

Topthemen dieser Ausgabe

NoSQL – Jenseits der relationalen Datenbank

In Hinblick auf Unternehmensanwendungen wurden relationalen Datenbanken und daraus entstehend Ende der 70er Jahre SQL entwickelt. Zu Beginn des 21. Jahrhunderts wurden freie Implementierungen von relationalen, per SQL abfragbaren Datenbanken wie MySQL, PostgreSQL und SQLite wirklich einsatzfähig und werden heute häufig genutzt. Dennoch ist die Entwicklung nicht stehen geblieben und es gibt inzwischen Alternativen zu den herkömmlichen relationalen Datenbanken. Der Artikel soll einen Blick auf solche neue Datenbankansätze werfen. ([weiterlesen](#))

Seite 4



MeeGo im Blick

Ende Mai dieses Jahres erschien Version 1.0 der Distribution MeeGo. MeeGo orientiert sich vornehmlich am Einsatz auf mobilen Geräten, wie Handhelds und Netbooks. Dabei setzt MeeGo auf eine Benutzeroberfläche, die in puncto Benutzbarkeit und Benutzerführung neue Akzente setzen will, gerade auf den genannten Geräteklassen. Hört sich interessant an? Ist es auch. ([weiterlesen](#))

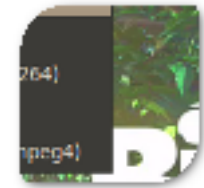
Seite 11



Videoschnitt mit Avidemux

Im digitalen Video-Zeitalter ist das nachträgliche Schneiden, Zusammenfügen oder Umwandeln von Videos auch im Heimbereich an der Tagesordnung. Sei es, um den gerade aufgenommenen Urlaubsfilm zu schneiden, die Werbung aus einer Reportage zu entfernen oder ganz schlicht, um das Lieblingsmusikvideo auf dem iPod abspielen zu können. Hier kommt Avidemux ins Spiel, ein leistungsfähiges, unter der GPL (GNU General Public License) stehendes Videoschnittprogramm, das kostenfrei erhältlich ist. ([weiterlesen](#))

Seite 26



Editorial

Über Pingbacks, Double-Opt-Ins und Flattr

Veränderungen der Webseite

Dieses Editorial fangen wir etwas technischer und selbstorientierter an, denn wir möchten auf zwei kleine Änderungen auf der **freiesMagazin**-Webseite [1] hinweisen.

Zum einen fanden wir es immer sehr schade, dass wir keinerlei Notiz davon nehmen konnten, wenn Leser über das Erscheinen von **freiesMagazin** berichten – außer wir durchstöberten das Netz nach Berichten zum Magazin. Aus diesem Grund haben wir für unser benutztes Content-Management-System Drupal ein Pingback-Modul [2] aktiviert, welches es nun erlaubt, die Erwähnung auf anderen Webseiten automatisch als Kommentar bei uns zu veröffentlichen.

Die zweite Änderung konnten die meisten RSS-Feed-Abonnenten bereits selbst sehen. Von Zeit zu Zeit wollen wir Umfragen in Bezug zur aktuellen **freiesMagazin**-Ausgabe abhalten, die wir dann im Folgemonat auswerten. Welche Themen und Fragen uns beschäftigen werden, wird die Zukunft zeigen.

Ein Thema, bei dem Sie uns vielleicht helfen können, liebe Leser, ist eine weitere Funktion, die wir Ihnen gerne bieten würden. Wir suchen noch ein Modul für Drupal, sodass die Kommentare einzelner **freiesMagazin**-Nachrichten abonniert

werden können. Alle bisher getesteten Lösungen haben den Nachteil, dass sie kein Double-Opt-In [3] anbieten. Darunter versteht man, dass man etwas nicht einfach abonniert, sondern das Abonnement noch einmal per E-Mail und Aktivierungslink bestätigen muss. Dies soll eine Ausnutzung durch Spammer verhindern. Sollte also jemand so ein Modul für Drupal kennen, wäre es schön, wenn sich derjenige unter redaktion@freiesMagazin.de bei uns melden würde.

Ihre Meinung zu Flattr

Als erste Umfrage [4] haben wir uns entschieden, das Thema „Flattr“ [5] aufzugreifen, zu dem wir bereits in der Juliausgabe [6] Stellung genommen hatten. Unsere Meinung hatten wir damit mitgeteilt, aber wir waren auch an den Meinungen unserer Leserschaft interessiert.

Die Reaktionen sind dabei doch recht verhalten, wenn man bedenkt, wie oft Flattr in diversen Blogpostings der letzten Monate erwähnt wurde. Von den 298 Teilnehmern der Umfrage nutzen nur 34 (11,4%) Flattr aktiv, d. h. sie sind bei dem Dienst angemeldet, um selbst Geld an andere Flattr-Teilnehmer spenden zu können oder im Falle von vier Umfrageteilnehmern wollen sie selbst Geld für ihre produzierten Inhalte einnehmen. Dies sind also recht wenig Flattr-Nutzer.

Inhalt

Linux allgemein

NoSQL – Jenseits der relationalen Datenbanken	S. 4
MeeGo im Blick	S. 11
Das neue openSUSE 11.3 ist da	S. 18
Juni und Juli im Kernelrückblick	S. 21

Anleitungen

Tiling Windows mit Compiz und dem Grid-Plugin	S. 22
L ^A T _E X-Symbole einfügen mit LSS	S. 24

Software

Videoschnitt mit Avidemux	S. 26
---------------------------	-------

Community

Rezension: Die Anarchie der Hacker	S. 31
Rezension: CouchDB – kurz & gut	S. 34
Rezension: Beginning CouchDB	S. 36

Magazin

Editorial	S. 2
Leserbriefe	S. 38
Veranstaltungen	S. 39
Vorschau	S. 39
Konventionen	S. 39
Impressum	S. 40

Für 120 Leser (40,3%) ist Flattr dabei ein sinnvoller Dienst, auch wenn, wie oben gesehen, ein Großteil Flattr nicht selbst nutzt. In etwa gleich vielen Lesern (131 bzw. 44,0%) ist Flattr aber egal. Nur 15,8% (47 Stimmen) der Umfrageteilnehmer können sich mit Flattr nicht anfreunden.

Dabei teilen sich aber auch die Meinungen, was die Verwendung von Flattr im **freiesMagazin**-Projekt angeht. Die einen sehen die Frage, ob freie Projekte Geld einnehmen dürfen bereits als „*religiös anmutend*“ an [7] und denken, dass wir aus dem Umgang mit Flattr für die Zukunft etwas für freie Inhalte im Allgemeinen lernen können. Für andere wiederum ist Flattr kommerziell und sie denken, dass „*ein freies Magazin frei von Kommerz*“ sein sollte [8].

Die **freiesMagazin**-Redaktion steht dabei auf dem Standpunkt, dass es auch freien Projekten erlaubt sein sollte, von ihren Inhalten zu leben. Es gibt zum Beispiel zahlreiche Künstler auf Jamendo [9], die Ihre Musik unter freien Lizenzen anbieten. Damit diese unterstützt werden, schadet es nicht, wenn ab und an jemand für die angebotene Musik zahlt. Wir selbst wollen aber, wie bereits in der letzten Ausgabe dargelegt, keinerlei finanziellen Vorteil aus **freiesMagazin** ziehen.

Zum Thema hat uns unser Leser Erik [10] auch noch auf einen Bericht bei gulli.com [11] hingewiesen. Dort findet man einige Auszüge der Einnahmen von Flattr-Teilnehmern – und die Feststellung, dass sich niemand allein von den Flattr-Einnahmen hauptberuflich ernähren könnte.

Chefredakteur wieder aufgetaucht

Wie einige aufmerksame **freiesMagazin**-Leser in den Kommentaren bei Pro-Linux [12] bemerkt haben, hat Dominik Wagenführ bereits für die Juli-Ausgabe von **freiesMagazin** wieder Artikel geschrieben. Das ist richtig. Und das, obwohl er zum besagten Zeitraum „offiziell“ noch als vermisst galt. Jetzt können wir ja Klartext reden: Was einige Witzbolde (also wir) „vermisst“ nannten, war tatsächlich nur ein Urlaub. Kein Grund zur Sorge also. Jetzt ist Dominik Wagenführ aber wieder da, schreibt wieder fleißig Artikel und hält auch sonst die Fäden im **freiesMagazin**-Team zusammen.

Interview bei DeimHart


Wer sich für **freiesMagazin** interessiert und vielleicht etwas mehr über die Hintergründe und Arbeitsabläufe während der Magazinerstellung erfahren möchte, der kann dazu mehr in einem Podcast bei DeimHart [13] hören. Dort stand Dominik Wagenführ seinen beiden Gesprächspartnern Dirk Deimeke und Roman Hanhart, den Machern von DeimHart, Rede und Antwort. In dem gut 102 Minuten langen Interview geht es aber noch um mehr als um **freiesMagazin**, auch das Thema „ \LaTeX “ wird besprochen. Das wundert nicht, schließlich ist \LaTeX das Hauptwerkzeug, mit dem **freiesMagazin** erstellt wird. Den Podcast kann man sich auf der DeimHart-Webseite [14] als OGG- oder MP3-Datei herunterladen. Auch ein weiteres Stöbern bei DeimHart lohnt sich mit

Sicherheit. Die beiden Podcaster berichten regelmäßig über Freie Software, Linux sowie über gesellschaftliche Themen.

Nun wünschen wir Ihnen wieder viel Spaß mit der neuen Ausgabe.

Ihre **freiesMagazin**-Redaktion

LINKS

- [1] <http://www.freiesmagazin.de/>
- [2] <http://de.wikipedia.org/wiki/Pingback>
- [3] http://de.wikipedia.org/wiki/Double_Opt-in
- [4] <http://www.freiesmagazin.de/20100704-umfrage-was-halten-sie-von-flattr>
- [5] <http://flattr.com/> 
- [6] <http://www.freiesmagazin.de/freiesMagazin-2010-07>
- [7] <http://www.freiesmagazin.de/20100704-umfrage-was-halten-sie-von-flattr#comment-1009>
- [8] <http://www.freiesmagazin.de/20100704-umfrage-was-halten-sie-von-flattr#comment-998>
- [9] <http://www.jamendo.com/de>
- [10] <http://www.freiesmagazin.de/20100704-umfrage-was-halten-sie-von-flattr#comment-1001>
- [11] <http://www.gulli.com/news/kommentar-flattr-und-co-das-finanzierungsm-rchen-2010-07-05>
- [12] <http://www.deimhart.net/>
- [13] <http://deimhart.net/index.php?/archives/82-Dominik-und-freiesMagazin-und-LaTeX.html>

Das Editorial kommentieren 



NoSQL – Jenseits der relationalen Datenbanken von Jochen Schnelle

Relationale Datenbanken, welche in der Regel (aber nicht zwingend) per SQL abgefragt werden, findet man heute im Hintergrund vieler Anwendungen – sei es bei Wikis (z. B. MediaWiki), Content Management Systemen (z. B. Typo3, Joomla), Blog-Systemen (z. B. Wordpress) oder Ticket-/Bugtracking-Systemen (z. B. Roundup). Ganz zu schweigen natürlich von vielen kommerziellen Systemen wie ERP-Software, Lagerverwaltungssoftware etc. Mit Hinblick auf Unternehmensanwendungen wurde eine der ersten relationalen Datenbanken, IBM System R [1], entwickelt. Zur Abfrage der Daten wurde auch eine spezielle Programmiersprache entwickelt, aus der Ende der 70er Jahre des vorigen Jahrhunderts dann SQL hervorging [2]. Zu Beginn des 21. Jahrhunderts wurden freie Implementierungen von relationalen, per SQL abfragbaren Datenbanken wie MySQL, PostgreSQL und SQLite wirklich einsatzfähig. Seitdem findet man diese in vielen freien Anwendungen, wie weiter oben bereits erwähnt.

Nun ist die Entwicklung der Datenbanken aber nicht stehen geblieben. Besonders Internetschwergewichte wie Google, eBay und in jüngerer Vergangenheit auch Twitter und Facebook stießen an die Grenzen und Möglichkeiten von konventionellen, relationalen Datenbanken und entwickelten bzw. adaptierten andere Lösungen, wie z. B. Google mit Bigtable [3] oder

Twitter mit Gizzard [4]. Dass dieses Gebiet auch für andere Firmen wie z. B. Yahoo und Adobe von Interesse ist, sieht man z. B. an der Entwicklung von Hadoop [5], einer Open-Source-Implementierung von Datenbank und Dateisystem für sehr große Datenmengen.

Zugegebenermaßen ist es wohl so, dass die allermeisten Leser von freiesMagazin bzw. Computernutzer im Allgemeinen auch nicht nur annähernd Datenbestände im hohen Terabyte- oder Petabyte-Bereich haben, geschweige denn selber (Web-)Applikation programmieren, welche riesige Datenmengen speichern und verarbeiten und gleichzeitig sehr viele An-/Abfragen pro Sekunde bedienen müssen. Trotzdem muss nicht immer eine relationale Datenbank zum Einsatz kommen. Je nach Anwendungsfall und Anforderung an die Struktur (und Kriterien für Abfragen) der Daten gibt es interessante Alternativen, welche im Folgenden vorgestellt werden. Dies sind zum einen dokumentenorientierte Datenbanken und zum anderen die sogenannten „Key-Value Stores“, was übersetzt so viel wie „Schlüssel-Wert-Speicher“ heißt.

Rahmenbedingungen

Wie in der Einleitung zu diesem Artikel bereits erwähnt, ist das Einsatzgebiet von relationalen Datenbanken sehr vielfältig und es würde den Rahmen dieses Artikels bei weitem sprengen, auf mögliche Implementierungen für dieses vielfältige Spektrum einzugehen. Daher wer-

den im Folgenden primär Webanwendungen wie Microblogging-Dienste (Twitter, identi.ca) und Blogsysteme bzw. CMS herangezogen.

Es wird auch keine komplette Implementierung in einer bestimmten Programmiersprache vorgestellt. Vielmehr werden Vorschläge für mögliche (Datenbank-)Schemata gemacht. Diese sind auch mehr als Startpunkt oder „Denkanstoß“ zu sehen, da individuelle Programme und Anwendungen gegebenenfalls speziellere oder andere Anforderungen an die Struktur stellen. Auf Themen wie Replikation, horizontale Skalierbarkeit der Datenbank [6] und Fehlertoleranz wird nur am Rande eingegangen, da diese Themen in der Regel erst bei sehr großen Anwendungen mit vielen zehntausend Nutzern und vielen quasi-gleichzeitigen Zugriffen relevant werden.

Klar ist auch: Wer dies unbedingt will, kann wahrscheinlich für jede Art von Anwendung jede Art der hier vorgestellten Datenbanken einsetzen. Ob dies aber immer unbedingt effizient ist oder der Aufwand für die Implementierung unverhältnismäßig hoch ist, ist dann ein anderes Thema.

Ein Gemeinsamkeit ist übrigens, dass für alle vorgestellten Datenbanken – egal ob SQL-basiert oder „SQL-frei“ – Schnittstellen für viele Programmiersprachen existieren, sodass es hier kaum Einschränkungen gibt. Zum Beispiel stehen mit **libsqlite** und **libcouchdb-glib** Bibliotheken für SQLite und CouchDB (siehe weiter unten) unter Linux zur Verfügung.

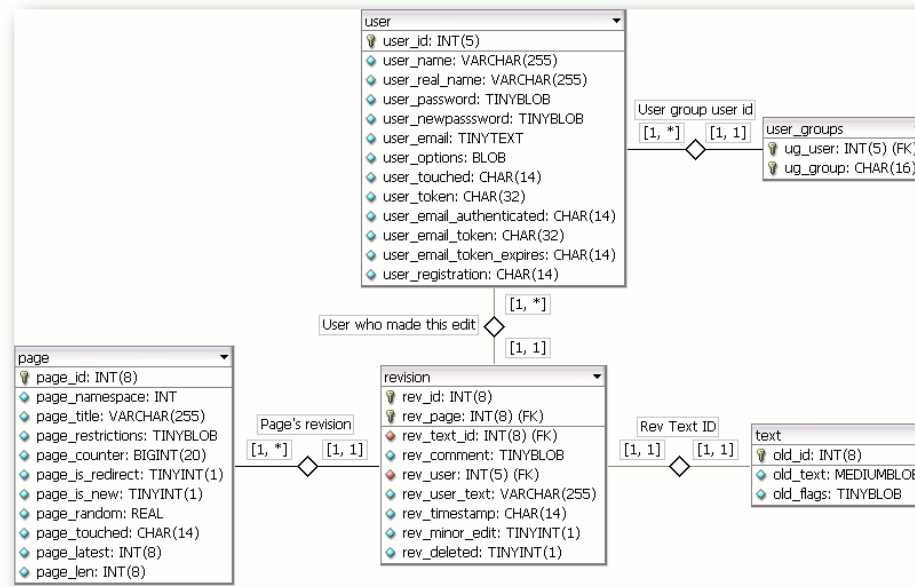
Relationale Datenbanken

Arbeitet man mit einer relationalen Datenbank, so steht vor dem eigentlichen Einrichten der Datenbank und dem Anlegen von Tabellen der Entwurf der Datenbank. Während erfahrene Nutzer dies bei kleinen Datenbanken auch im Kopf erledigen können, empfiehlt es sich bei etwas umfangreicheren Projekten, ein Datenbankschema in Form eines „Entity-Relationship-Modells“ [7] zu erstellen. Hier werden unter anderem die Beziehungen der Tabellen untereinander und Fremdschlüssel festgelegt (sogenannte „Foreign Key Constraints“ [8]). Anhand des Modells kann auch überprüft werden, ob die Datenbank komplett normalisiert ist [9], was gerade bei großen Datenbanken wichtig für das spätere fehlerfreie Arbeiten der Datenbank ist. Ist dies alles erledigt, können Datenbank, Tabellen und Tabellenfelder angelegt werden.

Tabellenfelder müssen in relationalen Datenbanken fest definiert sein. Das heißt, es wird ein Feldtyp (Integer, Float, Boolescher Wahrheitswert, Text usw.) sowie die Feldlänge festgelegt. Auch wenn Feldtyp und Feldlänge grundsätzlich im Nachhinein noch änderbar sind, bedeutet dies in der Regel einen (unter Umständen) erheblichen Aufwand und ein erhebliches Fehlerrisiko, gerade bei großen Datenbeständen. Deshalb muss man sich im Rahmen des Datenbankentwurfs auch zu Feldtypen und -längen eingehend Gedanken machen.

Die vordefinierten Felder haben z.B. bei Geschäftsanwendungen durchaus auch Sinn. Ist

eine Kundennummer z. B. immer fünfstellig, so kann man das Feld „kdnr“ als Typ Integer mit einer Länge von fünf definieren. Felder für Konten werden als Typ „Decimal“ mit zwei Nachkommastellen definiert. Der Kundenname darf maximal 30 Zeichen lang sein (weil er sonst z. B. nicht mehr in das Adressfeld des Briefpapiers passt), also ist der Feldtyp „Text“ mit einer Feldlänge von 30 Zeichen. Hier profitiert die Anwendung also vom Datenbankschema.



Beispiel für ein Entity-Relation-Modell. © AutumnSnow (GFDL)

Ein Vorteil von relationalen Datenbanken ist, dass SQL auch einfache Rechenoperationen für Zahlenfelder unterstützt. So kann man z. B. die Summe oder den Mittelwert einer definierten Auswahl von Feldern direkt innerhalb der Daten-

bankabfrage berechnen lassen. Dies hat den Vorteil, dass man das gewünschte Ergebnis direkt erhält und die Berechnung nicht noch selbst implementieren muss.

SQL-basierte Datenbanken bieten in der Regel auch eine feingranulare Rechteverteilung. Hier können die Benutzer und deren Rechte (Lesen, Schreiben, Löschen, Ändern, Anlegen usw.) definiert werden, diese Rechte lassen sich dann zumindest tabellenweise, teilweise auch bis auf Spaltenebene, vergeben. Dies kann besonders nützlich bei Datenbanken sein, bei denen nicht alle Nutzer in jeder Tabelle lesen bzw. schreiben dürfen.

Etwas anders kann die Situation bei textlastigen Anwendungen wie einem Blog oder einem CMS aussehen. Hier ist vorher schlecht bis gar nicht abzuschätzen, wie viel Text der Autor schreibt. Ist er gerade in Schreiblaune, kann ein Blogbeitrag vielleicht auch mehrere zehntausend Zeichen umfassen. Nun kennen die aktuellen Versionen relationaler Datenbanken auch Felder für lange Texte (also viele Zeichen) und reservieren dafür entsprechend Platz, was aber gerade bei kurzen



Einträgen Verschwendung sein kann. Alternativ kann man den Text natürlich auch auf viele Felder mit einer Länge von z. B. 255 Zeichen aufteilen lassen. Dann benötigt man allerdings eine zusätzliche Tabelle in der Datenbank, in der gespeichert wird, welches Textfeld bzw. Textfragment zu welchem Blogeintrag gehört und in welcher Reihenfolge die Textteile zusammengesetzt werden müssen, was die Datenbank komplexer macht.

Die Komplexität der Datenbank ist ein nicht zu unterschätzender Faktor, gerade für Einsteiger in diesem Bereich. So hat das CMS Typo3, welches zugegebenermaßen recht umfangreich und leistungsfähig ist, in der Grundinstallation rund 40 Tabellen in der Datenbank. Aber auch einfachere Blogsystem und CMS legen in der Regel schon fünf bis zehn Tabellen an.

Zwei weitere, noch erwähnenswerte Vorteile von relationalen Datenbanken sind Transaktionen und Fremdschlüsselbeziehungen, welche nützlich sind, um die Datenbank konsistent zu halten. Transaktion bedeutet, dass mehrere Aktionen (z. B. erst einen Datensatz lesen und darauf basierend zwei neue Datensätze schreiben) zu einer Einheit zusammengefasst werden. Diese Einheit (=Transaktion) kann dann ausgeführt werden, und zwar nur komplett. Das heißt, schlägt eine Aktion fehl, werden alle bisherigen Aktionen wieder rückgängig gemacht [10].

Relationale Open-Source-Datenbanken

Relationale Open-Source-Datenbanken gibt es in diversen Ausführungen, welche alle individuellen Stärken und Schwächen haben. SQLite [11]

ist eine besonders leichtgewichtige und portable Datenbank und daher gerade für „kleine Anwendungen“ und als eingebettete Datenbank populär. MySQL [12] findet man häufig als Datenbank hinter Webanwendungen. Der Tabellentyp „myisam“ ist recht schnell bei Schreib- und Lesevorgängen, bietet aber keine Transaktionen und es fehlt die Implementierung von Fremdschlüsseln. Beide Merkmale sind aber im Tabellentyp „InnoDB“ verfügbar. Weiterhin bietet MySQL relativ einfache Replizierbarkeit, zumindest für eine „Master-Slave-Replikation“. Nachteilig bei MySQL ist, dass der SQL-Standard an einigen Stellen nicht konsequent bzw. gar nicht implementiert ist. PostgreSQL bietet eine fast vollständige Unterstützung des SQL-Standards und auch einige weitere Funktionen, welche gerade in Unternehmensanwendungen relevant und von Interesse sind.

Dokumentenorientierte Datenbanken

Dokumentenorientierte Datenbanken verfolgen einen völlig anderen Ansatz als relationale Datenbanken. Hier werden die zu speichernden Daten nicht in Tabellen abgelegt, sondern in sogenannten Dokumenten. Ein Dokument ist hier nicht im Sinne eines Textdokuments zu sehen, sondern eher als „Sammelmappe“. Innerhalb eines Dokuments können beliebige Felder definiert werden, die sogenannten „Schlüssel“, welchen jeweils einen „Wert“ zugeordnet ist. Der Wert kann dabei in der Regel beliebig sein, also eine Zahl, ein Wort, ein Text, eine Liste von Worten bzw. Zahlen, eine Datei usw. Die Dokumente sind dabei komplett schemalos. Dies bedeutet, dass

man innerhalb eines Dokuments jederzeit beliebige neue Schlüssel mit einem beliebigen Wert anlegen kann. Der Schlüssel muss zwar innerhalb eines Dokuments einmalig sein, kann aber in anderen Dokumenten wieder vorkommen – muss es aber nicht. Ebenso kann der Wert eine beliebige Länge haben, diese muss vorab nicht definiert werden. Dies ist ein fundamentaler Unterschied zur vorab klar zu definierenden und quasi-starren Struktur von relationalen Datenbanken.

Daher eignen sich dokumentenorientierte Datenbanken besonders dann, wenn größere Mengen Text mit unbestimmter Länge zu speichern sind – wie z. B. bei Blogs und Content Management Systemen oder auch Wikis. Der Vorteil hier ist, dass alles, was zu einem Dokument (Wiki-Seite, Blogeintrag etc.) gehört, innerhalb der Datenbank auch in einem Dokument gespeichert werden kann. Dies wären z. B. die Meta-Daten einer Wiki-Seite (Tags, Datum der letzten Änderung etc.) oder bei einem Blog ebenfalls Metadaten oder auch die Kommentare, die Leser zum Blogeintrag hinterlassen haben. Bei relationalen Datenbanken würden diese Daten typischerweise auf mehrere Tabellen verteilt und wären nicht „auf einen Blick“ sichtbar.

Nutzung von dokumentenorientierten Datenbanken

Dadurch, dass die Daten komplett anders abgelegt werden als in relationalen Datenbanken ist bei der Nutzung von dokumentenorientierten Datenbanken vor allem eines wichtig: Umdenken. Was bei MySQL & Co. vielleicht auf mehrere Ta-



bellens verteilt wurde (bzw. werden musste), kann bei Dokumentenorientierung wahrscheinlich in einem Dokument abgelegt werden. Auch die Abfrage der Daten gestaltet sich anders, wie im folgenden Abschnitt erklärt wird, was beim Anlegen der Dokumente natürlich berücksichtigt werden sollte. Denn auch wenn die Datenbank an sich schemalos ist bzw. kein Schema verlangt, sollte man sich natürlich vorab schon Gedanken machen, wie man die Daten innerhalb der Datenbank und den Dokumenten organisiert. Ansonsten ist es später eventuell schwierig, gezielte Abfragen durchzuführen.

Viele dokumentenorientierte Datenbanken verwenden für Abfragen auf die Datenbank einen MapReduce-Ansatz. Diese ursprünglich von Google entwickelte Abfragetechnik [13] teilt die abzufragende Datenmenge in mehrere kleine Teilmengen auf und wendet dann das Abfragekriterium an. Liegen die Ergebnisse vor, ist der Map-Teil abgeschlossen. Anschließend werden die Ergebnisse der verschiedenen Map-Teile zusammengefasst und als Ergebnis ausgegeben, womit auch der Reduce-Teil ausgeführt ist. Der MapReduce-Prozess an sich ist für den Anwender nicht weiter sichtbar, er erhält immer das Ergebnis zurück. Der Vorteil von MapReduce ist, dass sowohl die Map-Prozesse als auch die Reduce-Prozesse (quasi-) parallel ausgeführt werden, was besonders bei größeren Datenmengen Geschwindigkeitsvorteile gegenüber gleichwertigen SQL-Abfragen an relationale Datenbanken bringt. Je nach Implementierung können auch dokumentenorientierte Datenbanken

das Anfragergebnis noch nach bestimmten Kriterien sortieren (aufsteigend, absteigend etc.).

Dagegen ist es in der Regel nur sehr umständlich zu realisieren, den Schlüssel zu einem Wert zu erhalten. Das heißt, Abfragen wie „Zeige alle Schlüsselnamen in allen Dokumenten, wo der Wert *foobar* lautet“ sind aufwendig und müssen in der Regel softwareseitig umgesetzt werden.

Das Datenformat, in der viele dokumentenorientierte Datenbanken die Daten speichern, ist JSON, die „JavaScript Object Notation“ [14] bzw. die binäre Variante „BSON“ [15]. Weiterhin ermöglichen die Datenbanken in der Regel einen einfachen Umgang mit binären Daten wie z. B. Dateien. Diese werden einfach wie jeder andere Wert auch in einem Feld gespeichert. Das

Ergebnis von Abfragen auf die Datenbanken wird dann auch im JSON- bzw. BSON-Format zurückgegeben. Dies hat den Vorteil, dass die allermeisten Programmiersprachen mit JSON-Daten umgehen können.

Dokumentenorientierte Open-Source-Datenbanken

Es gibt eine Reihe von Open-Source-Datenbanken, welche dokumentenorientiert sind. Die zur Zeit wohl am verbreitetsten sind CouchDB [16] und MongoDB [17].

Zu CouchDB gab es in [freiesMagazin](#), Ausgabe 6/2010 [18], einen ausführlichen Artikel. Einige Merkmale von CouchDB sind eine HTTP-basierte API, d.h. Daten können über reguläre HTTP-Requests abgefragt und geschrieben

Key	Value
"Audacity" ID: Audacity	{rev: "1-cd9ddd3cd0d82e4207f34aacbdd00052"}
"CouchDB" ID: CouchDB	{rev: "1-7d9e521da3bf0e75f3d9a72a2eadfab7"}
"Gedit" ID: Gedit	{rev: "1-bab8730b676e3b9c86cdacc95afba681"}
"MS Office" ID: MS Office	{rev: "1-427f40a3d77f31cef6ca8dfef87c244d"}
"_design/fm_bsp" ID: _design/fm_bsp	{rev: "6-c9f056efdda3c2827283a8c3f17a65aa"}
"iLife" ID: iLife	{rev: "3-22cff69263fa8e46eed71a7b1a2052ad"}

Futon, die Weboberfläche von CouchDB. 🔍



werden. Weiterhin ist standardmäßig eine Web-Oberfläche Namens „Futon“ enthalten, über die die Datenbank komplett verwaltet werden kann. Außerdem unterstützt CouchDB die automatische Revisionierung, d. h. bei jeder Änderung an einem Dokument wird eine neue Revision angelegt, die alten Daten bleiben aber erhalten. Dies gilt auch für Dateien, die als binäre Daten in der Datenbank abgelegt werden können. Außerdem ist die automatische Replikation der Datenbank, auch über mehrere Server hinweg und bidirektional, sehr einfach konfigurierbar.

MongoDB hat etwas andere Schwerpunkte als CouchDB. So unterstützt die Datenbank zusätzlich die Indizierung von Dokumenten, was die Abfragen bzw. die Suche innerhalb von Dokumenten schneller macht. Um direkten Zugriff auf die Datenbank zu bekommen, gibt es eine Schnittstelle für die Kommandozeile (ähnlich wie bei z. B. MySQL und SQLite). Zusätzlich stehen natürlich, ebenso wie bei CouchDB, APIs für eine Reihe von Programmiersprachen bereit. Stärken hat MongoDB weiterhin beim Umgang mit großen Datenmengen. Zum einen kann die Datenbank mit Hilfe von GridFS [19] auch sehr große Dateien (z. B. Videos) speichern. Zum anderen unterstützt MongoDB von Haus aus die horizontale Skalierung (das sogenannte „Sharding“ [20]), d. h. mehrere parallel laufende Datenbanken teilen die Anfrage untereinander auf.

Key-Value Stores

Key-Value Stores, in der Kurzform auch nur KV-Store genannt, speichern Daten in Form von

Schlüssel-Werte-Paaren. Jeder Schlüssel kann ein beliebiger, aber eindeutiger (=einmaliger) String sein. Was als Wert zulässig ist, hängt vom verwendeten KV-Store ab. Zulässig ist immer ein String; einige Programme unterstützen aber auch binäre Daten, Listen und Sets. Auf eine relationale Datenbanken übertragen kann man sich KV-Stores als eine Datenbank mit genau einer zweiseptigen Tabelle vorstellen. Die erste Spalte enthält den Schlüssel, welcher auch der Index ist, die zweite Spalte speichert den Wert.

Schaut man auf die Homepage der diversen KV-Stores, so fällt auf, dass so gut wie alle Benchmarkergebnisse auf die Startseite verlinken. Dies ist in sofern bemerkenswert, dass man bei relationalen oder dokumentenorientierten Datenbanken dafür in der Regel tiefer in die Homepage hinabsteigen muss – wenn überhaupt Benchmarks angegeben werden. Der Grund hierfür ist aber relativ simpel: KV-Stores sind schnell, sehr schnell. Eine 6-stellige Anzahl an Schreib-/ Leseoperationen pro Sekunde ist nicht unüblich – und das auf „normaler“ Hardware, also nicht auf speziellen Datenbankservern. Manche KV-Stores sind beim Schreiben sogar schneller als beim Lesen, was bei relationalen und dokumentenorientierten Datenbanken nie der Fall ist. Auch hier ist der Grund recht offensichtlich: KV-Stores haben aufgrund der einfachen Struktur sehr wenig „Overhead“ beim Zugriff auf die Datenbank, im Gegensatz zu den beiden anderen Datenbanktypen. Es gibt keine Verwaltungstabellen (oder Dokumente), die die Datenbank im Hintergrund aktuell hal-

ten muss, es müssen keine Indexe aktualisiert werden usw. Bei Schreiboperationen in einem KV-Store wird im günstigsten Fall nur geprüft, ob der zu schreibende Schlüssel schon existiert und dann das neue Schlüssel-Werte-Paar ans Ende der Datenbank angehängt.

Nutzung von Key-Value Stores

KV-Stores eignen sich besonders für Anwendungen, wo die Daten gut in Schlüssel-Werte-Paaren organisiert werden können. Dies ist z. B. bei Microblogging-Diensten der Fall. Hier kann der Schlüssel z. B. eine Kombination aus fortlaufender Nummer des Beitrags, Nutzernamen und Zeitstempel sein. Ein sehr gutes und absolut lesenswertes Beispiel findet man in der Dokumentation von Redis, welches weiter unten noch vorgestellt wird. In diesem Beispiel wird sehr anschaulich und nachvollziehbar erklärt, wie man die Daten für einen Twitter-Klon sinnvoll und effektiv in einem Key-Value Store organisiert [21]. Ein weiterer Vorteil ist auch hier die Geschwindigkeit von KV-Stores. Viele quasi-parallele Schreib- und Lesezugriffe sind kein Problem.

Aber der Einsatz von KV-Stores ist natürlich nicht auf Microblogging limitiert. Andere denkbare Anwendungen sind z. B. Notizbücher, bei denen der Schlüssel die Überschrift (oder der Zeitstempel) der Notiz ist und der Wert die Notiz an sich. Im gewerblichen Umfeld können KV-Stores z. B. dafür genutzt werden, Prüfwerte einer Serienfertigung oder ähnliches zu speichern. Der Schlüssel ist dann die Seriennummer oder die fortlaufende Nummer des produzierten Produkts, der Wert



die Prüfwerte. Aufgrund der Geschwindigkeit der Datenbank entsteht hier auch kein „Stau“ beim Schreiben der Daten, selbst bei sehr schnellen Fertigungsprozessen nicht.

Was bei der Nutzung bei KV-Stores wichtig ist: Die Schlüssel müssen keine einfachen Textstrings (wie z. B. „abc“ oder „eintrag1“) sein, sondern können ebenfalls ein Schema haben; gleiches gilt für den Wert. Dazu ein kurzes und einfaches Beispiel: Ein einfaches Blogsystem, bei dem mehrere Autoren schreiben, soll die Daten in einem KV-Store speichern. Im Schlüssel soll das Erstellungsdatum und der Vorname des Autors stehen, im Wert der Titel und der Text. Schlüssel und Wert könnten dann z. B. so aussehen:

Schlüssel: 2010-6-23:Jochen

Wert: KV-Stores|KV-Stores ist die Kurzform für Key-Value Stores ...

Der „Trick“ ist, dass man einfach ein beliebiges, aber eindeutiges Trennzeichen einfügt. In diesem Beispiel sind das der Doppelpunkt „:“ im Schlüssel und das Pipe-Zeichen „|“ im Wert. Nach dem Auslesen des Schlüssel- und Wert-Strings kann man diese dann einfach mittels der split-Funktion der genutzten Programmiersprache in zwei Strings aufteilen und weiter verarbeiten. Diese Methode des „sprechenden Schlüssels“ wird auch im oben genannten Beispiel ausführlich erklärt.

Bei KV-Stores ist es nicht möglich, einen zu einem Wert zugehörigen Schlüssel suchen zu lassen. Es wird grundsätzlich immer ein Schlüssel abgefragt und der zugehörige Wert zurückgegeben.

Open-Source-Key-Value-Stores

Einer der ältesten, noch aktiv genutzten KV-Stores ist BerkeleyDB, welches heute zu Oracle gehört [22]. BerkeleyDB kommt oft als eingebetteter KV-Store zum Einsatz oder dient im Hintergrund zum Speichern auf die Festplatte, während der vom Programm/Nutzer direkt angesprochene KV-Store im Speicher läuft. So ist z. B. MemcachedDB [23] aufgebaut.

Ein Key-Value Store mit erweiterter Funktionalität und sehr aktiver Entwicklung ist Redis [24]. Redis kann als Wert nicht nur Strings speichern, sondern auch Listen und Sets. Die Größe eines Werts kann bis zu 1 GB betragen, was für die meisten Fälle ausreichen sollte. Weiterhin beherrscht Redis Operationen mit Sets (z. B. Schnittmengen berechnen) und Sortierung von Listen und Sets nach vorzugebenden Kriterien. Praktisch ist auch, dass man Werte, sofern es sich um eine Ganzzahl handelt, um einen beliebigen Wert oder um eins erhöhen oder verringern kann. So lassen sich sehr einfache Zähler mit Bordmitteln innerhalb der Datenbank realisieren, z. B. für die aktuelle, fortlaufende Nummer eines neuen Eintrags in einem Microblog. Redis hat weiterhin eine Schnittstelle für die Kommandozeile, kann also aus der Shell heraus interaktiv genutzt werden. Redis hält die Daten im RAM

des Rechners, man kann diese aber natürlich auch auf Festplatte schreiben. Wie oft dies geschieht ist konfigurierbar. Das Schreiben erfolgt dabei asynchron im Hintergrund; die Datenbank ist währenddessen voll erreichbar, es werden keine Sperren gesetzt.

Tokyo Cabinet [25] verfolgt einen anderen Ansatz. Hierbei handelt es sich um einen reinen Server, der einen KV-Store bereitstellt. Interessant ist dabei, dass Tokyo Cabinet verschiedene Implementierungen des KV-Stores beherrscht. Diese unterscheiden sich primär darin, wie der Schlüssel indiziert wird und wie komplex der Wert sein kann bzw. darf. Die Unterschiede und Vorteile werden auf der Homepage ausführlich erklärt. Somit kann man bei Tokyo Cabinet immer die für die eigene Anwendung optimale Implementierung auswählen. Weiterhin gibt es für Tokyo Cabinet verschiedene Erweiterungen, wie z. B. eine optimierte Volltextsuche über Werte. Die Erweiterungen sind ebenfalls auf der Homepage verfügbar.

Keyspace [26] ist ein weiterer KV-Store. Hier liegt ein Schwerpunkt auf guter und einfacher Replizierbarkeit und horizontaler Skalierbarkeit. Daher eignet sich Keyspace z. B. für Datenbanken, die eine sehr hohe Ausfallsicherheit und sehr hohe Erreichbarkeit haben müssen. Interessant ist weiterhin, dass Keyspace auch eine HTTP-basierte API bietet. Abfragen werden als HTTP-Request gesendet; das Ergebnis erhält man entweder als Text, JSON oder HTML-Seite zurück.



Weitere Datenbanken

Die oben genannten Datenbanken aus den verschiedenen Bereichen, insbesondere KV-Stores und dokumentenorientierte, stellen nur eine Auswahl und keine vollständige Übersicht dar. Außerdem gibt es auch noch weitere „NoSQL“ Datenbanken mit anderen Datenstrukturen. Erwähnt werden soll hier noch BaseX, eine an der Universität Konstanz entwickelte XML-Datenbank [27]. Wer ausschließlich mit XML-Daten arbeitet, sollte sich die Datenbank näher anschauen.

Die englischsprachige Wikipedia widmet dem Thema NoSQL eine eigene Seite [28], auf der die Historie von NoSQL dargestellt, verschiedene Techniken erläutert und diverse Datenbanken genannt werden. Weiterhin findet man eine ausführliche Übersicht über diverse NoSQL-Datenbanken auf der Seite „NoSQL Database“ [29].

Zusammenfassung

Wer heute eine Datenbank benötigt, hat die Auswahl zwischen verschiedenen Datenbanktechniken. Somit ist es ohne Weiteres möglich, eine für die eigenen Bedürfnisse und Anwendungen optimale Lösung zu finden. Die Gründe dafür können vielfältig sein: Art der zu speichernden Daten, Geschwindigkeit der Datenbank auf der zur Verfügung stehenden Hardware, der Wunsch nach hoher Erreichbarkeit usw. SQL-basierte, relationale Datenbanken sind aber keinesfalls „aus der Mode“ – es gibt immer noch genug Anwendungsfälle, wo diese die beste Lösung darstellen. Nicht zu unterschätzen ist auch, dass SQL

schon lange existiert, d. h. es gibt hier eine Vielzahl von Entwicklern und Lösungen für dieses Gebiet. Fest steht aber auch: Es gibt Datenbanken jenseits von SQL, welche sehr gute Ansätze und optimierte Lösungen für bestimmte Aufgaben bieten. Der wichtigste Punkt ist dabei wahrscheinlich: Nicht die von SQL und relationalen Datenbanken bekannten Lösungen und Schemata übertragen, sondern sich den alternativen Möglichkeiten öffnen, um diese optimal und mit allen Vorteilen nutzen zu können.

LINKS

- [1] http://de.wikipedia.org/wiki/System_R
- [2] <http://de.wikipedia.org/wiki/SQL>
- [3] <http://labs.google.com/papers/bigtable.html>
- [4] <http://github.com/twitter/gizzard/>
- [5] <http://hadoop.apache.org/>
- [6] <http://de.wikipedia.org/wiki/Skalierbarkeit>
- [7] <http://de.wikipedia.org/wiki/Entity-Relationship-Modell>
- [8] <http://de.wikipedia.org/wiki/Fremdschlüssel#Fremdschlüssel>
- [9] <http://de.wikipedia.org/wiki/Datenbanknormalisierung>
- [10] http://de.wikipedia.org/wiki/Transaktion_%28Informatik%29
- [11] <http://www.sqlite.org/>
- [12] <http://dev.mysql.com/>
- [13] <http://de.wikipedia.org/wiki/MapReduce>
- [14] <http://www.json.org/>
- [15] <http://bsonspec.org/>
- [16] <http://couchdb.apache.org/>

- [17] <http://www.mongodb.org/>
- [18] <http://www.freiesmagazin.de/freiesMagazin-2010-06>
- [19] <http://www.mongodb.org/display/DOCS/GridFS>
- [20] <http://www.mongodb.org/display/DOCS/Sharding>
- [21] <http://code.google.com/p/redis/wiki/TwitterAlikeExample>
- [22] <http://www.oracle.com/technology/products/berkeley-db/index.html>
- [23] <http://memcachedb.org/>
- [24] <http://code.google.com/p/redis/>
- [25] <http://1978th.net/tokyocabinet/>
- [26] <http://scalien.com/keyspace/>
- [27] <http://www.inf.uni-konstanz.de/dbis/basex/>
- [28] <http://en.wikipedia.org/wiki/NoSQL>
- [29] <http://nosql-database.org/>

Autoreninformation



Jochen Schnelle hat diverse Intranet-Projekte mit MySQL als relationaler Datenbank umgesetzt und nutzt Internetanwendungen, welche ebenfalls MySQL nutzen (Typo3, Roundup). Über die dokumentenorientierte Datenbank CouchDB wurde sein Interesse für NoSQL geweckt.

Diesen Artikel kommentieren

MeeGo im Blick von Thorsten Schmidt

Ende Mai dieses Jahres erschien Version 1.0 der Distribution MeeGo. MeeGo orientiert sich vornehmlich am Einsatz auf mobilen Geräten, wie Handhelds und Netbooks. Dabei setzt MeeGo auf eine Benutzeroberfläche, die in puncto Benutzbarkeit und Benutzerführung neue Akzente setzen will, gerade auf den genannten Geräteklassen. Hört sich interessant an? Ist es auch.

Vorspiel

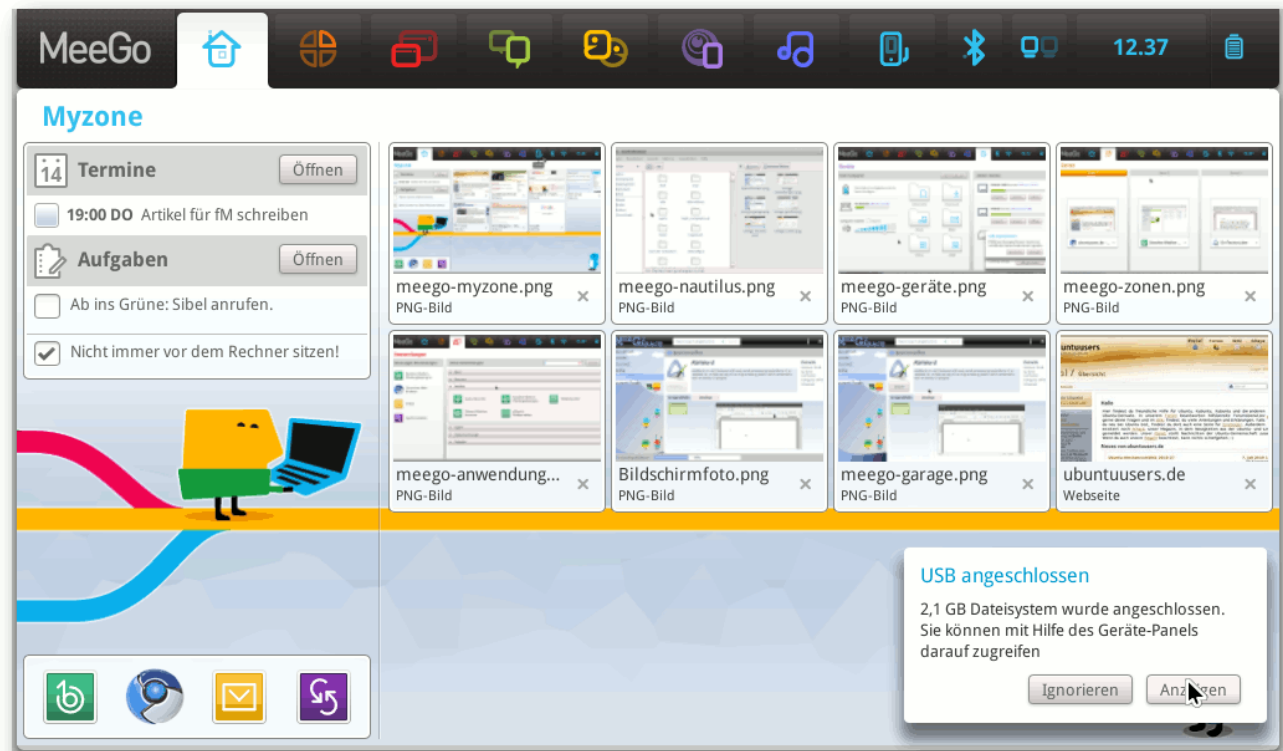
Hervorgegangen ist MeeGo [1] aus der Kooperation von Intel und Nokia. Beide Unternehmen entwickelten ihre beiden Plattformen Moblin und Maemo lange Zeit nicht nur unabhängig voneinander, sondern auch getrennt von den beiden anderen großen Konsorten – der Limo-Foundation [2] auf der einen Seite und der Open Handset Alliance [3] auf der anderen Seite. MeeGo steht nun unter der Schirmherrschaft der Linux-Foundation [4]. Das Ziel des Zusammenschlusses dürfte klar sein: eine bessere Unterstützung der eigenen Hardware durch Vereinigung der zuvor getrennten Kräfte zu erreichen.

Dabei richtet sich die Version 1.0 von MeeGo vor allem an Entwickler. Wie Imad Sousou und Valtteri Halla von der MeeGo Technical Steering Group selbst in ihrer Verlautbarung zur Veröffentlichung auf den MeeGo-Projektseiten [5] schreiben: „This release provides developers with a stable core foundation for application development and a rich user experience for Netbooks.“

Vermutlich stellt MeeGo also eine Art Referenzmodell dar: zukünftigen Herstellern eben erwähnter Geräteklassen, die auf Intel und ARM setzen, soll gezeigt werden, was mit einem Open-Source-Entwicklungsmodell möglich ist. Einen Überblick über das Entwicklungsmodell MeeGos gibt die folgende PDF-Datei [6]. Da aber die Zielgruppe dieser Distribution schlussendlich ganz klar bei den Endbenutzern von mobilen Geräten liegt, ist ein Ausprobieren natürlich interessant. Ob sich

MeeGo 1.0 schon für den alltäglichen, produktiven Einsatz eignet, wird man gleich sehen.

Als Hardwarevoraussetzungen für MeeGo wird auf den Projektseiten – wie erwartet – von Netbooks mit Intel Atom-Prozessoren und den ARM-basierten Nokia N900 Handhelds u. a. berichtet. Über den benötigten Speicherplatz auf der Festplatte und die Systemanforderungen in puncto Arbeitsspeicher werden keine Worte



Startseite „myzone“: Alles auf einen Blick. 



verloren. Allerdings verweist man auf den MeeGo-Projektseiten auf eine Liste mit erfolgreich benutzter Hardware [7].

Testumgebung

Als Grundlage dient ein Acer Aspire One (A 110 L) mit einer 8 GB SSD und einer 4 GB Speicherkarte. Hardwareprobleme sollte es damit also keine geben. Herunterladen kann man MeeGo ebenfalls auf den Projektseiten [8]. Die IMG-Datei liegt dabei in zwei Versionen vor: einmal als Version mit dem Google Chrome Browser (hier wird ein Akzeptieren der EULA nötig) und ein anderes Mal in der Version mit dem Chromium-Browser. Als Testversion wird hier die letztere Version benutzt.

Installation

Nach dem Herunterladen muss man die IMG-Datei als erstes per dd auf den USB-Stick bringen. Hat man das erledigt, kann es auch schon losgehen – reinstecken und booten: Im ersten Bildschirm nimmt man die Auswahl „Installation only“. Dann erscheint der Begrüßungsbildschirm „MeeGo Rocks“ – ein großes Versprechen ;).

Anschließend folgt die Installationsroutine, die über einen grafischen Installer abgewickelt wird. Hier sieht alles sehr gepflegt und ziemlich geradlinig aus. Es folgen die bei einer Installation typischen Schritte: Zunächst erscheint die Sprachauswahl (16 Sprachen stehen hier zur Verfügung). Danach folgt das Partitionierungswerkzeug: mehrere Möglichkeiten stehen zur Wahl. Wie es so Usus ist, reichen die Optionen vom Er-

stellen eines eigenen Partitionslayouts über das automatische Einrichten auf dem freien Platz der Festplatte bis hin zu „Linux-Partitionen auf Platten löschen und Standard-Layout erstellen“. Für Letzteres wurde sich hier entschieden – der Einfachheit wegen. Die Aufteilung des Standardlayouts sieht dann wie folgt aus:

sda1: /boot (ext3)

sda2: / (btrfs)

Die Speicherkarte mmcblk0p wurde mit einer Partition für /home mit dem Dateisystem btrfs

versehen (das musste aber selbst erzeugt werden). Vor der Partitionierung fragt das Werkzeug noch zwei Dinge ab: ob ein Passwort für den Bootloader gesetzt werden und welche Betriebssysteme noch in der Bootloader-Auswahl erscheinen sollen. Beide Fragen wurden verneint. Insgesamt dauerte die Installation gut 10 Minuten. Nach einem Neustart erscheint wiederum ein Begrüßungsbildschirm. Nun müssen noch die üblichen Einstellungen, wie Tastaturlayout, Zeitzone, Datum und Uhrzeit vorgenommen werden. Anschließend folgt die Frage, ob man willens ist, zur „Backtrace-Sammlung“



Einfache und übersichtliche Installation von Abiword in der Garage. 🔍



beizutragen – heißt konkret, hier werden Signaturen von Anwendungsabstürzen gespeichert und den Entwicklern zugesendet. Allerdings wird auch darauf hingewiesen, dass „Benutzereingaben zusammen mit dem Backtrace eingeschickt“ werden. Schließlich folgt noch die Benutzererstellung. Parallel dazu erhält man noch die Option, mit einem Klick auf eine Checkbox zu entscheiden, ob die Daten verschlüsselt werden sollen – sämtliche Details zu Fragen, was wie verschlüsselt werden soll, bleiben unklar. Danach ist man schon fertig.

Im Nachhinein stellt sich dann heraus, dass MeeGo per `ecryptfs-utils` das Homeverzeichnis des Nutzers verschlüsselt. Hier werden die Daten nach dem Login transparent verschlüsselt, das heißt, erst nach dem Abmelden und/oder Herunterfahren des Systems sollten sich die Daten im verschlüsselten Zustand befinden.

Im Einsatz

Startet man MeeGo das erste Mal, darf man über die Bootzeit erfreut sein: vom Einschalten bis zur grafischen Oberfläche vergehen gerade mal 18 Sekunden (ohne Grub). Nicht dass das entscheidend wäre, aber schön ist es schon. Ein Blick auf den verwendeten Kernel zeigt die Version 2.6.33 an. Ein weiterer Blick auf den Speicherverbrauch sagt, dass bei einem geöffneten Terminal (GNOME -Terminal 2.30) gut 274 MB von schmalen 512 MB verbraucht werden. Das ist eine ganze Menge, und man ahnt, da bleibt nicht viel Platz für alles Weitere, was man so benutzen

will. Startet man dann noch den Standardbrowser Chromium (V. 6.0) steigt die Speicherauslastung auf 419 MB. Natürlich braucht es dann nur noch zwei bis drei weitere Programme und schon wird die Oberfläche langsam. Bis dahin aber reagiert MeeGos Benutzeroberfläche sehr flüssig – gefühlt jedenfalls.

Was die mitgelieferten Programme angeht, hat man bei MeeGo vom Umfang her auf ein angepasstes GNOME 2.30 gesetzt. Ausnahmen sind auf den ersten Blick die Aufgabenverwaltung Task (aus dem Pimlico-Projekt [9]) sowie das schon erwähnte Chromium und das MeeGo-Projekt „Garage“, mit dem sich speziell für MeeGo angepasste Software auf leichte Weise mit einem Klick installieren und auch wieder entfernen lässt.

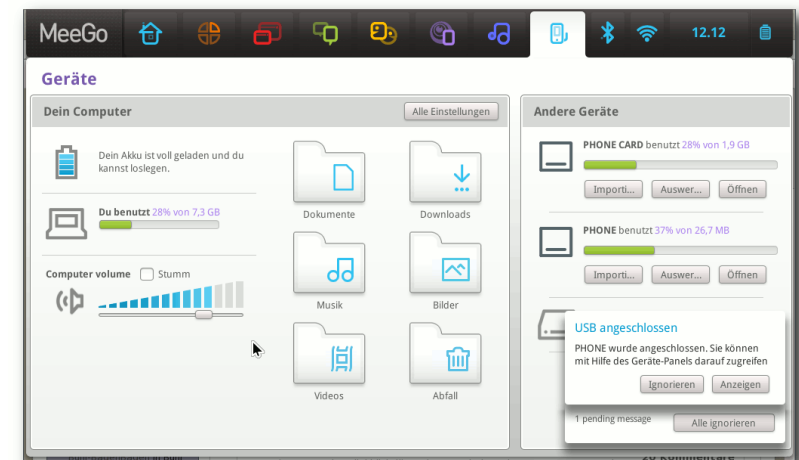
Aus Sicht des Endanwenders ist Garage in etwa identisch zum Softwarecenter unter Ubuntu (ab Version 9.10). Interessanterweise findet man mit `gpk-application` ein ähnliches Werkzeug unter „Anwendungen → Systemwerkzeuge“. Hinter den beiden Frontends wird wohl `yum` [10] als Paketmanagement arbeiten. So lässt sich die Paketverwaltung auch im Terminal bedienen. Folglich hat man hier schon drei Werkzeuge, die in etwa das Gleiche erledigen – nur mit verschiedenem Funktionsumfang. Wie sich später, aufgrund eines Problems mit `yum` wie auch dem

Frontend `gpk-application`, herausstellte, verfügt MeeGo auch noch über Zypper. Damit wäre man dann bei vier Werkzeugen angelangt. Wie auf der Mailingliste `meego-dev@meego.com` [11] nachzulesen ist, will man sich mit `yum` und `Zypper` zunächst beide Wege offen halten. Tendenziell lieber ärgert man aber mit `Zypper`.

Die Benutzeroberfläche

Offensichtlich hat man während der Entwicklung des Bedienkonzeptes viele – wenn nicht alle – Ideen von Moblin übernommen. Die Oberfläche „MeeGo UX“ stellt dabei eine Fortentwicklung des schon unter Moblin verwendeten, auf Clutter [12] basierenden Mx-Toolkits [13] dar.

Wichtigstes Navigationsmittel ist die horizontale Werkzeugleiste, auf deren Oberfläche in der Standardeinstellung zehn Icons den Zugriff zu wichtigen und häufig genutzten Aktionen möglich



Einfacher Zugriff und Überblick auf eingebundene Geräte. 🔍



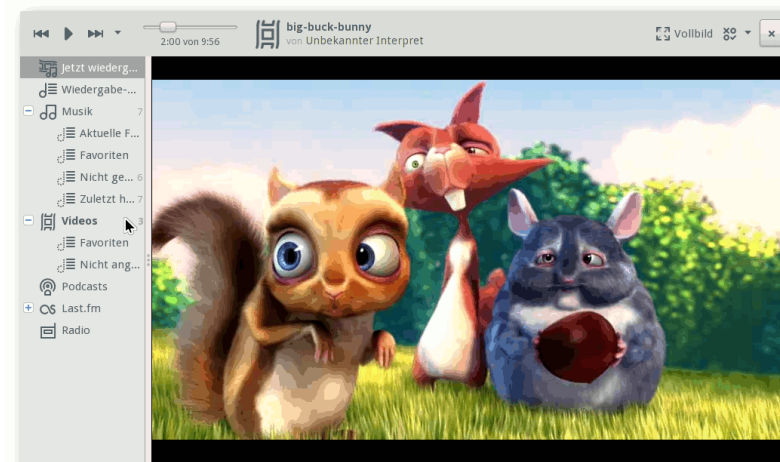
machen. Aktion ist hier mit Absicht gewählt, denn diese Icons verweisen auf – so kann man sagen – Zusammenfassungen. Zum Beispiel ruft das Icon „myzone“ die Startseite des Nutzers auf – hier werden Termine und Aufgaben, zuletzt besuchte Webseiten und bevorzugte Anwendungen zusammengefasst dargestellt.

Oder ein anderes Beispiel: Ein Klick auf das Icon „Geräte“ führt zu einer Überblickseite – Akku-stand, Speicherverbrauch auf der SD-Karte und Lautstärke, häufig benutzte Ordner im Homeverzeichnis und eingebundene Geräte, also USB-Sticks und so weiter werden angezeigt.

Nach einem Klick auf ein Element in dieser Ansicht verschwindet die eben erwähnte Werkzeugleiste und die Anwendung füllt dann den ganzen Bildschirmplatz aus. Führt man die Maus

zurück an den oberen Bildschirmrand, taucht die Werkzeugleiste wieder auf, und eine Navigation zu anderen Plätzen wird möglich. Zwischen den Anwendungen wechselt man über einen Klick auf das Icon „Zonen“ in der Werkzeugleiste. Auch hier wird man auf eine Zusammenfassung der laufenden Anwendungen geführt, zwischen denen man dann wählen kann. Alternativ kann natürlich mit **Alt** + **Tab** zwischen den Anwendungen gewechselt werden.

Damit ist grundsätzlich das Bedienkonzept der Oberfläche von MeeGo beschrieben. Weitere Informationen findet man auf den MeeGo-Seiten [14].



Der Medienabspieler Banshee im Einsatz. 🔍

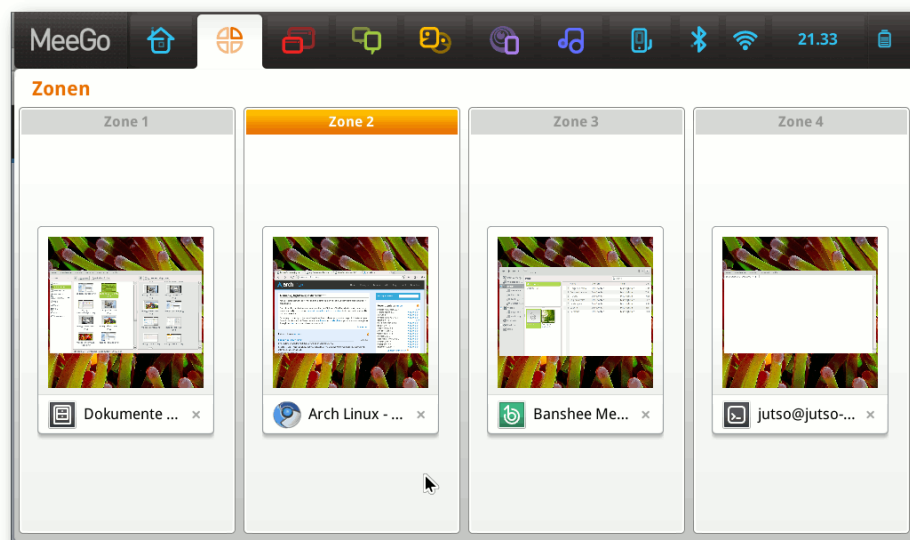
meego-core, meego-core-updates, meego-extra, meego-extra-updates, meego-netbook, meego-netbook-updates, meego-non-oss, meego-non-oss-updates und adobe-linux-i386. Während der Benutzung kamen auch Aktualisierungen herein – so zum Beispiel das Update auf die MeeGo Version 1.01 am 07.07.2010. Wie die Liste der mitgebrachten Änderungen des Updates auf den MeeGo-Projektseiten [15] [16] [17] zeigt, handelt es sich dabei vor allem um Fehlerkorrekturen, Verbesserungen und auch Sicherheitsupdates.

Anwendungen für Alltägliches

In Sachen Video und Audio kommt MeeGo nach einer Standardinstallation mit dem Abspielen von Ogg-Vorbis und Ogg-Theora zurecht – alle anderen wichtigen Codecs sind nicht enthalten. Sehr wahrscheinlich hat dies seine Gründe im üblichen Dickicht von Patenten und Lizenzen.

Und Updates?

Da die momentan veröffentlichte Version in erster Linie für Entwickler gedacht ist, stellt sich natürlich die Frage, inwiefern MeeGo schon jetzt für Endanwender und deren Belange geeignet sein kann. Aktualisierungen für Fehlerkorrekturen und Sicherheitsupdates sind das Stichwort: Einige Repositories stehen natürlich schon bereit. Blickt man unter `/etc/yum.repos.d:`



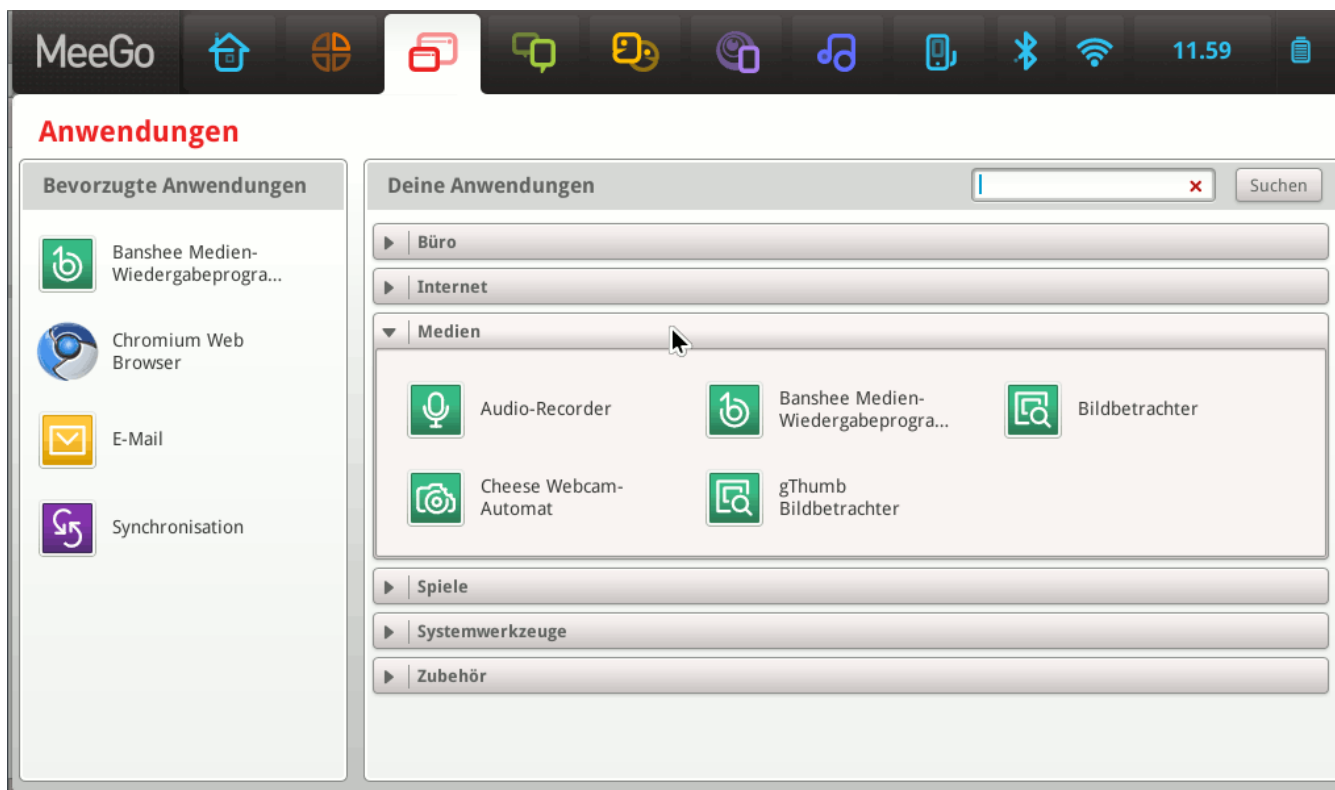
Zwischen Anwendungen wechselt man in „Zonen“. 🔍



Ein Nachinstallieren ist natürlich möglich, aber ob es zum Erfolg führt bleibt ungewiss. Der Einfachheit halber wurde hier der kostenlos herunterladbare MP3-Codec von Fluendo [18] ausprobiert. Der Codec steht dort als RPM-Datei zum Download bereit. Für alles Weitere müssen zwei bis drei Handgriffe mehr angebracht werden, wenn man zum Beispiel Totem samt diverser Plugins installieren will, wie es im MeeGo-Forum [19] gezeigt wird.

Will man Audio- und Videodateien abspielen, so steht einem als Medienabspieler Banshee [20] in Version 1.6.1 zur Verfügung. Das funktionierte – soweit ausprobiert – ohne Probleme. Ob man Banshee nun gut findet, ist natürlich Geschmackssache. Allerdings scheint die Einbettung der Benutzeroberfläche in die Zusammenfassung „Media“ von MeeGo ganz gut gelungen.

Ansonsten steht für E-Mail & Co. die Anwendung Evolution (Version 2.30.1) zur Verfügung. Damit lässt sich auch die Verwaltung von Kalendereinträgen und RSS-Feeds bewerkstelligen – und MeeGo integriert diese auch in die Benutzeroberfläche. Abgesehen von der Terminverwaltung, die ja von der Anwendung Tasks (s. o.) übernommen wird. Ach ja, Thunderbird lässt sich ohne Probleme nachinstallieren. Für alles rund um Instant Messaging steht Empathy bereit. Alle anderen Anwendungen orientieren sich wohl am Umfang der GNOME Desktopumgebung. So z. B. auch Abiword, das sich mit einem Klick über MeeGos Garage installieren lässt.



Zusammenfassung und Ausgangspunkt zum Start installierter Anwendungen. 🔍

Sind die Programme nicht in den Repositorien enthalten – und dort ist zurzeit nicht viel vorhanden – bleiben einem nur die zwei üblichen Wege:

Entweder besorgt man sich den Quelltext (z. B. auf MeeGo.gitorious.org [21]) oder auf den Seiten des jeweiligen Projektes und kompiliert sich die ausführbare Datei, in der Hoffnung alle Abhängigkeiten bereitzustellen und dass es dann auch funktioniert. Oder man lädt sich direkt RPM-Pakete herunter, auch wieder in der Hoffnung, dass das dann funktioniert.

Für soziale Netzwerke haben sich die Entwickler von MeeGo etwas Besonderes ausgedacht: Über das Icon „Status“ ist eine Zusammenfassung der genutzten „Web-Services“ zu erreichen. Last.fm und Twitter stehen in der Vorauswahl bereit und ein Konto lässt sich auf schnelle Weise hinzufügen.

Gleiches gilt für das Thema „Synchronisieren“ – hier lassen sich Kontakte, Termine etc. über einen Dienst aus voreingestellte Dienste, wie z. B. Google, Funambol, Meemento usw. auswählen.



Die Anwendung findet man unter „Anwendungen → Internet → Synchronisation“, lässt sich aber auch in die Werkzeuggeste transferieren.

Anmerkungen

Alles in allem – und wie erwartet – erweckt MeeGo den Eindruck, dass es sich im Entwicklungsstadium befindet. Klar, man kann es schon jetzt nutzen. Ob man im Zuge der weiteren Entwicklung mit unangenehmen Überraschungen rechnen muss, bleibt abzuwarten.

Obwohl die Benutzeroberfläche vom Aussehen und der Bedienung ganz passabel erscheinen, sind doch einige Schwächen nicht zu übersehen: Da wäre zum Beispiel die unter dem Chromium-Browser fast schon als Zierleiste anmutende Leiste der Bedienoberfläche zu erwähnen: Ihre einzige Aufgabe besteht darin, den Titel der Webseite anzuzeigen, also genau das, was man schon in den Tabs von Chromium selbst lesen kann. Hier wäre eine Integration der Browsertabs in die Leiste sinnvoll – was Platz schaffen würde.

Nun nutzen natürlich auch andere Programme ohne Tabfunktion diese Leiste, und bei diesen hat die eben genannte Leiste dann einen Zweck: So wird z. B. bei Abiword der Dateiname des geöffneten Dokuments angezeigt – zur Orientierung sinnvoll bei mehreren geöffneten Dokumenten. Eine Lösung dieses „Problems“ wird wohl nicht ganz einfach sein.

Noch einmal das Stichwort „Platz“: Auf der Werkzeuggeste zeigt sich am linken Ende der MeeGo-Schriftzug. Will man die Werkzeuggeste um ei-

nige Einträge erweitern, hat man den verfügbaren Platz schnell ausgereizt. Leider lässt sich der Schriftzug „MeeGo“ nicht entfernen. Genau das aber würde den notwendigen Raum für weitere Einträge bringen.

Natürlich sind noch andere, kleine Fehler aufgetreten: Schiebt man unter Netzwerke den Regler „Offline-Modus“ ganz nach links, sollte das alle Verbindungen deaktivieren – es wird aber nichts deaktiviert. Steckt man einen 512-MB-USB-Stick ein, wird dieser als 2,1-GB-Gerät erkannt. Ein wenig rätselhaft ist es auch, dass CUPS über xinetd auf Port 631 lauscht (zufällig gesehen bei einer Netzrecherche im Blog von Andre Bauer [22]):

Active Internet connections (only servers)									
Proto	Recv-Q	Send-Q	Local Address	Foreign Address	State	User	Inode	PID/Program name	
tcp	0	0	0.0.0.0:631	0.0.0.0:*	LISTEN	0	2031	533/xinetd	

Was einen auch noch wundert: MeeGo besitzt keine Schaltfläche für das Herunterfahren des Systems. Stattdessen setzt MeeGo auf den den ursprünglich zum Anschalten vorgesehenen Hardware-Button, der in diesem Fall für das Ausschalten herhalten muss. Weiß man das nicht, sucht man vergeblich und greift zum Terminal.

Apropos „Terminal“: Jenes muss man auch genau dann bemühen, wenn man einen Neustart des Systems herbeiführen möchte – verwunderlich: Es gibt keine Schaltfläche für einen Neustart, wie man das von anderen Distributionen und Betriebssystemen gewohnt ist.

Ebenso ist die Möglichkeit, mehrere Benutzer auf einem System einzurichten, zurzeit nicht vor-

handen. Wie man auf der Mailingliste „MeeGo-dev“ [23] im Mai dieses Jahres diskutierte, könnte sich das aber im Laufe der Entwicklung ändern. Zumindest hat man die fehlende Möglichkeit auf den Projektseiten als Bug [24] identifiziert.

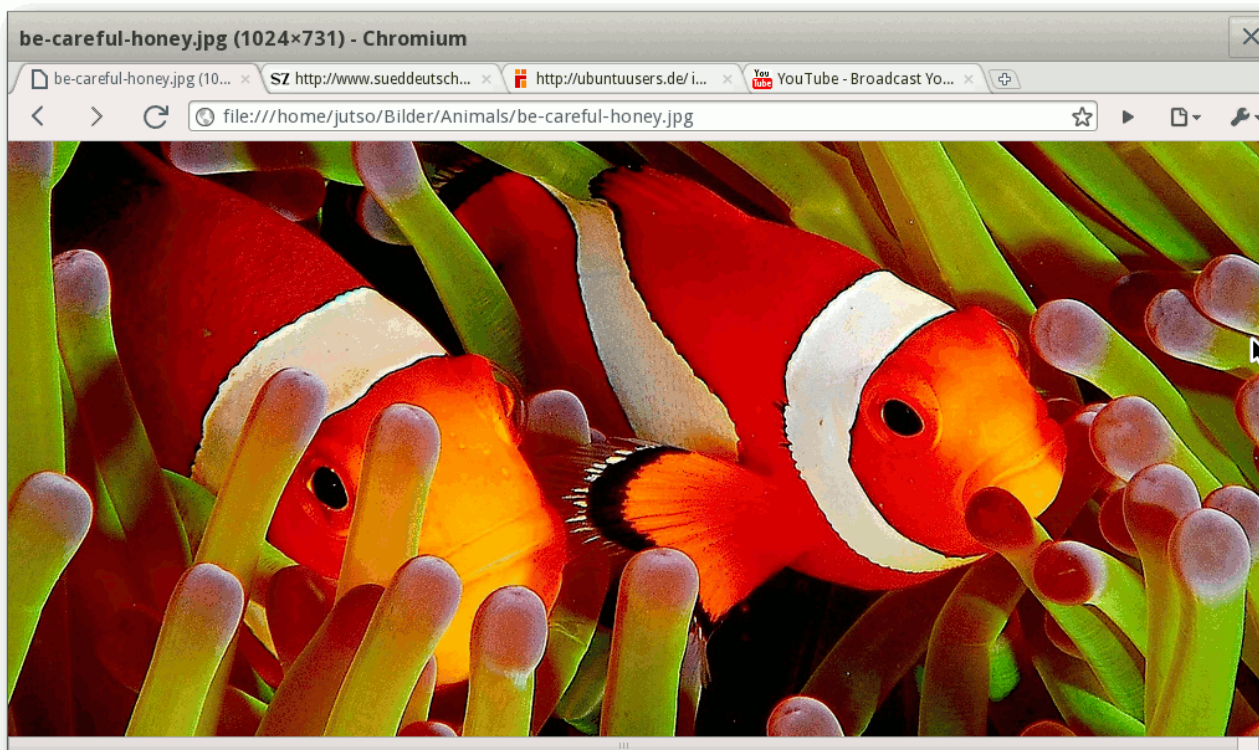
Bei einer längeren Nutzung wären einem bestimmt noch einige andere Sachen aufgefallen – sei's drum, das sind wohl die eben erwähnten, unangenehmen Überraschungen, mit denen man rechnen sollte. Schließlich richtet sich MeeGo noch nicht an Endbenutzer.

Vielleicht noch eine nicht ganz unwichtige Sache am Ende: Nutzer von Netbooks mit einem Intel

Poulsbo (GMA 500) Grafikchipsatz könnten mit MeeGo eine kleine Enttäuschung erleben. MeeGo wird aus lizenzrechtlichen Gründen nicht mit dem passenden Intel-Treiber ausgeliefert. Mehr dazu erfährt man auf Phoronix [25]. Allerdings steht der Treiber auf den Seiten von Intel zum Herunterladen bereit.

Fazit

Man darf gespannt sein, was MeeGo für den Endnutzer in den nächsten Monaten bringt. Wie auf den Seiten des MeeGo-Projektes nachzulesen ist, sind neue Versionen im Abstand von sechs Monaten zu erwarten – die nächste Version 1.1 soll im Oktober veröffentlicht werden. Bis dahin braucht man nicht warten – MeeGo sollte man



Doppelt angezeigte Webseitentitel. 🔍

einfach mal ausprobieren. Weitere Informationen, Hilfestellungen, Tipps und Tricks bekommt man im MeeGo-Forum [26].

LINKS

- [1] <http://meego.com/> 🇬🇧
- [2] <http://www.limofoundation.org/> 🇬🇧
- [3] <http://www.openhandsetalliance.com/> 🇬🇧
- [4] <http://www.linuxfoundation.org/> 🇬🇧
- [5] <http://meego.com/community/blogs/imad/2010/meego-v1.0-core-software-platform-netbook-user-experience-project-release> 🇬🇧
- [6] http://wiki.meego.com/images/MeeGo_Introduction.pdf 🇬🇧
- [7] <http://meego.com/devices/netbook/supported-hardware-platforms>
- [8] <http://meego.com/downloads> 🇬🇧
- [9] <http://www.pimlico-project.org/> 🇬🇧
- [10] <http://yum.baseurl.org/> 🇬🇧
- [11] <http://comments.gmane.org/gmane.comp.handhelds.meego.devel/2606> 🇬🇧
- [12] <http://www.clutter-project.org/> 🇬🇧
- [13] <http://gitorious.org/mx-toolkit> 🇬🇧

- [14] <http://help.meego.com/> 🇬🇧
- [15] <http://meego.com/community/blogs/dawnfoster/2010/meego-1.0-update-netbooks> 🇬🇧
- [16] <http://meego.com/downloads/releases/updates/meego-v1.0.1-core-os-update> 🇬🇧
- [17] <http://meego.com/downloads/releases/updates/meego-v1.0.1-netbook-update> 🇬🇧
- [18] <http://fluendo.com> 🇬🇧
- [19] <http://forum.meego.com/showthread.php?t=269> 🇬🇧
- [20] <http://banshee.fm/> 🇬🇧
- [21] <http://meeGo.gitorious.org> 🇬🇧
- [22] <http://anb-networkz.de/2010/06/meego-1-0-das-netbook-linux-von-intel-und-nokia/>
- [23] <http://lists.meego.com/pipermail/meego-dev/2010-May/002287.html> 🇬🇧
- [24] http://bugs.meego.com/show_bug.cgi?id=2682 🇬🇧
- [25] http://www.phoronix.com/scan.php?page=news_item&px=ODQxOA 🇬🇧
- [26] <http://forum.meego.com/> 🇬🇧

Autoreninformation

Thorsten Schmidt hat Spaß mit kleinen mobilen Helferlein und blickt neugierig auf aktuelle Entwicklungen in Sachen Linux-Distributionen für Netbooks.

Diesen Artikel kommentieren



Das neue openSUSE 11.3 ist da von Saskia Brückner

Nach acht Monaten ist es wieder soweit. Am 15. Juli hat das openSUSE-Projekt die neue Version seines Linux-Betriebssystems openSUSE 11.3 [1] veröffentlicht. Mit der neuen Version bietet das System einige Neuerungen und Verbesserungen, allerdings sind im Test auch negative Seiten aufgefallen.

openSUSE ist eine der ältesten Linux-Distributionen auf dem Markt. Gegründet wurde sie im Jahre 1992 als S.u.S.E GmbH. Seit Version SuSE 9.1 ist openSUSE ein von Novell [2] finanziertes und geleitetes Gemeinschaftsprojekt und bietet den Unterbau für die Enterprise-Versionen „SuSE Enterprise Linux Desktop/Server“ [3] [4] von Novell.

Oberfläche

Das neue openSUSE kommt seit der Version 11.2 wieder mit KDE SC 4.4.4 als Standard-Oberfläche, aber auch mit einer Vielzahl anderer Oberflächen, darunter GNOME 2.30.1, Xfce 4.6.1 und – ganz neu – LXDE [5] stehen zur Verfügung. KDE-Fans können ihr KDE effektiver und übersichtlicher auf Netbooks installieren. Das macht die neue KDE Plasma Netbook Edition möglich, die sich in den Systemeinstellungen aktivieren lässt. Für die GNOME-Anwender wurde das openSUSE-eigene Konfigurationstool YaST überarbeitet und stellt die Paketverwaltung nun übersichtlicher dar. Auch können sich GNOME-Anwender über den fehlenden HAL



Die neue KDE Plasma Netbook Edition. 

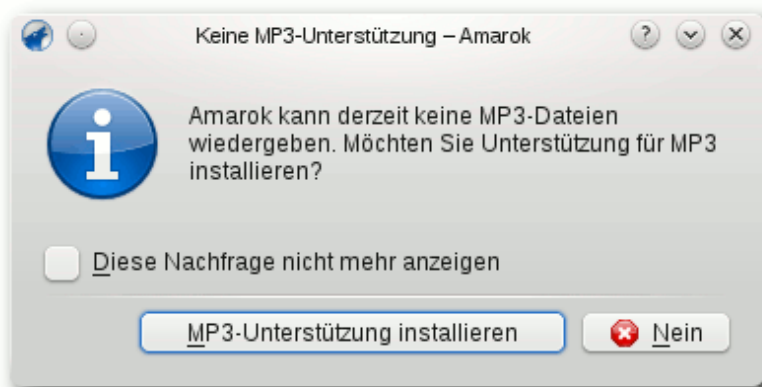
(Hardware Abstraction Layer [6]) freuen, der neuerdings nicht mehr im Basissystem vorhanden ist. KDE benötigt HAL allerdings noch für einige KDE-Anwendungen. Ganz neu mit dabei ist LXDE (Lightweight X11 Desktop Environment), eine schnelle Desktop-Oberfläche, die für die Installation auf leistungsschwächeren Systemen optimiert ist.

Installation

Die Installation selbst lief ohne größere Probleme ab. openSUSE installiert, wie seine Vorgänger auch, von Abbildern, was die Installation deutlich schneller macht. Die Software-Auswahl ist sehr allgemein gehalten. So kann man auswählen, ob man zum Beispiel **Spiele** installieren will. Zur genaueren Auswahl gelangt man über den Button



„Details“. Dort kann jedes Paket einzeln an- und abgewählt werden. Als Bootloader kommt standardmäßig Grub zum Einsatz. Auf Grub 2 wurde verzichtet, er kann aber alternativ installiert werden. Das einsatzbereite System stand auf dem Testgerät (Sony Vaio CS11S) nach ca. 20 Minuten zur Verfügung.



KSuseInstall installiert Codecs automatisch. 🔍

Multimedia

Der Versuch von openSUSE 11.3, „eingeschränkte Formate“, wie z. B. MP3s oder DVDs, abzuspielen, wird mit dem neuen KSuseInstall unternommen: Es soll die, aus lizenzrechtlichen Gründen fehlenden, Media-Codecs installieren. Leider ist dies in openSUSE nicht ganz so einfach wie in Ubuntu. Es öffnet sich zwar mit einem Medienplayer sofort die Nachfrage, ob die fehlenden Codecs installiert werden sollen, allerdings bricht die Installation mit einer Fehlermeldung ab, wenn man nicht vorher das Packman-Repository [7] eingebunden hat. Dieses ist in den Community-Repositories verfügbar und kann mit

„Software-Repositories“ [8] über YaST eingebunden werden. Ist das Repository aktiviert, funktioniert die Installation problemlos. openSUSE installiert dann nicht nur den benötigten Codec, sondern installiert gleich andere Codecs mit. Ubuntu hat dies einfacher gelöst: Hier installiert man einfach die „Restricted Formats“ aus dem Software-Center. Allerdings schaffen es beide Systeme nicht, CSS-geschützte DVDs abzuspielen, ohne vorher nochmals Hand anlegen zu müssen.

Grafiktreiber

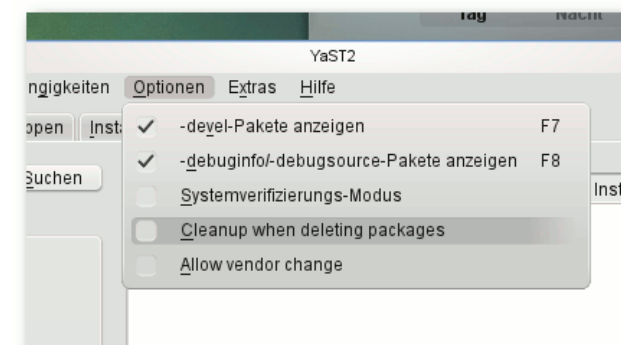
Probleme gibt es mit dem neuen Nouveau-Treiber, der den NVIDIA-Grafiktreiber ersetzen soll. Während es mit der Erkennung der richtigen Auflösung wenig Probleme gibt, macht der Treiber auf einigen Systemen Schwierigkeiten anderer Art. Einige Elemente auf dem Bildschirm werden verzerrt dargestellt und somit unkenntlich. Aber auch ganze Systemabstürze gab es auf dem Testsystem, deren Ursache wahrscheinlich auf den Nouveau-Treiber zurückgeht. Der proprietäre NVIDIA-Grafiktreiber [9] steht allerdings im Software-Repository zur Verfügung.



Verzerrte Darstellung mit dem Nouveau-Treiber. 🔍

Dateisysteme

Der Installer von openSUSE 11.3 bietet das neue Dateisystem btrfs [10] als Auswahl an. btrfs soll später die extended-Dateisysteme ext2, ext3 und ext4 ablösen (siehe „Das Dateisystem ext4“, freiesMagazin 04/2009 [11]), allerdings wird vom Installer davon abgeraten, btrfs für den Produktiveinsatz zu verwenden, da es noch experimentell ist. Verwendet man btrfs für die Root-Partition muss eine separate boot-Partition mit z. B. ext3 erstellt werden, von der der Kernel gestartet werden kann, da die Bootloader noch nicht von einer btrfs-formatierten Partition booten können.



Menüoption zum Aktivieren der neuen Funktion. 🔍

Paketverwaltung

Die Paketverwaltung zypper hat ebenfalls einige Verbesserungen bekommen. So ist es jetzt möglich, nicht gebrauchte Abhängigkeiten bei der Deinstallation eines Pakets zu ermitteln und ebenfalls zu löschen. Beispiel: Paket A installiert Paket B als Abhängigkeit mit. Wird Paket A jetzt gelöscht, prüft die Paketverwaltung, ob Paket B



noch von anderen Paketen gebraucht wird. Ist dies nicht der Fall, wird Paket B ebenfalls gelöscht. In YaST lässt sich die neue Funktion über das Menü aktivieren.

losen Dienst SpiderOak [13], bei dem es gleich zwei Gigabyte an Speicherplatz dazu gibt. Leider kam es hier zu Problemen: YaST konnte die Abhängigkeiten nicht auflösen, da ein Paket unter

anderem Namen in den Repositories vorhanden war: Es fehlte der Bindestrich im Namen.

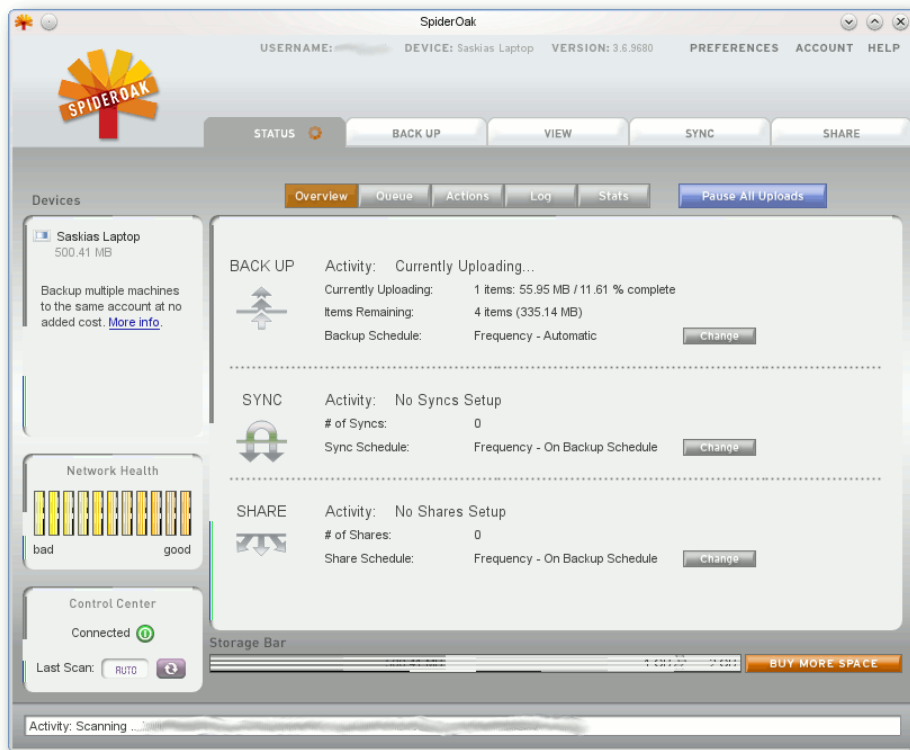
Fazit

Insgesamt macht openSUSE einen guten Eindruck. Einsteiger finden sich schnell im System zurecht und auch die Software-Auswahl scheint sehr zufriedenstellend. Die Hardware des Testgerätes wurde vollständig erkannt, selbst der Drucker funktionierte – wie es sein soll – ohne ein Zutun. Schade ist allerdings, dass das System vor allem durch die Verwendung des Nouveau-Treibers noch den einen oder anderen Fehler hat, was den Gesamteindruck schwächt. Abgesehen davon läuft das System sehr stabil und zufriedenstellend. Allerdings empfiehlt es sich, wenn ein entsprechender Internet-Anschluss vorhanden ist, die DVD kostenfrei aus dem Internet herunterzuladen, da die Verkaufsbbox, die es ab ca. 50 Euro zu kaufen gibt, außer

einem Handbuch und einem 90-Tage-Support keine weiteren Vorteile bietet.

LINKS

- [1] <http://de.opensuse.org/>
- [2] <http://www.novell.com>
- [3] <http://www.novell.com/products/desktop/>
- [4] <http://www.novell.com/products/server>
- [5] <http://lxde.org/>
- [6] http://de.wikipedia.org/wiki/HAL_%28Software%29
- [7] <http://de.opensuse.org/Packman#Packman>
- [8] http://de.opensuse.org/Paketquellen_als_YaST-Installationsquellen_einbinden
- [9] http://de.opensuse.org/Propriet%C3%A4re_NVIDIA-Grafiktreiber
- [10] <http://de.wikipedia.org/wiki/btrfs>
- [11] <http://www.freiesmagazin.de/freiesMagazin-2009-04>
- [12] http://de.wikipedia.org/wiki/Cloud_Computing
- [13] <http://www.spideroak.com>



Neu ist SpiderOak in openSUSE.

Cloud

Auch wenn openSUSE eine ganze Weile gebraucht hat, es hat die Cloud [12] entdeckt. Erstmals bietet openSUSE 11.3 einen Online-Dienst an, um Daten mit dem Internet zu synchronisieren. Gewählt haben die Entwickler den kosten-

freien Dienst SpiderOak [13], bei dem es gleich zwei Gigabyte an Speicherplatz dazu gibt. Leider kam es hier zu Problemen: YaST konnte die Abhängigkeiten nicht auflösen, da ein Paket unter anderem Namen in den Repositories vorhanden war: Es fehlte der Bindestrich im Namen.

Autoreninformation

Saskia Brückner benutzt seit fünf Jahren Linux. Angefangen bei SuSE Linux 9.3 ging der Weg über Mandriva zu Ubuntu und am Ende zurück zu openSUSE 11.3

Diesen Artikel kommentieren



Juni und Juli im Kernelrückblick von Mathias Menzer

Basis 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 2.6.35

Die Entwicklung des kommenden Kernels 2.6.35 schritt die letzten zwei Monate zügig voran. Anfang Juni brachte -rc2 [1] neben der Warnung an die anderen Kernel-Entwickler, nur noch Patches für Korrekturen einzureichen, auch noch Treiber für den staging-Bereich und Neuerungen, die bereits vor -rc1 eingesandt, jedoch nicht mehr berücksichtigt wurden.

Da Torvalds in Bezug auf die Ankündigung, große Änderungen nach Schließen des Merge Window nicht mehr zu berücksichtigen, stur blieb, stellte 2.6.35-rc3 [2] einen der kompaktesten -rc3 seit längerem dar und räumte mit einem Speicherproblem auf, das viele Nutzer des 2.6.35er Kernels traf und das sich in sehr vielfältiger Weise zeigte, je nachdem welchen Speicherbereich es gerade heimsuchte.

Die Zeitspanne bis zur Veröffentlichung des -rc4 [3] betrug fast vier Wochen, was jedoch auf Torvalds Urlaub und nicht auf weitere schwer aufzuspürende Probleme zurückzuführen war. Er

machte auch einen eher unspektakulären Eindruck, was wohl ebenfalls auf Torvalds neuer Beharrlichkeit bezüglich der Aufnahme größerer Neuerungen im Laufe der Entwicklung geschuldet ist. Dennoch wurde zum Beispiel die CAPI-Unterstützung [4] für Siemens Gigaset-Basisstationen als der CFQ-Scheduler (Completely Fair Queuing), der sich um die Verteilung der Zugriffe auf die Ein-Ausgabe-Queues durch die Prozesse kümmert, überarbeitet.

Eine seltene Erscheinung gab es beim -rc5 [5], da dieser tatsächlich einmal mehr Quelltext entfernt als hinzugefügt hat. Insgesamt mussten über 190000 Zeilen weichen, während nur etwas mehr als 3000 Zeilen hinzukamen, ein Faktor von etwa 60. All dies ist auf eine Aufräumaktion der Standardkonfiguration der ARM-Architektur zurückzuführen, bei der auskommentierte Optionen und „is not set“-Zeilen entfernt wurden und dabei das herunterzuladende Paket des gesamten rc5-Kernels schrumpfen ließen.

Den Abschluss des Juli bildete dann 2.6.35-rc6 [6] mit einem Patch von David Airlie, in dem Torvalds ein Heilmittel für die bereits länger anhaltenden Probleme mit dem verbreiteten GM945-Chipsatz sieht. Die sechste Vorabversion könnte – zumindest nach dem Willen Torvalds – auch die letzte vor Veröffentlichung des neuen Kernels sein. Doch meldete sich kurz nach seiner Ankündigung der Entwickler Stephen Rothwell und machte auf ein paar vergessene Patches in

linux-next aufmerksam [7], die unter anderem ein Problem mit dem Namensraum von Sysfs, einem virtuellen Dateisystem mittels dem Informationen vom Kernel- in den Userspace exportiert werden können, beheben. Greg Kroah-Hartmann meinte jedoch, dass diese Patches eine Veröffentlichung des 2.6.35 nicht aufhalten sollten.

LINKS

- [1] <http://lkml.org/lkml/2010/6/6/3>
- [2] <http://lkml.org/lkml/2010/6/11/420>
- [3] <http://lkml.org/lkml/2010/7/4/130>
- [4] http://de.wikipedia.org/wiki/Common_ISDN_Application_Programming_Interface
- [5] <http://lkml.org/lkml/2010/7/12/365>
- [6] <http://lkml.org/lkml/2010/7/22/433>
- [7] <http://lkml.org/lkml/2010/7/22/598>

Autoreninformation

Mathias Menzer wirft gerne einen Blick auf die Kernel-Entwicklung, um mehr über die Funktion von Linux zu erfahren und seine Mitmenschen mit seltsamen Begriffen und unverständlichen Abkürzungen verwirren zu können.

Diesen Artikel kommentieren



Tiling Windows mit Compiz und dem Grid-Plugin von Volker Duetsch

Große oder gar mehrere Bildschirme bieten deutlich mehr nutzbare Arbeitsfläche. Ein Nebeneffekt kann eine wahre Fensterflut sein. Wenn man den Durchblick behalten möchte, muss man Fenster manuell verkleinern, vergrößern, hin- und herschieben – oder man erleichtert sich die Arbeit mit einem Werkzeug. Das Stichwort ist Tiling.

Was ist Tiling?

Tiling steht für das Anordnen von mehreren Fenstern, ohne dass sich diese dabei gegenseitig verdecken bzw. überlagern. Die Idee ist nicht neu [1]. Bekannte Vertreter der Tiling Window Manager sind xmonad [2], Ratpoison [3], awesome [4], Bluetile [5] und dwm [6], um nur ein paar davon zu nennen. Diese setzen häufig einen manuellen Konfigurationsaufwand und eine gewisse Einarbeitungszeit voraus.

Compiz macht's möglich

Will man nun keinen der eben aufgeführten Fenstermanager benutzen, kann man auch auf Compiz zurückgreifen. Compiz [7] ist mehr als nur „grafischer Schnickschnack“, auch bekannt als Eye Candy [8].

Compiz bietet einige praktische Plugins, z. B. eine Bildschirmlupenfunktion (zoom-Plugin), die Darstellung aller geöffneten Fenster auf einer Ebene (plane-Plugin) oder das switcher-Plugin, welches bei der Fensterauswahl mittels **Alt** + **Tab** eine Live-Miniaturansicht der Fenster anzeigt. In diese Kategorie gehört zweifelsohne auch das in diesem Artikel vorgestellte Grid-Plugin, welches die Funktionalität eines Tiling Window Managers nachbildet.

Das Grid-Plugin aktivieren

Voraussetzung ist eine installierte und aktivierte Compiz-Umgebung. Diese ist bei vielen Distributionen bereits gegeben. Sollte es Probleme geben, ist Hilfe in den zahlreichen Foren [9] [10] zu finden, aber auch bei Youtube gibt es

etliche hilfreiche und distributionsspezifische Videos.

Das Plugin wird per CompizConfig Einstellungs-Manager aktiviert. Dieser ist bei Linux Mint 9 (Isadora) über „Menü → Kontrollzentrum → Darstellung“ zu erreichen. Sollte man den Eintrag dort nicht finden, z. B. bei anderen Linux-Distributionen, kann man das Programm auch per Terminal starten:

```
$ cccsm
```

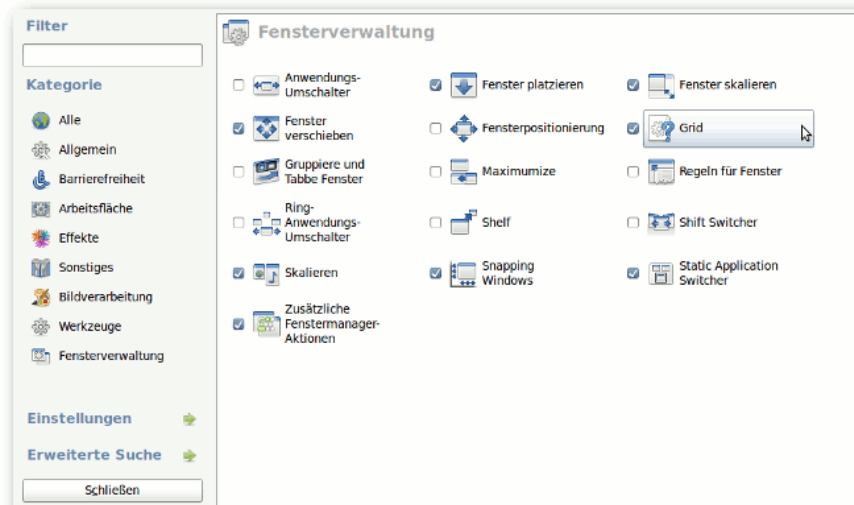
Weitere Einstellungen zum Grid-Plugin sind im Abschnitt Fensterverwaltung des CompizConfig Einstellungs-Managers möglich.

Sollte das Plugin dort nicht zu finden sein, lässt sich dieses leicht mit dem Paket **compiz-fusion-plugins-extra** über die Paketverwaltung nachträglich installieren.

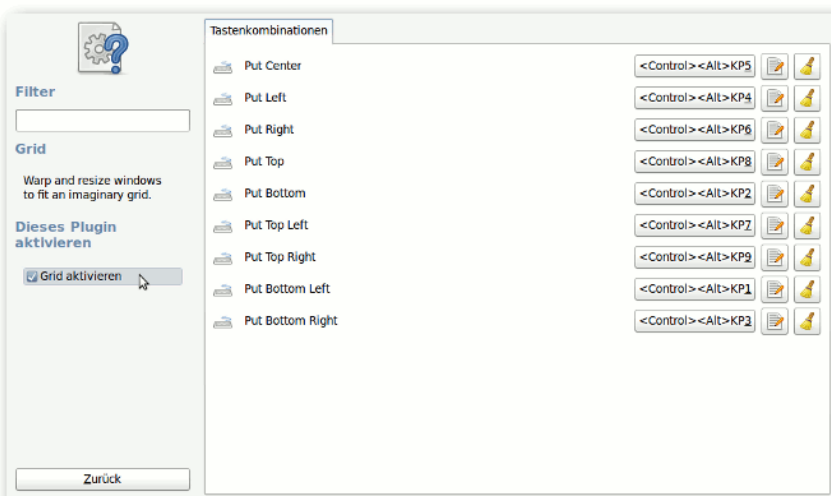
Nach dem Aktivieren des Plugins stehen die neuen Funktionen direkt zur Verfügung. Das Plugin kann natürlich nach eigenen Wünschen angepasst werden; die Voreinstellung ist aber empfehlenswert und sollte beibehalten werden.

Die Funktionen werden durch das gemeinsame Betätigen der Tasten **Strg** + **Alt** und einer Ziffer von **1** bis **9** auf dem Ziffernblock ausgeführt.

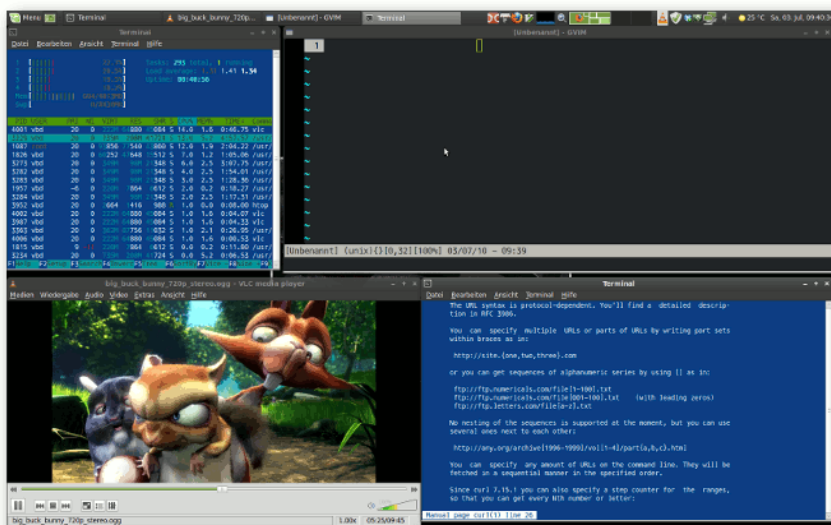
Die voreingestellten Tastenkombinationen des Grid-Plugins in tabellarischer Übersicht:



Hier findet man das Grid-Plugin. 🔍



Die voreingestellten Tastenkombinationen des Grid-Plugins. 🔍



mtop im Gnome-Terminal, GVim, VLC und nochmal GNOME-Terminal mit der manpage von curl (von links oben nach rechts unten). 🔍

Übersicht	
Tastenkombination	Fensteranordnung
Strg + Alt + 5	Vollbild
Strg + Alt + 4	Links
Strg + Alt + 6	Rechts
Strg + Alt + 8	Oben
Strg + Alt + 2	Unten
Strg + Alt + 7	Oben Links
Strg + Alt + 9	Oben Rechts
Strg + Alt + 1	Unten Links
Strg + Alt + 3	Unten Rechts

Durch wiederholtes Betätigen einer Ziffer wird die Fensterpositionierung weiter individuell verändert, d. h. bei mehrfachem Drücken der Tastenkombination **Strg** + **Alt** + **7**, wird die Fensterposition des aktiven Fenster im linken oberen Bildschirmbereich angepasst.

Fazit

Die Tiling-Funktionalität bietet eine gewisse Arbeitserleichterung und kann in Compiz einfach eingerichtet und getestet werden. Wenn die Neugierde geweckt wurde, sollte sich der interessierte Leser durchaus eingehender mit den im Artikel genannten Fenster-Managern

abseits von Compiz und des Mainstreams beschäftigen. Viele Tiling Window Manager sind bei ihren Hardwareansprüchen genügsamer und kommen auch mit älterer Hardware gut klar. Darüber hinaus bieten einige eine effiziente Desktopbedienung per Tastatur – z. B. in Anlehnung an Vim [11].

LINKS

- [1] [http://de.wikipedia.org/wiki/Tiling_\(Computer\)](http://de.wikipedia.org/wiki/Tiling_(Computer))
- [2] <http://www.xmonad.org/>
- [3] <http://www.nongnu.org/ratpoison/>
- [4] <http://awesome.naquadah.org/>
- [5] <http://www.bluetile.org/>
- [6] <http://dwm.suckless.org/>
- [7] <http://www.compiz.org/>
- [8] http://de.wikipedia.org/wiki/Eye_Candy
- [9] <http://www.ubuntuusers.de/>
- [10] <http://www.linuxmintusers.de/>
- [11] <http://www.vim.org/>

Autoreninformation

Volker Duetsch bevorzugt Software, die das KISS-Prinzip beherzigt und nicht zur Lösung von Problemen geschaffen wurde, die man ohne Computer gar nicht hätte.

Diesen Artikel kommentieren

L^AT_EX-Symbole einfügen mit LSS von Dominik Wagenführ

Im bereits erschienenen Artikel „Kurztipp: Unicode-Zeichen in L^AT_EX nutzen“ in freiesMagazin 06/2010 [1] wurde erklärt, wie man Unicode-Zeichen in einem L^AT_EX-Dokument einfügen kann. Eine Alternative für die Symbol-Suche ist der *L^AT_EX Symbols Selector*, kurz *LSS* [2].

Auch wenn *LSS* seit 2008 nicht mehr entwickelt wird, stellt es dennoch 478 L^AT_EX-Zeichen in verschiedenen Gruppen bereit und ermöglicht eine direkte Kopie der Zeichen in die Zwischenablage oder in den Editor (im Falle von gVIM).

Installation

Der *L^AT_EX Symbols Selector* steht auf der Webseite des Projekts zum Download im GPL2-lizenzierten Quelltext bereit. Für die Kompilierung werden neben einem C-Compiler die Entwicklerpakete von **libgtk2** und **libxml2** benötigt.

Danach kann man das Tar-Archiv in der aktuellen Version 0.1.6 entpacken und kompilieren:

```
$ tar -xzf lss-0.1.6.tar.gz
$ cd lss-0.1.6
$ ./configure
$ make
```

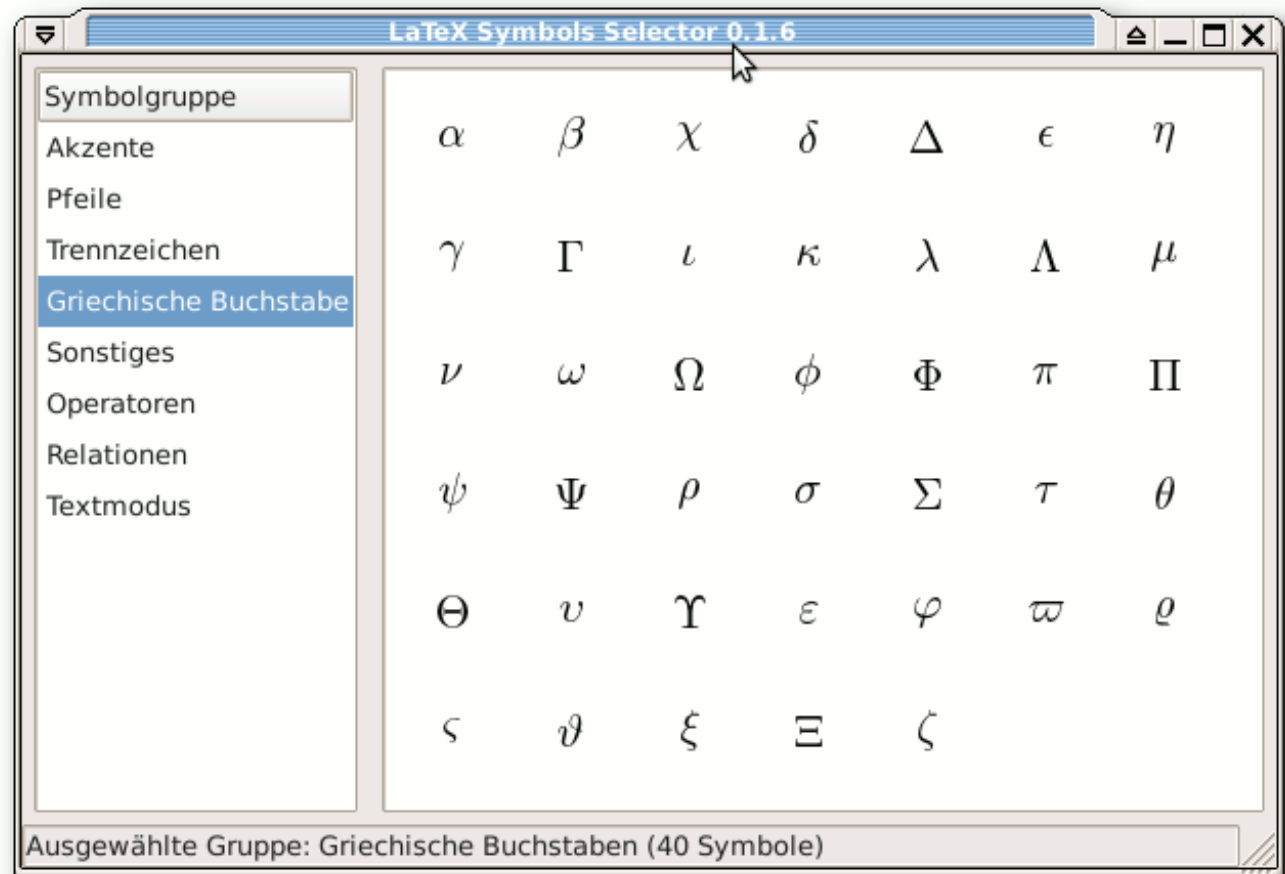
LSS steht dann bereits unter **src/lss** zur Ausführung bereit. Wer das Programm global, also auch für andere Benutzer, installieren möchte, kann dies mit einem der beiden folgenden Befehle tun:

```
# make install
```

bzw. mit Hilfe von

```
# checkinstall
```

über die Paketverwaltung.



Anzeige der griechischen Buchstaben. 🔍

LSS findet sich in einigen Distributionen wie ArchLinux, OpenBSD, openSUSE und Zenwalk auch in der Paketverwaltung.

Deutsche Sprache

LSS liegt derzeit (Juli 2010) nur in englischer Sprache und polnischer Übersetzung vor. Es gibt

aber eine deutsche Sprachdatei, die man nutzen kann. Man lädt sich die Datei `de.po` einfach herunter und speichert sie im Ordner `po` ab. Daneben muss man in diesem Ordner noch die Datei `Makefile.in` bearbeiten, indem man die Zeile mit den Übersetzungsdateien wie folgt anpasst:

```
POFILES = pl.po de.po
```

Danach muss man erneut `make` aufrufen und das Programm global installieren. Führt man `LSS` nur lokal aus, wird die deutsche Übersetzung sonst nicht gefunden.

Benutzung

Nachdem man das Programm durch den Aufruf von `lss` gestartet hat, sieht man bereits die „Akzente“, die in \LaTeX zur Verfügung stehen. Als Standard kann man nun doppelt auf einen Eintrag klicken, um diesen in die Zwischenablage zu kopieren.

Auf der linken Seite sieht man verschiedene Gruppen von Symbolen, die man auswählen kann. Auf diese Art findet man relativ schnell das gesuchte Zeichen – wenn es denn im $\text{\LaTeX Symbols Selector}$ geführt wird.

Mit der rechten Maustaste kann man das Optionsmenü aufrufen. Hier kann man einstellen, ob ein Zeichen an die Zwischenablage und/oder den Editor `gVIM` gesendet werden soll.

Daneben lässt sich im Optionsmenü noch einstellen, ob bereits ein Einzelklick das Symbol kopiert, ob auch die AMS-Symbole angezeigt wer-

den sollen (AMS steht für American Mathematical Society [3]) und ob die Symbolnamen, d. h. der \LaTeX -Code, direkt unter den Zeichen stehen soll. Alle Einstellungen befinden sich in der Datei `td/.lss.xml`.

Fazit

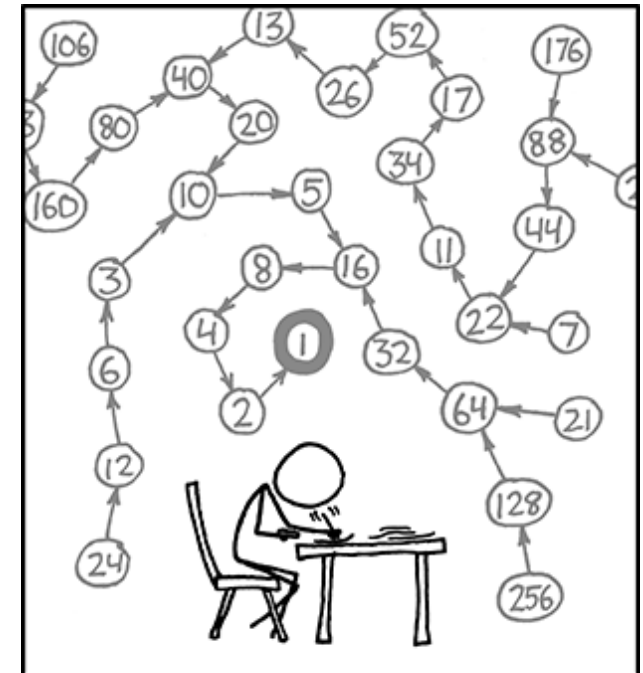
Der $\text{\LaTeX Symbols Selector}$ ist ein schönes kleines Programm, welches sich gut eignet, bekannte \LaTeX -Symbole wiederzufinden. Da die Auswahl aber beschränkt ist, muss man sicherlich ab und zu auf andere Quellen wie die \LaTeX -Symbol-Tabelle [4] oder Detexify² [5] zurückgreifen.

LINKS

- [1] <http://www.freiesmagazin.de/freiesMagazin-2010-06>
- [2] <http://clayo.org/lss/> 
- [3] <http://www.ams.org/> 
- [4] <http://www.ctan.org/tex-archive/info/symbols/comprehensive/symbols-a4.pdf> 
- [5] <http://detexify.kirelabs.org/classify.html> 

Autoreninformation

Dominik Wagenführ setzt viele Texte mit \LaTeX und verzweifelt immer wieder daran, dass er sich den \LaTeX -Code häufig genutzter Zeichen nicht merken kann.



THE COLLATZ CONJECTURE STATES THAT IF YOU PICK A NUMBER, AND IF IT'S EVEN DIVIDE IT BY TWO AND IF IT'S ODD MULTIPLY IT BY THREE AND ADD ONE, AND YOU REPEAT THIS PROCEDURE LONG ENOUGH, EVENTUALLY YOUR FRIENDS WILL STOP CALLING TO SEE IF YOU WANT TO HANG OUT.

„Collatz Conjecture“ © by Randall Munroe
(CC-BY-NC-2.5), <http://xkcd.com/710>

Diesen Artikel kommentieren 

Videoschnitt mit Avidemux von Kirsten Roschanski

Im digitalen Video-Zeitalter ist das nachträgliche Schneiden, Zusammenfügen oder Umwandeln von Videos auch im Heimbereich an der Tagesordnung. Sei es, um den gerade aufgenommenen Urlaubsfilm zu schneiden, die Werbung aus einer Reportage zu entfernen oder ganz schlicht, um das Lieblingsmusikvideo auf dem iPod abspielen zu können. Hier kommt *Avidemux* [1] ins Spiel, ein leistungsfähiges, unter der GPL (GNU General Public License) stehendes Videoschnittprogramm, das kostenfrei erhältlich ist.

Über Avidemux

Am 17. Mai 2010 ist die Version 2.5.3. von *Avidemux* erschienen. Sie kann alle gängigen Containerformate wie AVI, MPEG, PS/TS, VOB-Dateien (DVD-Video), MP4, OGM, FLV-Dateien, MKV, ASF/WMV, Sequenzen von Einzelbildern (BMP, JPEG oder PNG) und Nuppelvideo bearbeiten sowie in den wichtigsten Formaten auch wieder speichern. So werden unter anderem die Videoformate MPEG-1, MPEG-2, MPEG-4 SP/ASP, H.264/AVC, M-JPEG und DV, HuffYUV, FFv1, H.263, WMV und Flash-Video und die Audioformate MP3, AAC, Ogg Vorbis, MP2, AC-3, PCM und LPCM unterstützt.

Bemerkenswert sind neben der Vielzahl der unterstützten Formate vor allem die flexiblen Anpassungsmöglichkeiten der Video- und Audioparameter. Wenn man weniger geübt ist, bedient

man sich einfach der mitgelieferten Konvertierungen zu Video-CD, Super-Video-CD, iPod, PlayStation und DVD oder man definiert sich eigene Konvertierungsparameter. Dabei ist das automatische Aufteilen von Videos in die gewünschten Blockgrößen nicht minder praktisch als die Integration von Untertiteln in Containerformaten als ASCII- oder VOB-Sub.

Mit Hilfe von zahlreichen Filtern lassen sich gezielt Sättigung, Helligkeit und Kontrast beeinflussen. Aber auch das Drehen oder Beschneiden der Videos ist möglich. Durch das Einfügen von Effekten, wie „Wirbel“ oder „Mosaik“, verpasst man dem Video eine individuelle Note. Wenn das Meeresrauschen zu laut ist, wie wäre es da mit einer Optimierung der Lautstärke oder dem Einfügen einer zweiten Tonspur für die Klänge des eigenen komponierten Klavierstücks? All das und noch einiges mehr kann man mit *Avidemux* machen, der Kreativität sind also fast keine Grenzen gesetzt. Möglicherweise muss man, um alle Formate für Video und Audio nutzen zu können, einige Codecs nachinstallieren.

Um die vielfältigen Optionen und zahlreichen Möglichkeiten des Programms nutzen zu können, ist eine gewisse Einarbeitungszeit erforderlich. Wenn man sich gut eingearbeitet hat, offenbaren sich die zahlreichen Möglichkeiten des Programms. Durch die zwei verschiedenen grafischen Benutzeroberflächen, welche durch die Bibliotheken GTK+ und Qt realisiert werden

(welche für alle gängigen Betriebssysteme erhältlich sind), ist *Avidemux* plattformunabhängig. Für Fortgeschrittene bietet *Avidemux* zusätzlich eine Skriptsprache auf Kommandozeilenebene (ECMAScript-Interpreter SpiderMonkey [2]) an, mit der sich unter anderem viele sich wiederholende Arbeitsschritte automatisieren lassen.

Installation

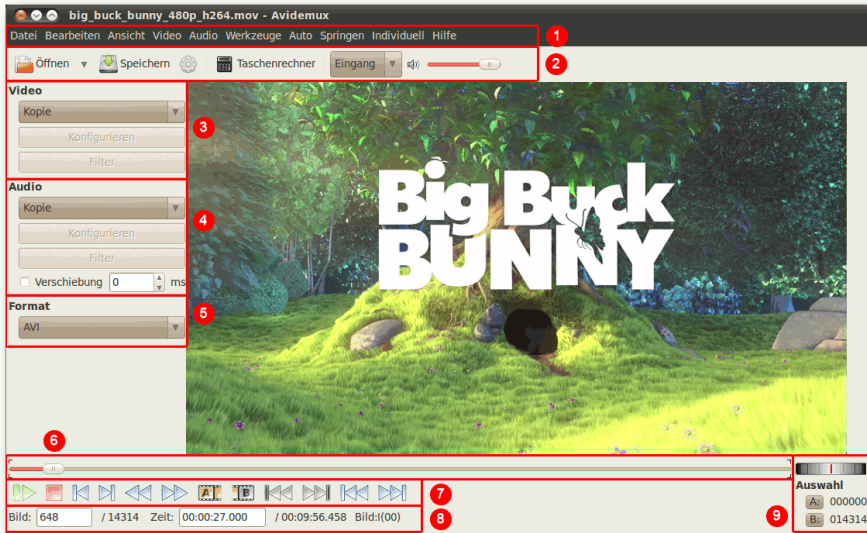
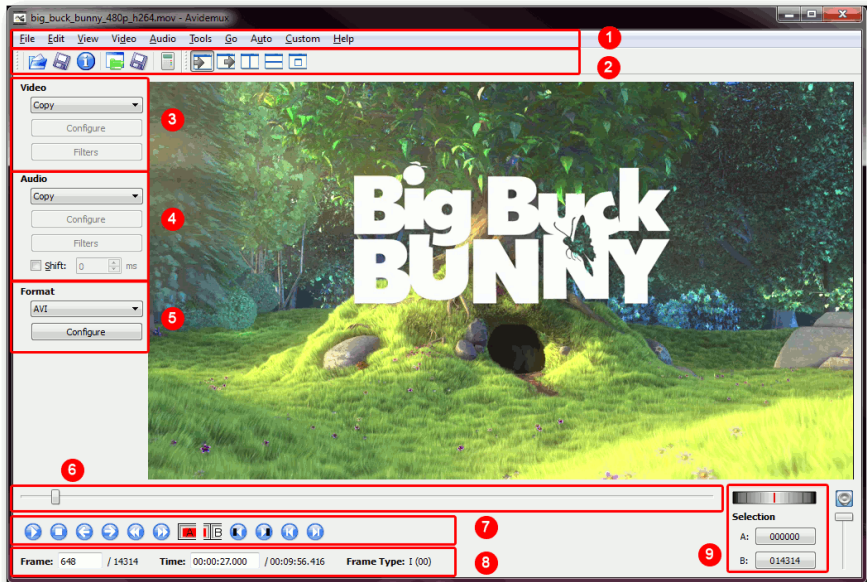
In den meisten Linux-Distributionen ist *Avidemux* in den Paketquellen enthalten. Daher kann man es über das Software-Center oder die Paketverwaltung installieren. Ansonsten kann man sich auch auf der Projektseite die entsprechende Installationsdatei bzw. das Installationspaket herunterladen und manuell installieren.

Hinweis: Für diesen Artikel wurde die Oberfläche der GTK+-Bibliothek installiert und die Übung zeigt anhand dieser grafischen Benutzeroberfläche die Funktionsweise.

Die Oberfläche

Vorab als Information: Die GTK+- und Qt-Oberfläche unterscheiden sich nicht in ihrer Funktionsweise, sondern nur in ihrem Aussehen. Daher braucht man nicht extra die GTK+-Bibliothek zu installieren, wenn gerade KDE (mit Qt-Bibliothek) verwendet wird.

Die Oberfläche lässt sich grob in vier Blöcke einteilen. So wie bei einer Internetwebseite gibt es eine Kopf- und eine Fußleiste, dazwischen ist

Avidemux mit GTK+-Bibliothek. Avidemux mit Qt-Bibliothek. 

der Inhalt gefasst, der sich in die linke Spalte und den Bildbereich teilt.

Im Kopf befinden sich ein Pull-down-Menü (1) und Symbole für den Direktzugriff (Symbolleiste) (2). Die linke Spalte bietet dem Nutzer die Videokonvertiereinstellungen (Codec und Optionen) (3), die Audiokonvertiereinstellungen (Codec und Optionen) (4) und das Dateicontainerformat (5). Im Bildbereich sieht man das aktuell angezeigte Bild.

Um durch das Bildmaterial in *Avidemux* zu navigieren, muss man in der Fußleiste die Symbole zur Navigation sowie Schnittoptionen (Marker) (7) benutzen. Auch weitere Angaben zur aktuellen Position (aktuelles Bild, aktuelle Zeit) (8), die Zeitleiste (6), den Schieberegler und die Markenposition (9) findet man in der Fußleiste.

Übungen zum Warmwerden

Obwohl die Verwendung von *Avidemux* relativ einfach und intuitiv ist, wird in den folgenden kleinen Übungen gezeigt,

wie man mit dem Programm in der grafischen Oberfläche arbeitet.

Dass man *Avidemux* in seinem Programmmenü gefunden und auch schon gestartet hat, ist jetzt einfach mal vorausgesetzt.

Datei öffnen

Nach dem Programmstart will man sich in der Regel ja nicht das Programm anschauen, sondern eine Videodatei damit bearbeiten, und dazu braucht man eine passende Datei. Im Artikel hat für die Schaubilder der Trailer von „Big Buck Bunny“ [3] hergehalten. Der Film und dessen Bilder stehen unter der CreativeCommons-Attribution-3.0-Lizenz [4].

Um eine zu bearbeitende Videodatei in den Editor zu laden, klickt man auf das Symbol „Öffnen“ in der Symbolleiste. Anschließend wird das gewünschte Video ausgewählt und per Klick auf die Schaltfläche „Öffnen“ geöffnet. Über die „Auswahlleiste“ unterhalb des Ordnerfensters besteht die Möglichkeit, die Anzeige von „alle“ auf „AVI-“ oder „MPEG-Videos“ beziehungsweise „Bitmap-Grafiken“ einzuschränken.

Der „Big Buck Bunny“-Trailer wurde im QuickTime-Videoformat (MOV-Datei) heruntergeladen, daher wird die Datei `trailer_480p.mov` geöffnet. Da es sich dabei um das H.264-Format handelt, sollte man die Frage nach Benutzung des „sicheren Modus“ auf alle Fälle bestätigen.

Das wäre geschafft! Nun ist die Datei geladen und man kann sie sich über den „Abspielen“-Knopf in der Navigationsleiste anschauen.

Schneiden

Das Schneiden ist wahrscheinlich das Werkzeug, das in einem Videoschnittprogramm am häufigsten genutzt wird. Darum ist es auch in der Bedienung sehr einfach.

Um das geöffnete Video zu schneiden, geht man mit dem „Schieberegler“ zum ungefähren Anfangspunkt für den Schnitt. Anschließend kann über die „Einfachfeile“ Bild für Bild vor und zurück geblättert werden, bis man das gewünschte Startbild ausgewählt hat. Mit einem Mausklick auf das Symbol „A“ (**AltGr** + **8**), ist der Anfangsmarker gesetzt. Genauso, wie schon das Startbild gefunden und gesetzt wurde, verfährt man mit dem gewünschten Endbild, klickt dabei aber auf das Symbol „B“ (**AltGr** + **9**), um den Endmarker zusetzen. Wenn man nun den zwischen den mit Markern markierten Bereich entfernen will, drückt man einfach die **Entf**-Taste oder geht über „Bearbeiten → Löschen“. Dabei ist zu beachten, dass die gesamte Auswahl inklusive Startbild „A“, aber ohne Endbild „B“ herausgeschnitten wird.

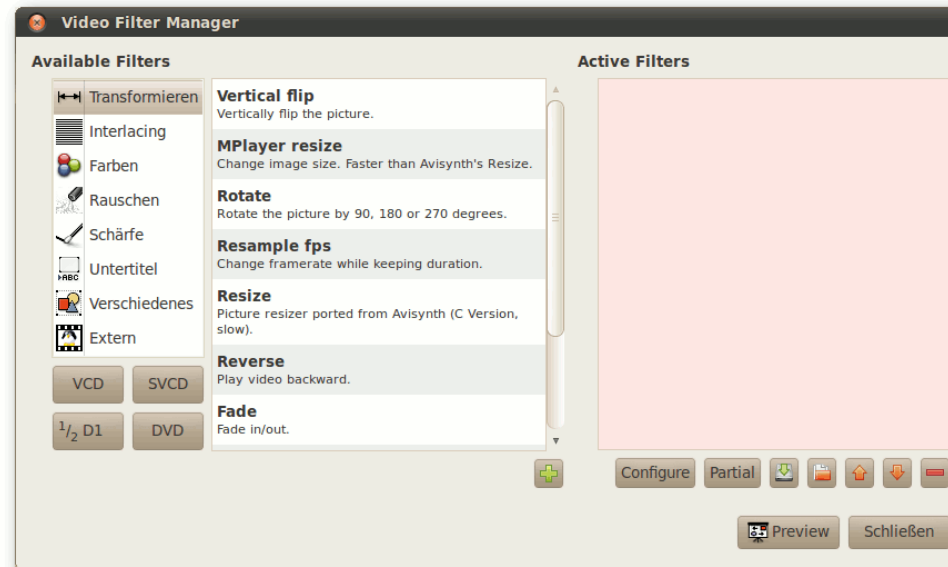
Die Auswahl lässt sich aktuell nicht invertieren, daher wird weggeschnitten, was markiert ist.

Filtern

Es gibt in *Avidemux* zwei Filtertypen, einen Filter für die Videoeigenschaften und einen, der auf die Tonspur Einfluss hat. Mit Filtern hat man die Möglichkeit, sein Bild-/Tonmaterial aufzupeppen bzw. aufzubereiten. Man muss aber beachten, dass Filter nur genutzt werden können, wenn die jeweilige Konvertierungseinstellung nicht auf „Kopie“ steht.

Videofilter

Bei den Videofiltern gibt es schon eine Reihe von vordefinierten und installierten Filtern: zum Beschneiden, Ränder schwärzen, Vergrößern mit Schwarzrand, zum Größe ändern, Drehen und Spiegeln, Rückwärtslaufen lassen sowie zum Ein- und Ausblenden.



Videofilter-Dialog von Avidemux. 

Man erreicht die Videofilter über die Schaltflächen „Video → Filter“ (**Strg** + **Alt** + **F**). Unter der Option „Configure“ im Videofilter besteht die Möglichkeit, Änderungen vorzunehmen. Es empfiehlt sich, wenn man erst einmal etwas sicherer im Umgang mit dem Programm ist, dort ein wenig zu experimentieren. Es hat sich aber gezeigt, dass die vordefinierten Werte in der Regel schon ganz brauchbare Ergebnisse liefern.

Hinweis: Man kann beliebig viele Filter hintereinander anwenden. So ist es z. B. sinnvoll, nach dem „Deinterlacing“ noch mal zu „Sharpen“.

Audiofilter

Die Audiofilter sind im Gegensatz zu den Videofiltern nicht ganz so umfangreich, haben aber dennoch ihre Berechtigung in der Videobearbeitung. Mit ihnen kann man unter anderem die Lautstärke regulieren oder aber auch die Ton- und die Videospur wieder synchronisieren.

Untertitelspuren

Untertitel sind nicht nur was für professionelle Filmproduktionen. So kann man z. B. Untertitelspuren auch für das Urlaubsvideo verwenden, indem man einfach den Tag oder den

Namen des zu sehenden Ortes kurz einblendet. Meistens werden Untertitelspuren jedoch im privaten Bereich für einen Film/Serie verwendet, die nicht in der gewünschten Sprache vorliegt.

Untertitelspur einfügen und bearbeiten

Die Untertitel gehören bei *Avidemux* zu den Videofiltern. Daher findet man sie unter „Video → Filter“. Wenn man nun einen Untertitel einfügen

möchte, muss man wissen, in welchem Format man diesen vorliegen hat.

Bezieht man die Untertitel aus dem Internet, liegen diese überwiegend als SRT-Datei vor. In solch einem Fall geht man wie folgt vor: Im ersten Schritt öffnet man die Videofilter und wählt „*Untertitel*“, dann als zweites „*Subtitled*“. Anschließend kann man die SRT-Datei auswählen und noch einige Einstellungen vornehmen, wenn man möchte. Oft ist es aber nicht mehr nötig, da die Untertiteldatei in der Regel alle relevanten Daten mitliefert.

Konvertieren einer DVD-Untertitelspur in das Textformat SRT

Wenn man eine Untertitelspur aus einer DVD bezieht, dann liegt diese für gewöhnlich im Vob-Sub-Format vor. Dabei handelt es sich eigentlich um zwei Dateien, eine SUB- und eine IDX-Datei, die alle Informationen über den Untertitel als Bilder gespeichert haben. Diese könnte man selbstverständlich auch in *Avidemux* unter sein Bildmaterial legen, doch wenn man diese zuerst wieder in eine Textdatei schreibt, ergeben sich folgende Vorteile: Die Untertitelspur ist erstens schöner, da eine bessere Kantenglättung erreicht wird und zudem kann man sie sich in einer beliebigen Schrift anzeigen lassen. Zweitens ist eine solche Textdatei vorwiegend nur bis zu 100 KB groß und drittens ist eine Nachbearbeitung möglich (z. B. wenn die Übersetzung nicht ganz so gut ist). Wie man eine solche DVD-Untertitelspur aus einer DVD extrahiert, wird im Wiki von ubuntusers.de [5] ganz gut beschrieben.

Da der Text in Bildern kodiert ist, braucht man eine Texterkennungssoftware (OCR), um ihn zu dekodieren. Dies ist mit *Avidemux* machbar. Als erstes muss man hierfür eine dynamische Zuordnungsdatei erstellen. In dieser werden die erkannten Zeichen zwischengespeichert und dem jeweiligen ASCII-Code zugeordnet. Es empfiehlt sich, diese Datei zu speichern, denn so kann man sie anschließend wiederverwenden. Wenn man nicht ganz von vorne beginnen möchte, kann man im Internet schon erstellte Dateien herunterladen. Im Gegenzug ist es fair, seine eigene erweiterte Zuordnungsdatei anderen Nutzern zur Verfügung zu stellen.

Um nun eine DVD-Untertitelspur mit *Avidemux* zu konvertieren, geht man über das Menü „*Werkzeuge* → *OCR* → *VobSub* → *SRT*“ in den gewünschten Dialog. In diesem wählt man die zugehörige IDX-Datei aus, gibt an, wie die Ausgabedatei heißen soll bzw. wie sie heißt (wenn schon eine angelegt wurde) und – falls vorhanden – kann man nun eine Zuordnungsdatei auswählen. Wenn im Vorfeld keine Zuordnungsdatei angelegt ist oder aus dem Internet heruntergeladen wurde, fragt die OCR-Software nach der korrekten Zuordnung von gescannten Zeichen zum ASCII-Code. Es kann vorkommen, dass Buchstabenkombinationen, die sehr eng zusammenstehen, wie z. B. „rn“ oder „tff“, oft nicht richtig erkannt werden und nachbearbeitet werden müssen. Je mehr man dem Programm solche Kombinationen beibringt, desto schneller und genauer wird es.

Sobald die SRT-Datei fertig ist, erscheint die Abfrage, ob man seine Zuordnungsdatei für eine spätere Verwendung speichern möchte. Wie schon gesagt, empfiehlt es sich, dieses anzunehmen und sich ggf. mit anderen Nutzern auszutauschen. Die so erstellte SRT-Datei lässt sich ohne Probleme in einem Schritt in ein progressives Format umwandeln.

Interlacing wesentlich verbessern

Auch wenn sich inzwischen in Deutschland das HD-Fernsehen mehr und mehr durchsetzt, so sind noch nicht alle Programme umgestellt. Das in Deutschland noch weit verbreitete PAL-Fernsehformat verwendet sogenannte Halbbilder. Hierbei liegen bei aufeinander folgenden Bildern nur die geradzahligen oder ungeradzahligen Bildzeilen vor. Dies stellt solange kein Problem dar, wie die Anzeigegeräte mit der originalen Bildwechselfrequenz von 25 Hertz diese genau nachvollziehen können. Billigere LCDs bzw. Computermonitore können keine oder nur sehr schlechte Zeilensprungverfahren (engl. Interlacing [6]) darstellen. Daher sind die Interlacing-Filter von besonderer Bedeutung in *Avidemux*. Mit ihnen kann man die Darstellung wesentlich verbessern.

Für die Anwendung geht man in den Videofilter und wählt „*Interlacing*“. Mit dem Menüpunkt „*Deinterlacing*“ kann die Bildqualität der Halbbilder verbessert werden. Als erstes wird man nach dem „Motion Threshold“-Wert gefragt, dann nach dem „Blend Threshold“. Beide haben einen voreingestellten Wert. Es empfiehlt sich auch hier, diesen erst einmal so stehen zu lassen.

Das „Deinterlacing“ bietet gute Verbesserungen und schont dabei die CPU. Wer aber mehr Wert auf Qualität legt, sollte mit „DG Bob“ [7] arbeiten.

Zweite Tonspur bei Video-Dateien hinzufügen

Eine zweite Tonspur einzufügen ist nicht schwer. Im Pull-down-Menü „Audio“ findet man „Zweite Audiospur“. Wählt man diesen Punkt, erscheint ein Dialog, in dem die hinzuzufügende Audio-datei ausgewählt wird. Das Verwenden einer MP3-Datei mit variabler Bitbreite ist nicht möglich, weswegen es sich empfiehlt, in einem solchen Fall die Audiospur im Vorhinein in AC3 (Adaptive Transform Coder 3) [8] zu konvertieren (z. B. mit Audacity [9]). Wenn man die Audiospur dabei gleich noch ausrichtet, braucht man sie nur einzufügen und muss in *Avidemux* keine Synchronisation mehr durchführen.

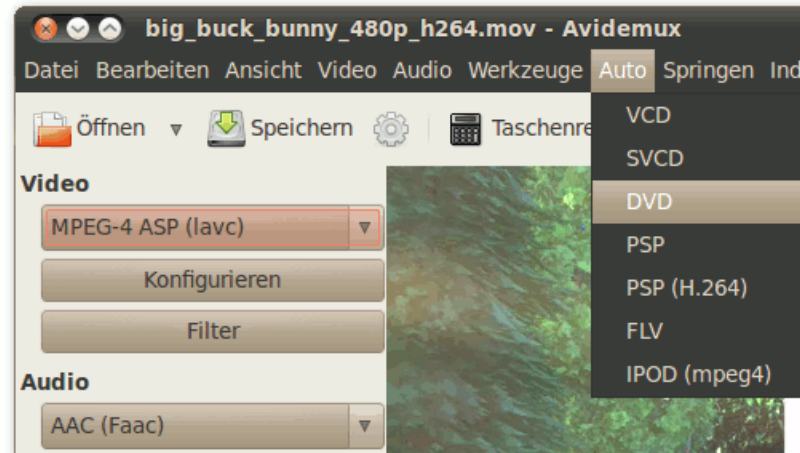
Synchronisation von Bild und Ton

Wenn es nun doch einmal vorkommt, dass Bild und Ton asynchron laufen, kann man das mit dem Audiofilter wieder beheben. Dazu schaut man sich die Spur unter „Audio → Haupt-/Zweite Audiospur“ an. Ist hier schon eine Verschiebung eingetragen, so sollte versucht werden, diese über „Audio → Filter“ zu beseitigen. Dazu trägt man den Wert unter „Verschiebung“ ein. Wenn der Wert auf „0 ms“ steht, heißt es mit der „Verschiebung“ zu experimentieren, bis Audio- und Videospur synchron laufen.

Formatsache

Um das Projekt, welches mühsam geschnitten, gefiltert usw. wurde, beispielsweise auf dem DVD-Player abzuspielen, muss es gespeichert werden. Nun hat man die Qual der Wahl. Es

besteht die Möglichkeit der Formatumwandlung, sodass das Video gleich in dem gewünschten Format vorliegt. Am Einfachsten geht dieses über das Pull-down-Menü „Auto“, denn hier sind schon einige Formate vordefiniert und man kann sich



Automatischer Export des Projektes zum DVD-Format. 🔍

seine gewünschte Variante auswählen. Unter anderem stehen VCD (Video-CD), SVCD (Super-Video-CD), DVD oder PSP (Playstation Portable) als Zielformat zur Verfügung. Das Schöne dabei ist, dass *Avidemux* alle Parameter automatisch einstellt. Wird im Menü „Auto“ zum Beispiel „DVD“ gewählt, so braucht man anschließend nur noch das Seitenverhältnis von Quell- und Zielvideo anzugeben und mit „OK“ bestätigen. Das Ergebnis kann dann auf eine DVD gebrannt und anschließend über den DVD-Player auf dem Fernseher angeschaut werden.

Fazit

Der Artikel regt hoffentlich an, mal selber das eine oder andere kleine Video zu bearbeiten. Mit *Avidemux* gibt es ein sehr leistungsfähiges Videoschnittprogramm, das sich nicht zu verstecken braucht und kostenfrei erhältlich ist.

LINKS

- [1] <http://www.avidemux.org/> 
- [2] <https://developer.mozilla.org/en/spidermonkey> 
- [3] <http://www.bigbuckbunny.org/> 
- [4] <http://creativecommons.org/licenses/by/3.0/> 
- [5] http://wiki.ubuntuusers.de/DVDs_manuell_rippen#optional-Untertitel-rippen
- [6] <http://de.wikipedia.org/wiki/Zeilensprungverfahren>
- [7] <http://avisynth.org.ru/docs/english/externalfilters/dgbob.htm> 
- [8] http://de.wikipedia.org/wiki/Dolby_Digital
- [9] <http://audacity.sourceforge.net/> 

Autoreninformation

Kirsten Roschanski nutzt Avidemux unter Ubuntu im privaten Bereich zum Nachbearbeiten von Urlaubsvideos und Fernsehaufzeichnungen.

Diesen Artikel kommentieren 

Rezension: Die Anarchie der Hacker von Dominik Wagenführ

Viele Menschen wissen nicht, dass „Freie Software“ und „Open Source“ zwei unterschiedliche Dinge beschreiben, die nur ansatzweise auf das Gleiche abzielen. Der Begriff „Freie Software“ wurde von Richard M. Stallman geprägt, der die Ansicht vertritt, dass Software allen Menschen gehören sollte. Das Buch „Die Anarchie der Hacker“ von Christian Imhorst behandelt die Entstehung der Freie-Software-Bewegung und Richard Stallmans Einflüsse darauf.

Richard Stallman als Anarchist

Der Begriff „Anarchist“ bzw. „Anarchie“ im Allgemeinen hat heutzutage einen sehr schlechten Beigeschmack, bringt man ihn doch meist mit Chaos und Zerstörung in Verbindung. Dabei bedeutet Anarchie [1] doch nur die Abwesenheit einer Hierarchie als Form der Unterdrückung, die die individuelle und kollektive Freiheit einschränkt. Eine Freiheit, die Richard Stallman [2] seit den 1970er Jahren propagiert.

In der wissenschaftlichen Arbeit „Die Anarchie der Hacker“ zeigt der Autor Christian Imhorst die Beweggründe Stallmans auf, die eigentlich anfangs ganz simpel waren: Zum einen sah er sich in seiner Arbeit in Harvard unterdrückt, weil die Rechenzeitvergabe der damals noch teuren Computer sehr ungerecht vorging. Daneben war eines der größten Ärgernisse ein simpler Xerox-Drucker, dessen Treiber Stallman nicht anpassen

durfte, weil der Entwickler des Treibers ein Geheimhaltungsabkommen (Nondisclosure Agreement) mit Xerox unterschrieben hatte (S. 51 ff). Dies veranlasste Stallman dazu, zu einem Verfechter Freier Software zu werden. Software, die allen gehört, und von keinem – bis auf einige Grundsätze – eingeschränkt werden darf.

Die GPL als Meilenstein

Ein Meilenstein im Kampf gegen proprietäre Software war die GNU General Public License (kurz GPL [3]). Auf Basis der Hackerethik [4], die unter anderem Informationsfreiheit und Dezentralisation propagiert (S. 23 ff), wurde zuerst die Emacs-Lizenz (zum zugehörigen von Stallman entwickelten Editor Emacs [5]) verfasst und danach im Jahr 1989 etwas allgemeiner die GPL (S. 57).

Die Lizenz stellte sicher, dass jeder ein GPL-lizenziertes Programm ausführen, den Quellcode des Programms studieren und anpassen, das Programm verbreiten und Veränderungen veröffentlichen darf (S. 56).

Bill Gates, Gründer von Microsoft, war kein Freund der Hackerethik und Freier Software, denn sein Geschäftsmodell basierte gerade auf der Nutzung freier Informationen ohne diese weitergeben zu müssen. So wurde beispielsweise das erfolgreiche MS-DOS aus dem aufgekauften QDOS von Tim Patterson entwickelt, welches wiederum den Code des frei zugänglichen CP/M „geklaut“ hatte (S. 44 f).

Mit der GPL war damit aber Schluss. Natürlich können Firmen GPL-lizenzierten Code immer noch verkaufen, aber sie müssen dem Käufer den Quellcode unter den gleichen Bedingungen zur Verfügung stellen, unter dem sie ihn selbst erhalten haben. Dies führte dazu, dass heutzutage weniger mit GPL-Software Geld verdient wird, sondern eher mit Zusatzmaterial und Dienstleistungen rund um die angebotene Software (S. 69). Etwas, wovon Microsoft sehr wenig hat, da ein Großteil des Geldes durch Windows-Lizenzen erzielt wird.

Frei oder frei?

Nicht jeder sah die GPL aber als Segen an. Vor allem Anhänger der Berkeley Software Distribution License (BSD-Lizenz [6]) waren über die virale Verbreitung der GPL nicht erfreut, da sie die Freiheit eines Programmierers einschränkte. Mit der BSD-Lizenz war wirklich alles erlaubt, solange man den Urheber eines Stück Codes nannte. Auch die Verarbeitung zu proprietärer Software war damit möglich. Etwas, was die GPL gerade verhindern soll. In der Freie-Software-Bewegung fanden sich somit zwei Lager, die doch beide das Gleiche wollten: Einen freien Informationsaustausch.

Dabei hat Freie Software vor allem im Englischen ein Problem: Das Wörtchen „free“ steht für „frei“, aber eben auch für „kostenlos“ (S. 50). Daher stammt auch der Ausspruch „Frei wie in Freiheit, nicht wie in Freibier“, wenn es um Freie Software

geht. Aus dem Grund setzten sich 1998 einige wichtige Mitglieder der Szene zusammen, darunter Bruce Perens [7] und Eric S. Raymond [8], um einen neuen Begriff zu finden, der zum einen von dem „kommunistischen“ Wörtchen „free“ wegführen, aber auch den Begriff „Freie Software“ nicht mehr so eng fassen sollte, sodass beispielsweise auch die BSD-Lizenz davon erfasst wird. Das Ergebnis war der Begriff „Open Source“ und die Gründung der Open Source Initiative (kurz OSI [9]) (S. 70 f).

Die wichtigsten Punkte der Open-Source-Definition sind, dass die Software frei verteilt werden darf, der Quellcode aber offen liegen muss und Ableitungen erlaubt sein müssen (S. 75). Der Begriff hat sich bis heute aber nicht richtig gegenüber Freier Software durchgesetzt, da der Name „Open Source“ vor allem eben Quelloffenheit propagiert, man aber nicht direkt schließen kann, dass auch eine Veränderung des Codes erlaubt sein muss. Etwas, was sich viele Firmen zu Eigen machen, um mit dem Begriff „Open Source“ werben zu können (S. 76).

Das führt auch dazu, dass „Freie Software“ und „Open Source“ in den meisten Open-Source-Projekten synonym verwendet werden. Einzig die extremen Anhänger beider Lager bestehen auf einer Differenzierung der Begriffe [10] (siehe auch Wikipedia [11]).

Über das Buch

Das Buch „Die Anarchie der Hacker“ von Christian Imhorst gibt einen sehr guten Einblick in

die Entstehungsgeschichte Freier Software, aber ebenso auch in die der Open-Source- und proprietären Software.

Aufgrund des Datums der Ausgabe (Ende 2004, Anfang 2005) sind die Informationen bzw. die Schlüsse darin aber leider etwas veraltet. Sätze wie „Trotzdem war das Mozilla-Projekt nie wirklich erfolgreich, weil der Aufbau einer Entwickler-Gemeinschaft nicht wirklich funktioniert hat, was allerdings mehr an der Organisation des Projektes lag als an der Lizenz.“ (S. 71) lesen sich seltsam, wenn man bedenkt, dass der Browser Firefox des Mozilla-Projektes im Jahr 2010 einen Marktanteil von über 50 % hat [12]. Von „nicht erfolgreich“ kann also aus heutiger Sicht keine Rede sein, auch wenn es vor fünf Jahren anders aussah. Davon abgesehen treffen die meisten Aussagen, die sich auf das letzte Jahrtausend beziehen, dennoch zu.

Richard Stallmans Einfluss ist zwar der Aufhänger des Textes, dennoch nimmt er nicht den Großteil des Buches für sich in Anspruch. Imhorst wechselt immer wieder auch auf andere Bereiche und wichtige Persönlichkeiten der damaligen Geschichte. Das Buch ist also keine Biographie von Richard Stallman, sondern sieht eher die Freie-Software-Bewegung im Vordergrund.

Sprachlich ist der Text sehr gut und selten langweilig, obwohl er mit zahlreichen Zitaten und Fußnoten gespickt ist. Da es eine Erstauflage ist, enthält das Buch noch kleinere Rechtschreibfehler, die aber wirklich nur sehr selten zu finden und damit zu verschmerzen sind.

Wer Interesse an der Person Richard Stallman und vor allem an Freier Software gefunden hat, sollte sich „Die Anarchie der Hacker“ unbedingt durchlesen.

Ganz im Sinne Freier Software

Und so, wie Richard Stallman es propagiert, kann das Buch nicht nur bei diversen Buchhändlern für 19,90 Euro bezogen werden, nein, seit dem 1. Mai 2010 [13] ist es auch in digitaler Form kostenlos unter der Creative-Commons-Attribution-NonCommercial-ShareAlike-Lizenz [14] auf der Seite des Autors Christian Imhorst [15] verfügbar. Der Tectum Verlag hält nur noch die Rechte an der (kommerziellen) Verwertung der Druckausgabe in Deutschland.

„Die Anarchie der Hacker“ kann direkt als HTML angeschaut oder im mobilen PDF- oder EPUB-Format heruntergeladen werden. Die CC-Lizenz erlaubt die Veröffentlichung des Buches (oder Ausschnitten) sowie Veränderungen davon, solange der Autor weiterhin genannt und das Werk nicht kommerziell vertrieben wird.

Wer nach dieser Rezension also Interesse daran gefunden hat, wie die Freie-Software-Bewegung entstanden ist und was Richard Stallman damit zu tun hatte, kann sich das Buch in digitaler Form sofort besorgen.

Redaktioneller Hinweis: *Vielen Dank an den Tectum Verlag für die schnelle und unkomplizierte Bereitstellung eines gedruckten Rezensionsexemplares.*

Buchinformationen

Titel	Die Anarchie der Hacker. Richard Stallman und die Freie-Software-Bewegung
Autor	Christian Imhorst
Verlag	Tectum, 2004
Umfang	89 Seiten
ISBN	978-3828887695
Preis	10,90 Euro (gedruckt) / Download kostenlos

LINKS

- [1] <http://de.wikipedia.org/wiki/Anarchismus>
- [2] http://de.wikipedia.org/wiki/Richard_Stallman
- [3] <http://www.gnu.org/licenses/gpl.html> 
- [4] <http://www.freie-gesellschaft.de/wiki/Hackerethik>

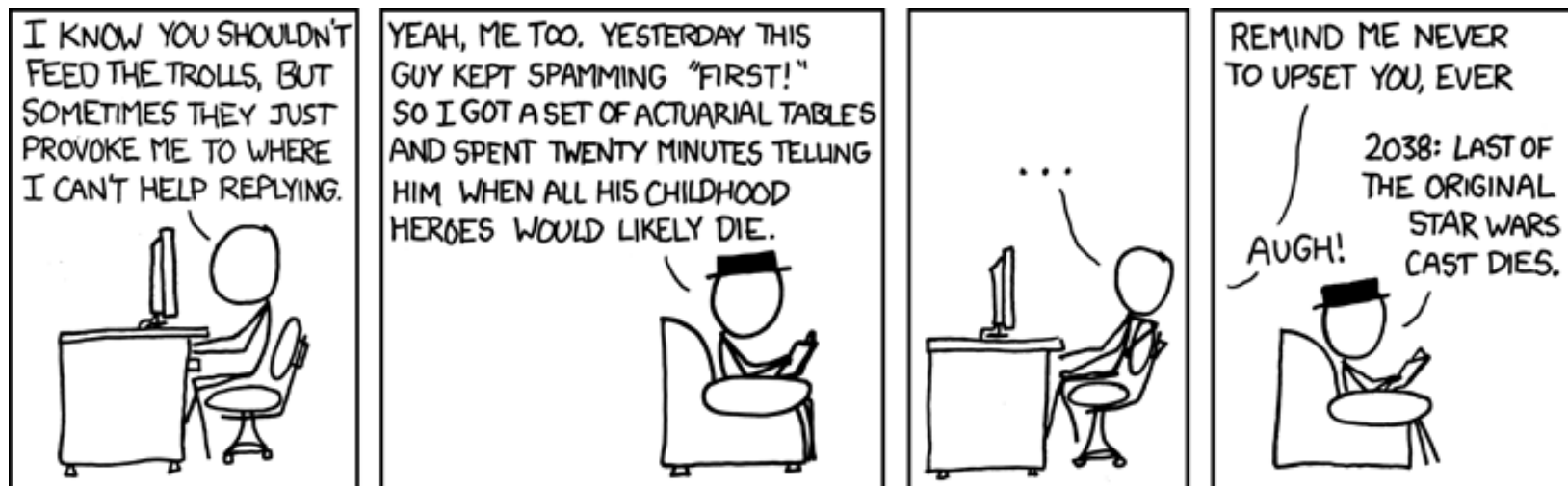
- [5] <http://www.gnu.org/software/emacs/> 
- [6] <http://de.wikipedia.org/wiki/BSD-Lizenz>
- [7] http://de.wikipedia.org/wiki/Bruce_Perens
- [8] http://de.wikipedia.org/wiki/Eric_S._Raymond
- [9] http://de.wikipedia.org/wiki/Open_Source_Initiative
- [10] <http://www.gnu.org/philosophy/free-software-for-freedom.de.html>
- [11] http://de.wikipedia.org/wiki/Freie_Software
- [12] <http://www.webmasterpro.de/portal/news/2010/03/04/webanalyse-firefox-ueber-50-in-deutschland.html>
- [13] <http://www.datenteiler.de/free-as-in-freedom-die-anarchie-der-hacker/>

- [14] <http://creativecommons.org/licenses/by-nc-sa/3.0/de/>
- [15] <http://www.datenteiler.de/>

Autoreninformation 

Dominik Wagenführ liest gerne und viel. Ab und zu fällt ihm dabei auch ein Buch über Linux oder Freie Software in die Hände.

Diesen Artikel kommentieren 



„Actuarial“ © by Randall Munroe (CC-BY-NC-2.5), <http://xkcd.com/493>

Rezension: CouchDB – kurz & gut von Hans-Joachim Baader

Das kleine und preiswerte Buch „CouchDB – kurz & gut“ des O'Reilly-Verlags ist als Einstiegshilfe für den dokumentenbasierten Datenbankansatz und auch als Nachschlagewerk für den CouchDB-Praktiker gedacht.

Redaktioneller Hinweis: Der Artikel „CouchDB – kurz & gut“ erschien erstmals bei Pro-Linux [1] unter der GNU Free Documentation License [2].

Kaum eine ernsthafte Anwendung kommt heute ohne Datenbank aus. Oft ist es eine relationale Datenbank mit SQL, die dabei zum Einsatz kommt. Man könnte daher leicht den Eindruck bekommen, dass relationale Datenbanken die IT dominieren. Doch das ist keineswegs so sicher. Datenbanken gab es schon, bevor relationale Datenbanken aufkamen. Und nach dem Entstehen von relationalen und objektrelationalen Datenbanken war die Forschung nicht beendet – neue Datenbanktypen wurden konzipiert, und momentan scheinen dokumentenorientierte Datenbanken besonders angesagt zu sein.

CouchDB [3] ist eine solche Datenbank (siehe auch „CouchDB – Datenbank mal anders“, freiesMagazin 06/2010 [4]). Für Kenner von SQL-Datenbanken dürfte es zunächst einmal schwer vorstellbar sein, wie man damit Daten und sogar Relationen speichert und wie man sie wiederfindet – denn ein Schema gibt es genauso wenig wie SQL. Der Unterschied zu relationalen Datenbanken könnte also kaum größer sein.

Hier will das kleine und preiswerte Buch „CouchDB – kurz & gut“ des O'Reilly-Verlags Abhilfe schaffen. Laut Umschlagtext ist es als Einstiegshilfe für den dokumentenbasierten Datenbankansatz und auch als Nachschlagewerk für den CouchDB-Praktiker gedacht.

Das Buch

Das Buch besitzt 11 Kapitel und keine Anhänge. Im Vergleich zu anderen Titeln der „kurz & gut“-Reihe, die in einem kompakten Format hergestellt wird, ist es mit 200 Seiten recht umfangreich ausgefallen. Nach einer kurzen Einführung wird in Kapitel 2 die Installation behandelt. Der Autor bezieht sich überwiegend auf Version 0.10 von CouchDB und warnt davor, dass die Informationen schnell veralten könnten, wenn die Entwicklung von CouchDB voranschreitet. Gelegentlich erwähnt er die Entwicklerversion 0.11.0, die einige erweiterte Funktionen enthält. Stand 0.11.0 ein Veröffentlichungskandidat für Version 1.0, welches Mitte Juli 2010 erschienen ist [5]. Man könnte dieses Kapitel für entbehrlich halten, aber es gibt auf der Webseite von CouchDB keine Installationsanleitung – und im Quellcodepaket nur eine englischsprachige.

Kapitel 3 beschreibt die Konfiguration. Wegen des begrenzten Platzes des Buches beschränkt es sich auf eine Auflistung der Optionen der Konfigurationsdatei mit minimalen Erläuterungen. Zumindest bei einigen weniger klaren Punkten wäre eine bessere Erklärung angebracht gewesen.

Als Referenz für die Optionen dürfte das Kapitel allerdings nützlich sein.

Das nächste Kapitel gibt eine kurze Übersicht über die Entwicklungsumgebung, das sind in diesem Fall die zur Verfügung stehenden Bibliotheken in verschiedenen Programmiersprachen und die grafische Oberfläche CouchApp. Kapitel 5 widmet sich dem CouchDB-Server. Dieser ist interessanterweise in Erlang geschrieben und wird über das HTTP-Protokoll (mit einigen Erweiterungen) angesprochen. Das heißt, jede Anwendung, die HTTP-Kommandos absetzen kann (Browser, Kommandozeilenprogramme wie curl und wget oder eigene Software) kann mit dem Server interagieren. Dokumente werden im JSON-Format ein- und ausgegeben. Das geschieht nicht von ungefähr. JSON ist syntaktisch eine JavaScript-Datenstruktur, und Abfragen an CouchDB (Views) werden in JavaScript geschrieben. Außerdem ist JSON ein kompaktes Format, kompakter als XML. Das Kapitel beschreibt die erlaubten Datentypen in den Dokumenten, globale Abfragen des Servers und die vordefinierten JavaScript-Funktionen.

Im Kapitel 6 geht es um Datenbanken in CouchDB: Wie man sie anlegt, verdichtet (weil sonst der Verbrauch an Festplattenplatz immens wird) und abfragt. Auch die Replikation wird behandelt, die in CouchDB bereits eingebaut ist und sicher zu ihrem Erfolg mit beigetragen hat. Kapitel 7 widmet sich den Daten in diesen Da-

tenbanken, also den Dokumenten. Dabei stehen das Anlegen, Auflisten und andere Operationen sowie die Versionierung im Mittelpunkt.

Die für Neueinsteiger spannendste Frage ist sicher, wie Abfragen in CouchDB funktionieren. Diese wird in Kapitel 8 beantwortet. Dokumente besitzen Attribute, diese entsprechen im Prinzip den Feldern eines Datensatzes in relationalen Datenbanken, nur dass die Attribute beliebig komplex sein können. Mit dem Map-Reduce-Verfahren kann man nun beliebige Abfragen konstruieren und dabei Verknüpfungen zwischen den Dokumenten herstellen. Das Verfahren besteht aus zwei getrennten Schritten, Map und Reduce, die wie bereits angedeutet in JavaScript programmiert werden. Solche Abfragen definieren Views, die in der Datenbank – an beliebiger Stelle – gespeichert werden können.

Kapitel 9 gibt einen Überblick über die optionalen Transformationsfunktionen, die beim Speichern, Auslesen und Ändern von Dokumenten aufgerufen werden können. Der Sicherheit und Validierung widmet sich Kapitel 10. Eine einfache Beispielanwendung, die Aufgabenlisten für mehrere Benutzer verwaltet, bildet das letzte Kapitel. Ein kurzer Index rundet das Buch ab. Die verfügbaren 200 Seiten wurden nicht voll ausgeschöpft, sodass am Schluss noch einige leere Seiten zu finden sind.

Fazit

Man sollte von „CouchDB – kurz & gut“ nicht zu viele tiefere Informationen erwarten; die-

se kann das Buch wegen der begrenzten Größe nicht liefern. Es eignet sich recht gut, um eine grobe Vorstellung von den Konzepten von CouchDB zu bekommen, und regt vielleicht sogar zu eigenen Experimenten an. Für die Nutzer, die CouchDB regelmäßig nutzen, ist es eine bequeme Referenz zu verschiedenen Punkten. Echte Mängel sind nicht zu entdecken, und seinen Preis ist es jedenfalls wert. Wer das Buch als Einführung in CouchDB liest, sollte nach der Lektüre überzeugt sein, dass CouchDB kein Spielzeug ist, sondern eine professionelle Datenbank, deren Möglichkeiten mit relationalen Datenbanken mindestens gleichauf liegen, wie Funktionen wie Replikation, ACID-Eigenschaften, MVCC, Zugriffskontrolle sowie das Äquivalent von Triggern und Stored Procedures belegen. Lediglich über die Geschwindigkeit schweigt sich „CouchDB – kurz & gut“ aus. Ein konkreter Vergleich mit anderen Datenbanken bleibt wohl den Benutzern überlassen.

Buchinformationen

Titel	CouchDB – kurz & gut
Autor	Mario Scheliga
Verlag	O'Reilly, 2010
Umfang	200 Seiten
ISBN	978-3897215597
Preis	9,90 EUR (gedruckt)

LINKS

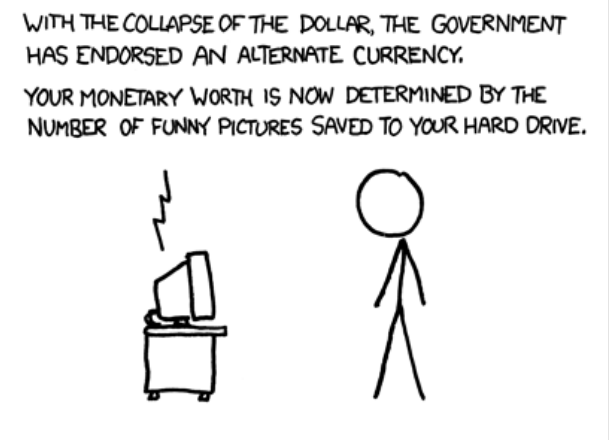
- [1] <http://www.pro-linux.de/artikel/2/1193/couchdb-kurz-und-gut.html>
- [2] <http://www.gnu.org/copyleft/fdl.html> 
- [3] <http://couchdb.apache.org/> 

- [4] <http://www.freiesmagazin.de/freiesMagazin-2010-06>
- [5] <http://www.pro-linux.de/news/1/15916/couchdb-10-freigegeben.html>

Autoreninformation

Hans-Joachim Baader befasst sich seit 1993 mit Linux. 1994 schloss er sein Informatikstudium erfolgreich ab, machte die Softwareentwicklung zum Beruf und ist einer der Betreiber von Pro-Linux.de.

Diesen Artikel kommentieren 



I HAVE BEEN PREPARING FOR THIS MOMENT MY WHOLE LIFE.

„Alternate Currency“ © by Randall Munroe
(CC-BY-NC-2.5), <http://xkcd.com/512>

Rezension: Beginning CouchDB von Jochen Schnelle

Zur dokumentenorientierten Datenbank CouchDB [1] gab es schon den Artikel „CouchDB – Datenbank mal anders“ in freiesMagazin 06/2010 [2]. Wer sich weitergehend mit diesem Thema beschäftigen möchte, der sollte einen Blick auf das englischsprachige Buch „Beginning CouchDB“ aus dem Apress Verlag werfen. Das Buch in der hier rezensierten ersten Auflage behandelt zwar CouchDB 0.10, welches bei Drucklegung aktuell war, kann aber auch größtenteils auf die im April 2010 erschienene Version 0.11 angewendet werden.

Wie der Titel schon vermuten lässt, möchte das Buch den Leser beim Einstieg in die Datenbank Schritt für Schritt begleiten. Dabei richtet sich das Buch gleichermaßen an Datenbankneueinsteiger als auch an Nutzer, die bereits mit anderen, z. B. relationalen, Datenbanken gearbeitet haben. Programmiererfahrung muss man nicht unbedingt haben, eine (minimale) Erfahrung mit JavaScript erleichtert es aber ungemein, die späteren Beispiele zu verstehen.

Das Buch beginnt mit der kurzen Geschichte von CouchDB, mit welcher Motivation CouchDB entworfen wurde, welche Programmiersprachen und Bibliotheken zum Einsatz kommen, sowie einem Überblick über die Versionshistorie. Danach folgt ein recht ausführlicher Teil zur Installation von CouchDB. Positiv fällt hier auf, dass die Plattformen Linux, MacOS und Windows gleichermaßen

bedacht werden. Hierauf folgt eine grundlegende Einführung in CouchDB, also wie man eine Datenbank und Dokumente anlegt und Abfragen gestaltet. Alle Beispiele in diesem Abschnitt verwenden das Kommandozeilenprogramm curl zur Kommunikation mit CouchDB.

Im zweiten Teil des Buches, welcher den Hauptteil zur Nutzung von CouchDB ausmacht, wird zuerst die webbasierte, grafische Oberfläche namens „Futon“, welche standardmäßig enthalten ist, erläutert. Dabei wird ebenfalls das Anlegen von Datenbanken und Dokumenten erklärt, ebenso das Generieren von Abfragen, den sogenannten „Views“. Dies erfolgt wesentlich ausführlicher und detaillierter als in der grundlegenden Einführung im ersten Teil des Buchs. Im zweiten Teil bekommt der Leser auch eine Einführung in JSON, das Format, in dem CouchDB Daten speichert und zurück gibt. Im Rahmen der Erläuterung von Views erfolgt auch eine verständliche Erläuterung der zugrunde liegenden „Map/Reduce“-Technik.

Der dritte Teil des Buchs ist mit „Advanced CouchDB Topics“ überschrieben und ist in sich wiederum zweigeteilt. Zum einen wird auf Themen wie Konfiguration, Benutzerrechte, Replikation und „Load Balancing“ eingegangen. Zum anderen wird gezeigt, wie man CouchDB in Kombination mit den Programmiersprachen Python und Ruby nutzt und wie man eigenständige Anwendungen mit Hilfe von „CouchApp“ realisiert.

Den Abschluss des Buchs bildet eine vollständige Übersicht über die HTTP-API von CouchDB sowie eine Überblick über HTTP-Requests, welche im Umgang mit CouchDB ebenfalls genutzt werden.

Das Buch ist durchweg in gut verständlichem, wenig technischem Englisch geschrieben. Da der Autor eher einen lockeren „Plauderton“ als eine nüchterne Darstellung pflegt, ist das Buch flüssig und kurzweilig zu lesen. Trotzdem ist das Buch durchweg sachlich und kompakt. Der Verständlichkeit kommt sicherlich auch zugute, dass sehr viele Bildschirmfotos zur Unterstützung des Texts enthalten sind. Die Bilder sind aber, wie der komplette Rest des Buchs auch, ausschließlich in Schwarz-Weiß gehalten. Weiterhin enthält das Buch sehr viele Beispiele, teils kurze, teils auch etwas ausführlichere, je nach Notwendigkeit des gerade behandelten Themas.

Im größten Teil des Buchs geht der Autor recht gemächlich – aber ohne langatmig oder langweilig zu werden – vor, sodass auch Einsteiger ohne Probleme folgen können. Nur an zwei Stellen nimmt das „Tempo“ kurz rapide zu. Dies ist zum einen bei einem der Beispiele zu Map/Reduce, einem etwas ausführlicheren JavaScript-Code so. Hier kann man ohne Kenntnisse in JavaScript das komplette Beispiel wohl nicht verstehen – aber der Sinn hinter der Vorgehensweise wird trotzdem klar. Die zweite Stelle ist der Abschnitt zu CouchApp. Hier kann man dem an sich



LINKS

sehr schönen Beispiel ohne Kenntnisse in HTML, CSS und JavaScript schwerlich folgen. Da der Abschnitt aber in sich abgeschlossen ist, kann man diesen auch ohne Weiteres überspringen, ohne Verständnisprobleme in den folgenden Abschnitten und Kapiteln zu haben.

Die Kapitel zur Nutzung von CouchDB als Datenbank für ein eigenes Python- bzw. Ruby-Projekt erfordert natürlich Kenntnisse in der jeweiligen Programmiersprache, zumindest auf dem Niveau eines fortgeschrittenen Einsteigers. Da aber auch diese Kapitel in sich abgeschlossen sind, können sie von Nicht-Programmierern einfach übersprungen werden.

Zusammenfassend kann gesagt werden, dass das Buch einen durchaus gelungenen Einblick

und Einstieg in CouchDB bietet. Wer das Buch einmal durchliest und die zahlreichen und verständlichen Beispiele nachvollzieht, der sollte keine Probleme haben, CouchDB für eigene Projekte einzusetzen. Die Auflistung der API im Anhang kann dabei immer wieder zum schnellen Nachschlagen genutzt werden, auch für erfahrene Programmierer.

Buchinformationen

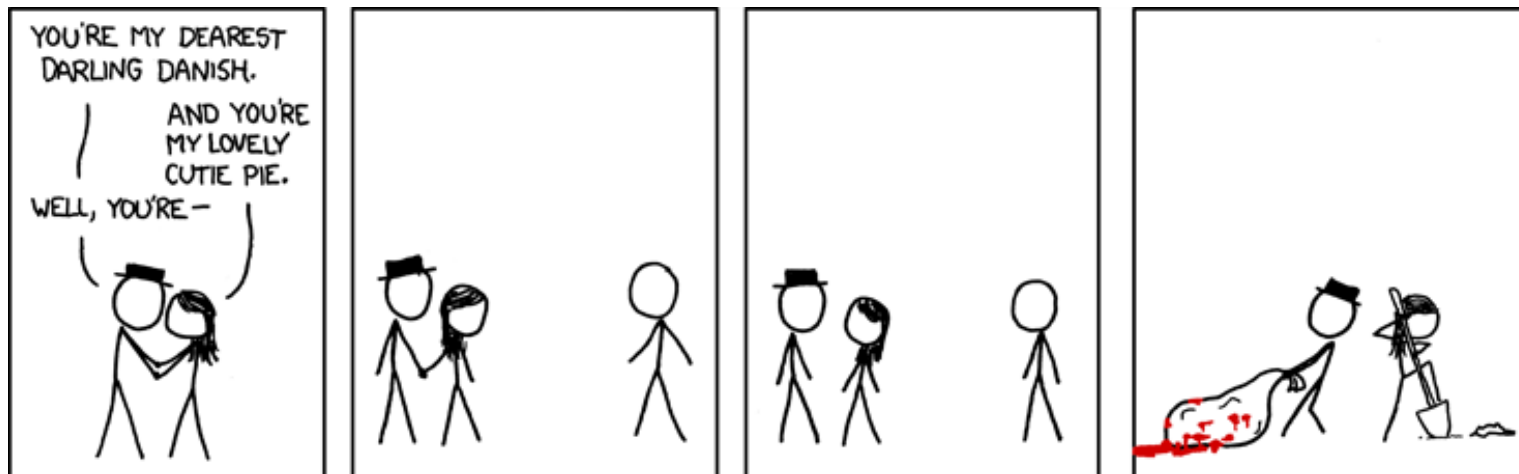
Titel	Beginning CouchDB
Autor	Joe Lennon
Verlag	Apress, 2009
Umfang	Umfang: 300 Seiten
ISBN	9781430272373
Preis	30,20 Euro

- [1] <http://couchdb.apache.org/>
- [2] <http://www.freiesmagazin.de/freiesMagazin-2010-06>

Autoreninformation

Jochen Schnelle ist zufällig im Internet auf CouchDB gestoßen und hat damit aus Neugier ein wenig herum experimentiert. Das Buch „Beginning CouchDB“ hat ihn beim Erlernen der Datenbank unterstützt.

Diesen Artikel kommentieren



„No One Must Know“ © by Randall Munroe (CC-BY-NC-2.5), <http://xkcd.com/515>



Leserbriefe

Für Leserbriefe steht unsere E-Mailadresse 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

Spieleausgabe Juli

❖ In Eurer Juli-Ausgabe wurde u. a. „Pioneers“ besprochen. Zur genannten Vorgehensweise für ein lokales Spiel gegen Computergegner eine kleine Ergänzung: Bei mir läuft das Spiel in Version 0.12.2-4 auf Ubuntu 9.10. In dieser Konstellation verweigert der Server bei der von Euch vorgeschlagenen Vorgehensweise die Anmeldung. Stattdessen greift folgender Weg:

- Server starten.
- Im Client „127.0.0.1“ statt „localhost“ angeben, um sich selbst anzumelden.
- Die Computergegner an der Konsole per `pioneersai -s 127.0.0.1 -p 5556 -n <Spielername>` anmelden.

René Franke

❖ Bei VICE sind die notwendigen ROM-Images auch dabei, wenn man sich den Quelltext vom

Projekt herunterlädt. Nur im Debian/Ubuntu-Paket fehlen sie, weil auch hier das Urheberrecht im Weg steht, wie bei anderer „Abandonware“. Wer also keinen originalen C64 (plus Diskettenlaufwerk) besitzt, dürfte auch keinen Emulator verwenden, der auf den Inhalt von den ROM-Bausteinen im C64 (und des Diskettenlaufwerks) angewiesen ist. **BlackJack** (Kommentar)

❖ Ich bin leidenschaftlicher OpenTTD-Spieler und möchte anmerken, dass OpenTTD nun vollständig frei ist. Es werden nicht länger die Transport Tycoon Deluxe Spieldateien gebraucht. OpenTTD liegt zudem bereits in der Version 1.0.2 vor und wird stets weiterentwickelt.

Tim (Kommentar)

❖ Vielen Dank für die vielen Hinweise. Die Artikel sind in der Regel bereits etwas älter gewesen, weswegen nicht alle Informationen darin dem aktuellen Stand der Zeit entsprechen. Aus dem Grund sind Ihre Anmerkungen um so wichtiger. **Dominik Wagenführ**

❖ Gamebase64 [1] ist eine Datenbank fast aller C64-Spiele (zur Zeit 21.000). Man kann die Datenbank herunterladen und mit einem Frontend wie jGamebase [2] durchsehen. Wenn man die Musik und Spiele sein Eigen nennt, kann man diese auch bequem aus dem Frontend heraus starten. **Frank Gerbig** (Kommentar)

❖ Ich finde es wirklich super, dem Klischee „auf Linux gibt es keine Spiele“ bemerkenswert entgegen

zu setzen und eine Ausgabe nur über Spiele zu machen. Echt super Idee!

Ich kenne noch ein super Spiel, das auf Linux funktioniert: Simutrans [3] [4]. Es ist eine Wirtschafts- und Transportsimulation, in der man Waren auf unterschiedlichsten Wegen transportieren muss um Geld zu erwirtschaften.

Markus Zeindl

❖ Danke für die beiden Tipps. Simutrans ist sicherlich ein interessantes Spiel, welches vor allem Wirtschaftssimulationsspieler erfreuen wird. Und auch die jGamebase klingt interessant.

Sollten Sie Lust haben, würden wir uns über Artikel zu den beiden Themen freuen. Natürlich kann auch jeder andere Leser da draußen sich den Themen annehmen und die Spielesartikellreihe fortsetzen, sodass in zwei Jahren wieder so eine schöne Sonderausgabe von freiesMagazin zustande kommt. **Dominik Wagenführ**

LINKS

[1] <http://www.gb64.com/>

[2] <http://jgamebase.sourceforge.net/>

[3] <http://www.simutrans.de/>

[4] <http://www.simutrans.com/>

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

Die Leserbriefe kommentieren

Veranstungskalender

Messen

Veranstaltung	Ort	Datum	Eintritt	Link
TechTalk: Network Security	München	17.08.2010	frei	http://www.opensourceschool.de
FrOSCon	Sankt Augustin	21.08-22.08.2010	5 EUR	http://www.froscon.de
FrOSCamp	Zürich	17.09.-18.09.2010	frei	http://frosccamp.org
Software Freedom Day	Weltweit	18.09.2010	frei	http://softwarefreedomday.org
Kieler Linuxtage	Kiel	01.10.-02.10.2010	frei	http://www.kieler-linuxtage.de
Ubucon	Leipzig	15.10.-17.10.2010	-	http://www.ubucon.de
TechTalk: Mobile Anwendungen mit Qt	München	19.10.2010	frei	http://www.opensourceschool.de
Brandenburger Linux-Infotag	Potsdam	06.11.2010	frei	http://blit.org/2010
OpenRheinRuhr	Oberhausen	13.11.-14.11.2010	3 EUR	http://www.openrheinruhr.de

(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.

Vorschau

freiesMagazin erscheint immer am ersten Sonntag eines Monats. Die September-Ausgabe wird voraussichtlich am 5. September unter anderem mit folgenden Themen veröffentlicht:

- Notepad++ und Notepad2 – Open-Source-Editoren für Windows
- Rezension: Python – Essential Reference

Es kann leider vorkommen, dass wir aus internen Gründen angekündigte Artikel verschieben müssen. Wir bitten dafür um Verständnis.

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.

Impressum

freiesMagazin erscheint als PDF und HTML einmal monatlich.
Redaktionsschluss für die August-Ausgabe: 22. August 2010

Kontakt

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

Autoren dieser Ausgabe

Hans-Joachim Baader **S.34**
Saskia Brückner **S.18**
Volker Duetsch **S.22**
Kirsten Roschanski **S.26**
Mathias Menzer **S.21**
Thorsten Schmidt **S.11**
Jochen Schnelle **S.4, S.36**
Dominik Wagenführ **S.24, S.31**

Logo-Design

Arne Weinberg ([GNU FDL](#))

ISSN 1867-7991

Erscheinungsdatum: 1. August 2010

Redaktion

Dominik Honnef Thorsten Schmidt
Dominik Wagenführ (Verantwortlicher Redakteur)

Satz und Layout

Tillo Bosshart Ralf Damaschke
Yannic Haupenthal Nico Maikowski
Sebastian Schlatow Matthias Sitte
Günther Wutz

Korrektur

Daniel Braun Frank Brungräber
Stefan Fangmeier Mathias Menzer
Karsten Schuldt Franz Seidl
Stephan Walter

Veranstaltungen

Ronny Fischer

Dieses Magazin wurde mit \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 und Beiträge in freiesMagazin unter der [GNU-Lizenz für freie Dokumentation \(FDL\)](#). Das Copyright liegt beim jeweiligen Autor. freiesMagazin unterliegt als Gesamtwerk ebenso der [GNU-Lizenz für freie Dokumentation \(FDL\)](#) mit Ausnahme von Beiträgen, die unter einer anderen Lizenz hierin veröffentlicht werden. Das Copyright liegt bei Dominik Wagenführ. Es wird die Erlaubnis gewährt, das Werk/die Werke (ohne unveränderliche Abschnitte, ohne vordere und ohne hintere Umschlagtexte) unter den Bestimmungen der GNU Free Documentation License, Version 1.2 oder jeder späteren Version, veröffentlicht von der Free Software Foundation, zu kopieren, zu verteilen und/oder zu modifizieren. Die xkcd-Comics stehen separat unter der [Creative-Commons-Lizenz CC-BY-NC 2.5](#). Das Copyright liegt bei [Randall Munroe](#).