eintrag *Virtualisierung* [3]. Ich habe diesen um Xen und VMware ergänzt.

**Dominik Wagenführ**

### State of the Commons

Sehr schöne Zusammenfassung der Veröffentlichung. Die Mischung aus erklärendem Text plus der passenden Bilder finde ich sehr gelungen.

Danke. **txt.file** (Kommentar)

Da ja so wenige Leserbriefe kommen, hier mal ein nettes Dankeschön aus dem fernen Schöneberg als „nach Nikolaus“.

Mit welchem Programm bzw Plug-in wurde eigentlich das Original erstellt? **Gast** (Kommentar)

Das Original-Bild? Gute Frage. Ich hätte spontan gesagt mit Inkscape. **Dominik Wagenführ**

### Eine Einführung in Octave

Bitte mache weiter mit der Einführung in Octave. Mich interessieren insbesondere die Themen:

- Matrizen
- Grafische Anzeige
- Symbolische Berechnung

**Gast** (Kommentar)

Mehr ist in der Mache. Einfach überraschen lassen!  
**Jens Dörpinghaus**

### Lob

[Ich] freue mich immer wieder über Eure schönen Beiträge. Verlinke die auch gerne mal in einem Forum, um User zu Linux zu bewegen, wenn ich darf. Da waren die Win-Umstieg-Beiträge passend. Macht weiter so und danke schön.

**Michael**

Danke für dieses coole Magazin. Ich lese es echt gerne und sind immer gute Artikel dabei.

**Alex**

Ich weiß nicht recht, wie ich anfangen soll, das ist mein erster Leserbrief überhaupt. Tja, ich bin seit ungefähr 2006 Linuxuser, zuerst Dualboot und seit 2008 fast nur unter Linux, eigentlich der Distribution Ubuntu unterwegs. Mir gefällt der Gedanke von freier Software und freiem Wissen für alle Menschen auf diesem Planeten. Irgendwie bin ich ab und zu über Ubuntuusers.de auf Euer Magazin gestossen und habe die PDFs gelesen. Es sind immer viele interessante Artikel dabei. Manche kann man sofort gebrauchen, manche lese ich und sage „Aha“ weil es mich interessiert und viele die ich nur so durchlese um Neues oder Interessantes zu lesen.

Ich muss Euch hiermit ein großes Lob für Euer Engagement und Beharrlichkeit für die Erstellung und Veröffentlichung Eures Magazin aussprechen! Macht bitte weiter so, es wäre echt schade.

PS: Ich habe zu Weihnachten einen Tolino Vision 2 bekommen und bin gerade dabei, systematisch alle 2014er Ausgaben zu lesen. Danach werde ich danke Eures gut archivierten Inhalts mal freiesMagazin ganz von vorne zu lesen beginnen. Leider sind die PDFs auf dem Tolino nicht so berauschend zu lesen (getestet!), aber epub ist genial und super strukturiert.

In diesem Sinne danke und Euch Autoren und Verlegern ein Gutes Neues Jahr.

**Karl Frühwirt**

Ich wünsche allen ein frohes neues Jahr und macht weiter so.

**Dominic**

Vielen Dank für das Lob und die guten Wünsche. Wir versuchen, auch im Jahr 2015 allen Lesern von freiesMagazin gute und informative Artikel zukommen zu lassen.

**Dominik Wagenführ**

### LINKS

- [1] <http://svij.org/blog/2014/11/01/git-fur-einsteiger-teil-2/>
- [2] <http://forum.ubuntuusers.de/>
- [3] <http://www.freiesmagazin.de/artikelwuensche#sonstiges>

Die Redaktion behält sich vor, Leserbriefe gegebenenfalls zu kürzen. Redaktionelle Ergänzungen finden sich in eckigen Klammern.

 [Teilen](#)

 [Kommentieren](#)

## Veranstungskalender

### Messen

Veranstaltung	Ort	Datum	Eintritt	Link
Global Game Jam	Weltweit	23.01.–25.01.2015	frei	<a href="http://globalgamejam.org/">http://globalgamejam.org/</a>
FOSDEM	Brüssel	31.01.–01.02.2015	–	<a href="https://fosdem.org/">https://fosdem.org/</a>
Erlanger Linuxtag	Erlangen	28.02.2015	–	<a href="http://www.erlug.de/">http://www.erlug.de/</a>
FOSSGIS	Münster	11.03.–13.03.2015	frei	<a href="https://www.fossgis.de/">https://www.fossgis.de/</a>
Software Freedom Day	Weltweit	15.03.2015	frei	<a href="http://hfd.org/">http://hfd.org/</a>
Chemnitzer Linux-Tage	Chemnitz	21.03.–22.03.2015	8 EUR	<a href="http://chemnitzer.linux-tage.de/">http://chemnitzer.linux-tage.de/</a>
Augsburger Linux-Infotag	Augsburg	28.03.2015	frei	<a href="http://www.luga.de/">http://www.luga.de/</a>

(Alle Angaben ohne Gewähr!)

Sie kennen eine Linux-Messe, welche noch nicht auf der Liste zu finden ist? Dann schreiben Sie eine E-Mail mit den Informationen zu Datum und Ort an [redaktion@freiesMagazin.de](mailto:redaktion@freiesMagazin.de).

## Vorschau

freiesMagazin erscheint am ersten Sonntag eines Monats. Die Februar-Ausgabe wird voraussichtlich am 1. Februar u. a. mit folgenden Themen veröffentlicht:

- Bildbearbeitung mit GIMP – Teil 2
- Suricata
- Rezension: Praxisbuch Ubuntu Server 14.04 LTS

## Konventionen

An einigen Stellen benutzen wir Sonderzeichen mit einer bestimmten Bedeutung. Diese sind hier zusammengefasst:

- \$: Shell-Prompt
- #: Prompt einer Root-Shell – Ubuntu-Nutzer können hier auch einfach in einer normalen Shell ein **sudo** vor die Befehle setzen.
- ↵: Kennzeichnet einen aus satztechnischen Gründen eingefügten Zeilenumbruch, der nicht eingegeben werden soll.
- ~: Abkürzung für das eigene Benutzerverzeichnis **/home/BENUTZERNAME**
- : Kennzeichnet einen Link, der auf eine englischsprachige Seite führt.
- : Öffnet eine höher aufgelöste Version der Abbildung in einem Browserfenster.

# freiesMagazin-Index 2014

## A

### Android

- Rezension: Android – Einstieg in die Programmierung 09/2014
- Rezension: Android – kurz & gut 02/2014
- Rezension: Apps mit HTML5 und CSS3 für iPad, iPhone und Android 02/2014
- Rezension: Spieleprogrammierung mit Android Studio 10/2014

### Arduino

- Arduino 10/2014

## B

### Barrierefreiheit

- Browser: alle doof. – Eine Odyssee durch die Welt der Web-betrachter 09/2014

### Bildbearbeitung

- Äquivalente Windows-Programme unter Linux – Teil 3 01/2014
- Äquivalente Windows-Programme unter Linux – Teil 4: Bildbearbeitung (2) 02/2014

### Browser

- Browser: alle doof. – Eine Odyssee durch die Welt der Web-betrachter 09/2014
- Mit Tarnkappe im Netz – Das Tor Browser Bundle 05/2014
- Präsentationen mit HTML und reveal.js 03/2014
- Äquivalente Windows-Programme unter Linux – Teil 5: Internet-Programme (1) 04/2014
- Äquivalente Windows-Programme unter Linux – Teil 5: Internet-Programme (2) 06/2014

## Buch

- Im Test: PocketBook Touch 622 04/2014
- Rezension: 97 Things Every ... Should Know 08/2014
- Rezension: Android – Einstieg in die Programmierung 09/2014
- Rezension: Android – kurz & gut 02/2014
- Rezension: Apps mit HTML5 und CSS3 für iPad, iPhone und Android 02/2014
- Rezension: BeagleBone für Einsteiger 05/2014
- Rezension: C++: Das komplette Starterkit für den einfachen Einstieg 08/2014
- Rezension: Clean Coder 06/2014
- Rezension: Datendesign mit R 09/2014
- Rezension: Debian GNU/Linux – Das umfassende Handbuch 02/2014
- Rezension: Design Patterns mit Java 09/2014
- Rezension: Django Essentials 11/2014
- Rezension: Eclipse IDE 05/2014
- Rezension: Eclipse IDE – kurz & gut 01/2014
- Rezension: Geheimnisse eines JavaScript-Ninjas 07/2014
- Rezension: Git – kurz & gut 11/2014
- Rezension: Inkscape – Der Weg zur professionellen Vektorgrafik 07/2014
- Rezension: Java – Eine Einführung in die Programmierung 12/2014
- Rezension: JavaScript – Grundlagen, Programmierung, Praxis 05/2014
- Rezension: Kanban in der IT 04/2014
- Rezension: Kids programmieren 3D-Spiele mit JavaScript 08/2014
- Rezension: Linux-Kommandoreferenz 04/2014
- Rezension: Linux-Server mit Debian 7 GNU/Linux 11/2014
- Rezension: Raspberry Pi programmieren mit Python 04/2014
- Rezension: Schrödinger lernt HTML5, CSS3 und JavaScript 12/2014
- Rezension: Schrödinger programmiert Java 03/2014
- Rezension: Seven Web Frameworks in Seven Weeks 10/2014

**Buch (Fortsetzung)**

Rezension: Spieleprogrammierung mit Android Studio	10/2014
Rezension: Using SQLite	01/2014
Rezension: Vim in der Praxis	06/2014
Rezension: Weniger schlecht programmieren	01/2014
Rezension: Wien wartet auf Dich! – Produktive Projekte und Teams	03/2014
Rezension: Wissenschaftliche Arbeiten schreiben mit $\LaTeX$	06/2014

**Büro**

Professionelles Database Publishing	06/2014
Äquivalente Windows-Programme unter Linux – Systemprogramme und Zubehör	09/2014

**C****CSS**

Rezension: Apps mit HTML5 und CSS3 für iPad, iPhone und Android	02/2014
Rezension: Schrödinger lernt HTML5, CSS3 und JavaScript	12/2014

**Community**

Bericht: Herbsttagung DANTE e.V. 2014 in Karlsruhe	10/2014
Definition Freier Inhalte	11/2014
Die dritte Katastrophe – Teil 1	11/2014
Die dritte Katastrophe – Teil 2	12/2014
Fo Rensiker – sofort!	10/2014
Interview mit den Musikpiraten	12/2014
PyLadies Vienna – Interview mit Floor Drees	04/2014
Recht und Unrecht	09/2014
Rezension: Clean Coder	06/2014
Rezension: Kanban in der IT	04/2014
Rezension: Wien wartet auf Dich! – Produktive Projekte und Teams	03/2014
Siebter freiesMagazin-Programmierwettbewerb: Tron	10/2014

**Community (Fortsetzung)**

State of the Commons – Zustand der Creative-Commons-Lizenzen	12/2014
Ubucon 2014 in Katlenburg-Lindau	12/2014
freiesMagazin als HTML und EPUB erstellen	05/2014

**D****DOS**

FreeDOS – Totgesagte leben länger	11/2014
-----------------------------------	---------

**Dateiverwaltung**

Shellskript podfetch – Podcasts automatisch herunterladen	12/2014
Äquivalente Windows-Programme unter Linux – Systemprogramme und Zubehör	09/2014

**Datenbanken**

Neues von Redis	04/2014
Rezension: Using SQLite	01/2014

**Datensicherung**

Retten von verlorenen Daten mit PhotoRec	07/2014
--	---------

**Debian**

Administration von Debian & Co im Textmodus – Teil I	08/2014
Administration von Debian & Co im Textmodus – Teil II	09/2014
Administration von Debian & Co im Textmodus – Teil III	10/2014
Reprepro – Debian-Systeme mit einem selbst aufgesetzten Paket-Repo versorgen	10/2014
Rezension: Debian GNU/Linux – Das umfassende Handbuch	02/2014
Rezension: Linux-Server mit Debian 7 GNU/Linux	11/2014

**Desktop**

Kurzvorstellung: elementary OS	03/2014
--------------------------------	---------

**Distribution**

Fedora 20	02/2014
Kurzvorstellung: elementary OS	03/2014
Pinguine haben kurze Beine – Die Laufzeiten der Linuxdistributionen	03/2014
Red Hat Enterprise Linux 7	08/2014
Ubuntu und Kubuntu 14.04 LTS	06/2014
Ubuntu und Kubuntu 14.10	12/2014

**E****E-Mail**

E-Mails filtern mit KMail	05/2014
Spamfilterung mit bogofilter	07/2014
Äquivalente Windows-Programme unter Linux – Teil 5: Internet-Programme (1)	04/2014
Äquivalente Windows-Programme unter Linux – Teil 5: Internet-Programme (2)	06/2014

**EPUB**

freiesMagazin als HTML und EPUB erstellen	05/2014
---	---------

**Editor**

Kurztipp: Suchen und Finden mit ack statt grep	06/2014
Rezension: Eclipse IDE	05/2014
Rezension: Vim in der Praxis	06/2014
jEdit – der unterschätzte Texteditor	05/2014
Äquivalente Windows-Programme unter Linux – Systemprogramme und Zubehör	09/2014

**F****Fedora**

Fedora 20	02/2014
-----------	---------

**Firefox**

Äquivalente Windows-Programme unter Linux – Teil 5: Internet-Programme (1)	04/2014
Äquivalente Windows-Programme unter Linux – Teil 5: Internet-Programme (2)	06/2014

**Freie Projekte**

ADempiere – der lange Weg zur kurzen Installation eines Open-Source-ERP-Systems	11/2014
Bericht: Herbsttagung DANTE e.V. 2014 in Karlsruhe	10/2014
Definition Freier Inhalte	11/2014
FreeDOS – Totgesagte leben länger	11/2014
GPS: Tracks und Routen erstellen mit QLandkarte GT	06/2014
Interview mit den Musikpiraten	12/2014
Kassensystem in C.U.O.N.	08/2014
Multimedia- und Liedprojektion mit OpenLP	05/2014
Professionelles Database Publishing	06/2014
R-Webapplikationen mit Shiny und OpenCPU	09/2014
State of the Commons – Zustand der Creative-Commons-Lizenzen	12/2014

**G****Grafik**

Retten von verlorenen Daten mit PhotoRec	07/2014
Rezension: Inkscape – Der Weg zur professionellen Vektorgrafik	07/2014
Sozi – Eine kurze Einführung in das Inkscape-Plug-in	02/2014

**H****HTML**

Rezension: Apps mit HTML5 und CSS3 für iPad, iPhone und Android	02/2014
Rezension: Schrödinger lernt HTML5, CSS3 und JavaScript	12/2014
freiesMagazin als HTML und EPUB erstellen	05/2014

**Hardware**

Arduino	10/2014
Im Test: PocketBook Touch 622	04/2014
Linux auf dem Laptop (Acer Aspire V3-771G)	05/2014
Rezension: BeagleBone für Einsteiger	05/2014
Vorstellung: BeagleBone Black	05/2014
WLAN-AP mit dem Raspberry Pi	03/2014

**I****Installation**

Linux auf dem Laptop (Acer Aspire V3-771G)	05/2014
--	---------

**Internet**

Browser: alle doof. – Eine Odyssee durch die Welt der Web-betrachter	09/2014
Mit Tarnkappe im Netz – Das Tor Browser Bundle	05/2014
Rezension: Django Essentials	11/2014
Rezension: Seven Web Frameworks in Seven Weeks	10/2014
Statische Webseiten mit Pelican erstellen	07/2014
Torify: Programme im Terminal anonymisieren	06/2014
Äquivalente Windows-Programme unter Linux – Teil 5:	04/2014
Internet-Programme (1)	
Äquivalente Windows-Programme unter Linux – Teil 5:	06/2014
Internet-Programme (2)	

**Interview**

Interview mit den Musikpiraten	12/2014
PyLadies Vienna – Interview mit Floor Drees	04/2014
Spielend programmieren: Interview mit Horst Jens	03/2014

**J****Java**

Rezension: Design Patterns mit Java	09/2014
Rezension: Eclipse IDE – kurz & gut	01/2014

**Java (Fortsetzung)**

Rezension: Java – Eine Einführung in die Programmierung	12/2014
Rezension: Schrödinger programmiert Java	03/2014

**JavaScript**

Arduino	10/2014
Rezension: Geheimnisse eines JavaScript-Ninjas	07/2014
Rezension: JavaScript – Grundlagen, Programmierung, Praxis	05/2014
Rezension: Kids programmieren 3D-Spiele mit JavaScript	08/2014
Rezension: Schrödinger lernt HTML5, CSS3 und JavaScript	12/2014
Vorstellung: BeagleBone Black	05/2014

**K****KDE**

Ubuntu und Kubuntu 14.10	12/2014
--------------------------	---------

**Kernel**

Der April im Kernelrückblick	05/2014
Der August im Kernelrückblick	09/2014
Der Dezember im Kernelrückblick	01/2014
Der Februar im Kernelrückblick	03/2014
Der Januar im Kernelrückblick	02/2014
Der Juli im Kernelrückblick	08/2014
Der Juni im Kernelrückblick	07/2014
Der Mai im Kernelrückblick	06/2014
Der März im Kernelrückblick	04/2014
Der November im Kernelrückblick	12/2014
Der Oktober im Kernelrückblick	11/2014
Der September im Kernelrückblick	10/2014

**Kommerzielle Software**

Broken Age	12/2014
Gone Home	02/2014

**Kommerzielle Software (Fortsetzung)**

Planet Explorers	07/2014
Review: Papers, Please	06/2014
Typographie lernen mit Type:Rider	07/2014
Äquivalente Windows-Programme unter Linux – Teil 3	01/2014
Äquivalente Windows-Programme unter Linux – Teil 4: Bildbearbeitung (2)	02/2014

**Konsole**

Kurztipp: Suchen und Finden mit ack statt grep	06/2014
Rezension: Linux-Kommandoreferenz	04/2014
Shell-Skripte – Kleine Helfer selbst gemacht	02/2014
tmux – Das Kung-Fu der Terminal-Ninjas	05/2014

**Kubuntu**

Ubuntu und Kubuntu 14.10	12/2014
--------------------------	---------

**Kurzgeschichte**

Die dritte Katastrophe – Teil 1	11/2014
Die dritte Katastrophe – Teil 2	12/2014
Fo Rensiker – sofort!	10/2014
Recht und Unrecht	09/2014

**L****LaTeX**

Bericht: Herbsttagung DANTE e.V. 2014 in Karlsruhe	10/2014
Explizite Positionierung in $\LaTeX$	02/2014
Professionelles Database Publishing	06/2014
Registerhaltiger Satz mit $\LaTeX$	07/2014
Rezension: Wissenschaftliche Arbeiten schreiben mit $\LaTeX$	06/2014
freiesMagazin als HTML und EPUB erstellen	05/2014

**Linux allgemein**

Administration von Debian & Co im Textmodus – Teil I	08/2014
--	---------

**Linux allgemein (Fortsetzung)**

Administration von Debian & Co im Textmodus – Teil II	09/2014
Administration von Debian & Co im Textmodus – Teil III	10/2014
Creative Commons 4.0 vorgestellt	02/2014
Pinguine haben kurze Beine – Die Laufzeiten der Linuxdistributionen	03/2014
Reprepro – Debian-Systeme mit einem selbst aufgesetzten Paket-Repo versorgen	10/2014
Rezension: Linux-Kommandoreferenz	04/2014
jEdit – der unterschätzte Texteditor	05/2014
Äquivalente Windows-Programme unter Linux – Teil 3	01/2014
Äquivalente Windows-Programme unter Linux – Teil 4: Bildbearbeitung (2)	02/2014
Äquivalente Windows-Programme unter Linux – Teil 5: Internet-Programme (1)	04/2014
Äquivalente Windows-Programme unter Linux – Teil 5: Internet-Programme (2)	06/2014

**Lizenz**

Creative Commons 4.0 vorgestellt	02/2014
Definition Freier Inhalte	11/2014
State of the Commons – Zustand der Creative-Commons-Lizenzen	12/2014

**M****Magazin**

Creative Commons 4.0 vorgestellt	02/2014
Registerhaltiger Satz mit $\LaTeX$	07/2014
Siebter freiesMagazin-Programmierwettbewerb: Tron	10/2014
freiesMagazin als HTML und EPUB erstellen	05/2014

**Multimedia**

Ein Einstieg in LIRC mit inputlirc	04/2014
Multimedia- und Liedprojektion mit OpenLP	05/2014

**Multimedia (Fortsetzung)**

- Retten von verlorenen Daten mit PhotoRec 07/2014
- Shellskript podfetch – Podcasts automatisch herunterladen 12/2014

**N****Navigation**

- GPS: Tracks und Routen erstellen mit QLandkarte GT 06/2014

**Netzwerk**

- Hidden in Plain Sight: Netzlaufwerke ausspähsicher nutzen 04/2014
- Kurztipp: Heimcontainer oder Datentresor ohne TrueCrypt 04/2014
- Spacewalk – Teil 1: Einführung, Übersicht und Installation 08/2014
- Spacewalk – Teil 2: Registrierung und Verwaltung von Systemen 09/2014
- Spacewalk – Teil 3: Automatisierung und Kickstart 11/2014
- WLAN-AP mit dem Raspberry Pi 03/2014

**O****OpenStreetMap**

- GPS: Tracks und Routen erstellen mit QLandkarte GT 06/2014

**P****PDF**

- Im Test: PocketBook Touch 622 04/2014
- Professionelles Database Publishing 06/2014

**Programmierung**

- Ein Blick auf Octave 3.8 04/2014
- Eine Einführung in Octave 12/2014
- Explizite Positionierung in  $\LaTeX$  02/2014
- Git-Tutorium – Teil 1 12/2014
- Kurztipp: Suchen und Finden mit ack statt grep 06/2014
- PyLadies Vienna – Interview mit Floor Drees 04/2014
- Registerhaltiger Satz mit  $\LaTeX$  07/2014

**Programmierung (Fortsetzung)**

- Rezension: 97 Things Every . . . Should Know 08/2014
- Rezension: Android – Einstieg in die Programmierung 09/2014
- Rezension: Android – kurz & gut 02/2014
- Rezension: Apps mit HTML5 und CSS3 für iPad, iPhone und Android 02/2014
- Rezension: BeagleBone für Einsteiger 05/2014
- Rezension: C++: Das komplette Starterkit für den einfachen Einstieg 08/2014
- Rezension: Clean Coder 06/2014
- Rezension: Datendesign mit R 09/2014
- Rezension: Design Patterns mit Java 09/2014
- Rezension: Django Essentials 11/2014
- Rezension: Eclipse IDE 05/2014
- Rezension: Eclipse IDE – kurz & gut 01/2014
- Rezension: Geheimnisse eines JavaScript-Ninjas 07/2014
- Rezension: Git – kurz & gut 11/2014
- Rezension: Inkscape – Der Weg zur professionellen Vektorgrafik 07/2014
- Rezension: Java – Eine Einführung in die Programmierung 12/2014
- Rezension: JavaScript – Grundlagen, Programmierung, Praxis 05/2014
- Rezension: Kanban in der IT 04/2014
- Rezension: Kids programmieren 3D-Spiele mit JavaScript 08/2014
- Rezension: Raspberry Pi programmieren mit Python 04/2014
- Rezension: Schrödinger lernt HTML5, CSS3 und JavaScript 12/2014
- Rezension: Schrödinger programmiert Java 03/2014
- Rezension: Seven Web Frameworks in Seven Weeks 10/2014
- Rezension: Spieleprogrammierung mit Android Studio 10/2014
- Rezension: Weniger schlecht programmieren 01/2014
- Scratch-2-Tutorial 03/2014
- Shell-Skripte – Kleine Helfer selbst gemacht 02/2014
- Shellskript podfetch – Podcasts automatisch herunterladen 12/2014
- Siebter **freiesMagazin**-Programmierwettbewerb: Tron 10/2014

**Programmierung (Fortsetzung)**

Spielend programmieren: Interview mit Horst Jens	03/2014
Statische Webseiten mit Pelican erstellen	07/2014
jEdit – der unterschätzte Texteditor	05/2014

**Präsentation**

Multimedia- und Liedprojektion mit OpenLP	05/2014
Präsentationen mit HTML und reveal.js	03/2014
Sozi – Eine kurze Einführung in das Inkscape-Plug-in	02/2014

**Python**

Arduino	10/2014
PyLadies Vienna – Interview mit Floor Drees	04/2014
Rezension: Django Essentials	11/2014
Rezension: Raspberry Pi programmieren mit Python	04/2014
Statische Webseiten mit Pelican erstellen	07/2014
Vorstellung: BeagleBone Black	05/2014

**R****Raspberry Pi**

Rezension: Raspberry Pi programmieren mit Python	04/2014
WLAN-AP mit dem Raspberry Pi	03/2014

**Rezension**

Gone Home	02/2014
Im Test: PocketBook Touch 622	04/2014
Planet Explorers	07/2014
Review: Papers, Please	06/2014
Rezension: 97 Things Every ... Should Know	08/2014
Rezension: Android – Einstieg in die Programmierung	09/2014
Rezension: Android – kurz & gut	02/2014
Rezension: Apps mit HTML5 und CSS3 für iPad, iPhone und Android	02/2014

**Rezension (Fortsetzung)**

Rezension: BeagleBone für Einsteiger	05/2014
Rezension: C++: Das komplette Starterkit für den einfachen Einstieg	08/2014
Rezension: Datendesign mit R	09/2014
Rezension: Debian GNU/Linux – Das umfassende Handbuch	02/2014
Rezension: Design Patterns mit Java	09/2014
Rezension: Django Essentials	11/2014
Rezension: Eclipse IDE	05/2014
Rezension: Eclipse IDE – kurz & gut	01/2014
Rezension: Geheimnisse eines JavaScript-Ninjas	07/2014
Rezension: Git – kurz & gut	11/2014
Rezension: Inkscape – Der Weg zur professionellen Vektorgrafik	07/2014
Rezension: Java – Eine Einführung in die Programmierung	12/2014
Rezension: JavaScript – Grundlagen, Programmierung, Praxis	05/2014
Rezension: Kanban in der IT	04/2014
Rezension: Kids programmieren 3D-Spiele mit JavaScript	08/2014
Rezension: Linux-Kommandoreferenz	04/2014
Rezension: Linux-Server mit Debian 7 GNU/Linux	11/2014
Rezension: Raspberry Pi programmieren mit Python	04/2014
Rezension: Schrödinger lernt HTML5, CSS3 und JavaScript	12/2014
Rezension: Schrödinger programmiert Java	03/2014
Rezension: Seven Web Frameworks in Seven Weeks	10/2014
Rezension: Spieleprogrammierung mit Android Studio	10/2014
Rezension: Using SQLite	01/2014
Rezension: Vim in der Praxis	06/2014
Rezension: Weniger schlecht programmieren	01/2014
Rezension: Wien wartet auf Dich! – Produktive Projekte und Teams	03/2014
Rezension: Wissenschaftliche Arbeiten schreiben mit L <sup>A</sup> T <sub>E</sub> X	06/2014

**S****Server**

Rezension: Linux-Server mit Debian 7 GNU/Linux 11/2014

**Sicherheit**

Hidden in Plain Sight: Netzlaufwerke ausspähsicher nutzen 04/2014

Kurztipp: Heimcontainer oder Datentresor ohne TrueCrypt 04/2014

Mit Tarnkappe im Netz – Das Tor Browser Bundle 05/2014

Pinguine haben kurze Beine – Die Laufzeiten der Linuxdistributionen 03/2014

Torify: Programme im Terminal anonymisieren 06/2014

**Softwareinstallation**

ADempiere – der lange Weg zur kurzen Installation eines Open-Source-ERP-Systems 11/2014

**Spiele**

Broken Age 12/2014

Gone Home 02/2014

Minetest – Block für Block zur kantigen Traumwelt 11/2014

Planet Explorers 07/2014

Review: Papers, Please 06/2014

Rezension: Spieleprogrammierung mit Android Studio 10/2014

Roll'm Up - Ein altes Flipperspiel neu entdeckt 03/2014

Scratch-2-Tutorial 03/2014

Spielend programmieren: Interview mit Horst Jens 03/2014

Typographie lernen mit Type:Rider 07/2014

**Systemverwaltung**

Administration von Debian & Co im Textmodus – Teil I 08/2014

Administration von Debian & Co im Textmodus – Teil II 09/2014

Administration von Debian & Co im Textmodus – Teil III 10/2014

Linux auf dem Laptop (Acer Aspire V3-771G) 05/2014

**Systemverwaltung (Fortsetzung)**

Reprepro – Debian-Systeme mit einem selbst aufgesetzten 10/2014

Paket-Repo versorgen

Spacewalk – Teil 1: Einführung, Übersicht und Installation 08/2014

Spacewalk – Teil 2: Registrierung und Verwaltung von Systemen 09/2014

Spacewalk – Teil 3: Automatisierung und Kickstart 11/2014

WLAN-AP mit dem Raspberry Pi 03/2014

Äquivalente Windows-Programme unter Linux – Systemprogramme und Zubehör 09/2014

**Systemüberwachung**

Spacewalk – Teil 1: Einführung, Übersicht und Installation 08/2014

Spacewalk – Teil 2: Registrierung und Verwaltung von Systemen 09/2014

Spacewalk – Teil 3: Automatisierung und Kickstart 11/2014

**T****Tipps & Tricks**

ADempiere – der lange Weg zur kurzen Installation eines Open-Source-ERP-Systems 11/2014

Shell-Skripte – Kleine Helfer selbst gemacht 02/2014

Statische Webseiten mit Pelican erstellen 07/2014

jEdit – der unterschätzte Texteditor 05/2014

**U****Ubuntu**

Ubucon 2014 in Katlenburg-Lindau 12/2014

Ubuntu und Kubuntu 14.04 LTS 06/2014

Ubuntu und Kubuntu 14.10 12/2014

**V****Veranstaltung**

Bericht: Herbsttagung DANTE e.V. 2014 in Karlsruhe 10/2014

**Veranstaltung (Fortsetzung)**

Ubucon 2014 in Katlenburg-Lindau 12/2014

**Verschlüsselung**

Hidden in Plain Sight: Netzlaufwerke ausspähsicher nutzen 04/2014

Kurztipp: Heimcontainer oder Datentresor ohne TrueCrypt 04/2014

**Versionsverwaltung**

Git-Tutorium – Teil 1 12/2014

Rezension: Git – kurz & gut 11/2014

**W****Wettbewerb**

Siebter freiesMagazin-Programmierwettbewerb: Tron 10/2014

**Windows**

FreeDOS – Totgesagte leben länger 11/2014

Äquivalente Windows-Programme unter Linux – Systemprogramme und Zubehör 09/2014

**Windows (Fortsetzung)**

Äquivalente Windows-Programme unter Linux – Teil 3 01/2014

Äquivalente Windows-Programme unter Linux – Teil 4: Bildbearbeitung (2) 02/2014

Äquivalente Windows-Programme unter Linux – Teil 5: Internet-Programme (1) 04/2014

Äquivalente Windows-Programme unter Linux – Teil 5: Internet-Programme (2) 06/2014

**Wissen und Bildung**

Ein Blick auf Octave 3.8 04/2014

Eine Einführung in Octave 12/2014

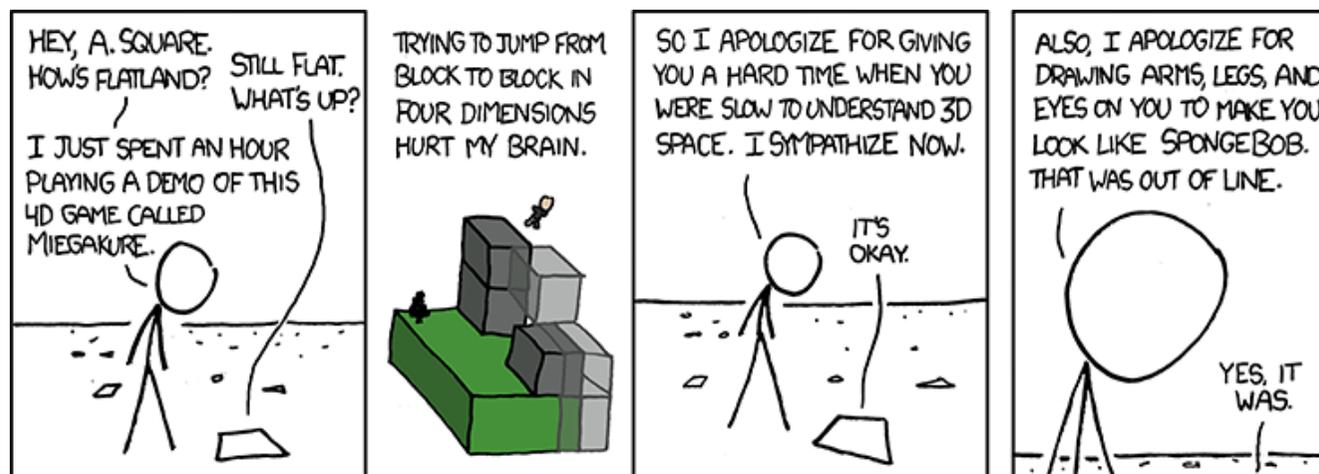
Mathematik mit Open Source 07/2014

Präsentationen mit HTML und reveal.js 03/2014

R-Webapplikationen mit Shiny und OpenCPU 09/2014

Scratch-2-Tutorial 03/2014

State of the Commons – Zustand der Creative-Commons-Lizenzen 12/2014



“Flatland” © by Randall Munroe (CC-BY-NC-2.5), <http://xkcd.com/721/>

## Impressum

freiesMagazin erscheint als PDF, EPUB und HTML einmal monatlich.

### Kontakt

E-Mail [redaktion@freiesMagazin.de](mailto:redaktion@freiesMagazin.de)  
 Postanschrift **freiesMagazin**  
 c/o Dominik Wagenführ  
 Beethovenstr. 9/1  
 71277 Rutesheim  
 Webpräsenz <http://www.freiesmagazin.de/>

### Autoren dieser Ausgabe

Die GIMPer **S. 5**  
 Mathias Menzer **S. 3**  
 Jochen Schnelle **S. 44**  
 Maria Seliger **S. 31**  
 Matthias Sitte **S. 25**  
 Christian Stankowic **S. 17**  
 Sujeevan Vijayakumaran **S. 11, S. 23**  
 Dominik Wagenführ **S. 35**  
 Stefan Wichmann **S. 46**

ISSN 1867-7991

Erscheinungsdatum: 4. Januar 2015

### Redaktion

Christian Schnell Matthias Sitte  
 Dominik Wagenführ (Verantwortlicher Redakteur)

### Satz und Layout

Jonas Dabelow Moritz Kiefer  
 Christoph Lehmann Kai Welke

### Korrektur

Daniel Braun Frank Brungräber  
 Vicki Ebeling Stefan Fangmeier  
 Mathias Menzer Christian Schnell  
 Karsten Schuldt

### Veranstaltungen

Ronny Fischer

### Logo-Design

Arne Weinberg ([CC-BY-SA 4.0 Unported](#))

Dieses Magazin wurde mit  $\text{\LaTeX}$  erstellt. Mit vollem Namen gekennzeichnete Beiträge geben nicht notwendigerweise die Meinung der Redaktion wieder. Wenn Sie freiesMagazin ausdrucken möchten, dann denken Sie bitte an die Umwelt und drucken Sie nur im Notfall. Die Bäume werden es Ihnen danken. ;-)

Soweit nicht anders angegeben, stehen alle Artikel, Beiträge und Bilder in freiesMagazin unter der [Creative-Commons-Lizenz CC-BY-SA 4.0 International](#). Das Copyright liegt beim jeweiligen Autor. Die Kommentar- und Empfehlens-Icons wurden von Maren Hachmann erstellt und unterliegen ebenfalls der [Creative-Commons-Lizenz CC-BY-SA 4.0 International](#). freiesMagazin unterliegt als Gesamtwerk der [Creative-Commons-Lizenz CC-BY-SA 4.0 Unported](#) mit Ausnahme der Inhalte, die unter einer anderen Lizenz hierin veröffentlicht werden. Das Copyright liegt bei Dominik Wagenführ. Es wird erlaubt, das Werk/die Werke unter den Bestimmungen der Creative-Commons-Lizenz zu kopieren, zu verteilen und/oder zu modifizieren. Die xkcd-Comics stehen separat unter der [Creative-Commons-Lizenz CC-BY-NC 2.5 Generic](#). Das Copyright liegt bei [Randall Munroe](#).