



NEWS

DAS CLUBCOMPUTER MAGAZIN

CLUBPOCKETPC

Datalogger iGot-U GT-600

Datenbank synchronisieren

CLUBDEV

Homepage Download

Windows 7 Entwicklung

Vektorgrafik auf Webseiten

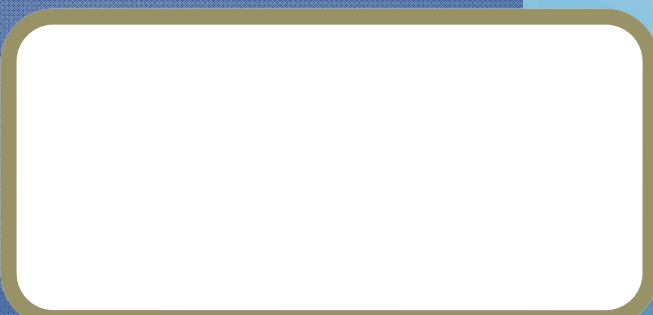
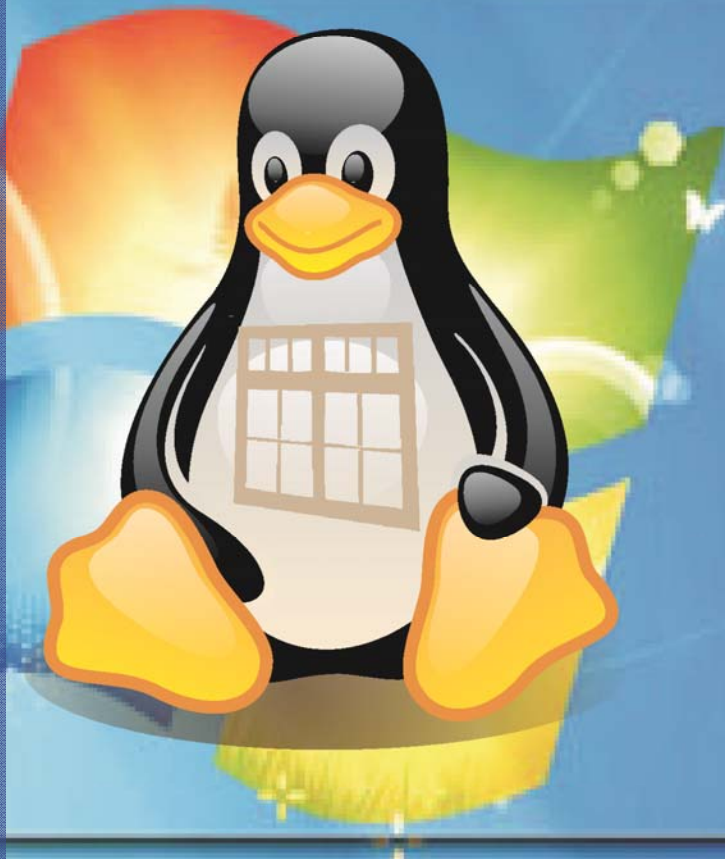
Google Charts

CLUBDIGITALHOME

Mobiles Wertkarteninternet

CLUBSYSTEM

Linux für Umsteiger



Autoren

Belcl Paul 1966

6,8



EDV-Koordinator, Trainer für Windows Mobile, Promotor bei Microsoft Österreich, Direktor des ClubPocketPC
Firma BELCL EDV-Koordination & Systemberatung
Club CCC
Hobbies Fotografieren, Reisen, Segway
pbelcl@ccc.at
<http://www.belcl.at/>

Berger Christian

2



Karikaturist und Comiczeichner für verschiedene Kärntner Zeitungen
Firma Karicartoons
[karicartoons.aon.at](http://www.karicartoons.aon.at)
<http://www.bergercartoons.com/>

Fiala Franz Dipl.-Ing. 1948

1,2,18,19,23,24



Leitung der Redaktion und des Verlags der PCNEWS, Obmann des PCC; Lehrer für Nachrichtentechnik und Elektronik i.R.
Firma PCNEWS
Werdegang BFPZ-Arsenal, TGM Elektronik, Web-Entwicklung
Club CCC MCCA PCC VIT
Absovent TU-Wien, Nachrichtentechnik
Privates verheiratet, 1 Kind
[fiala@clubcomputer.at](http://www.fiala@clubcomputer.at)
<http://www.fiala.cc/>

Hartl Günter Ing. 1963

12



Wirtschaftsingenieur, Systemadministrator für Windows Clients und Linux Server in Logistikcenter
Club CCC
Hobbies Krav Maga, Windsurfen, Lesen
gghartl3@gmail.com

Pflegerl Siegfried (Gruppe Or-Om)

4

Schriftsteller Philosophie, Kunst- und Evolutionstheorie
Club PCC
siegfried.pflegerl@chello.at
<http://portal.or-om.org/>

Reinwart Thomas 1973

21



Softwareentwickler, MCAD, MCSD, MCDBA, MCSA, MCSE Zertifizierungen
Firma Reinwart
office@reinwart.com
<http://www.reinwart.com/>

Schneider Christian

10



AHS-Lehrer für Deutsch
schneider@gmx.at

Weissenböck Martin Dir.Dr. 1950

3



Direktor der HTL Wien 3 Rennweg, Leiter der ADIM und Autor von ADIM-Skripten, Leiter der ARGE Telekommunikation
Schule HTL Wien 3R, ADIM
Club ADIM CCC PCC
martin@weissenboeck.at
<http://www.weissenboeck.at/>



Impressum

Impressum, Offenlegung

Richtung Auf Anwendungen im Unterricht bezogene Informationen über Personal Computer Systeme. Berichte über Veranstaltungen des Herausgebers.
Erscheint 5 mal pro Jahr; Feb, Apr, Jun, Sep, Nov
Verleger PCNEWS-Eigenverlag
 Siccardsburggasse 4/1/22 1100 Wien
 0664-1015070 FAX: 01-6009933-9210
pcnews@pcnews.at
<http://www.pcnews.at/>
Herausgeber ClubComputer
 Siccardsburggasse 4/1/22 1100 Wien
 01-6009933-11 FAX: -12
office@clubcomputer.at
<http://www.clubcomputer.at/>
<http://heilm.ccc.at/>
<http://mailto:enable.ccc.at/>
Druck Ultra Print
 Technická 7 SK-82104 Bratislava
<http://www.ultraprint.eu/>
Versand GZ 022031324

ClubComputer

Leitung, CCC Werner Illsinger
 01-6009933-220 FAX: -9220
werner.illsinger@clubcomputer.at
<http://illsinger.at/>
<http://illsinger.at/blog/>
PCNEWS, PCC Franz Fiala
 01-6009933-210 FAX: -9210
franz.fiala@clubcomputer.at
<http://www.fiala.cc/>
<http://www.fiala.cc/blogpcnews/>
Marketing Ferdinand De Cassan
 01-6009933-230 FAX: -9230
ferdinand.de.cassan@clubcomputer.at
<http://spielefest.at/>
CC|Akademie Georg Tsamis
 01-6009933-250 FAX: -9250
georg.tsamis@clubcomputer.at
ClubPocketPC Paul Belcl
 01-6009933-288 FAX: -9288
paul.belcl@clubcomputer.at
<http://www.belcl.at/>
<http://blog.belcl.at/>
ClubDigitalHome Christian Haberl
 01-6009933-240 FAX: -9240
christian.haberl@clubcomputer.at
<http://blog.this.at/>
WebDesign Herbert Dobsak
 01-2637275 FAX: 01-2691341
dobsak@ccc.or.at
<http://www.dobsak.at/>
Digitalfotografie Andreas Kunar
andreas.kunar@clubcomputer.at
<http://www.fotocommunity.de/pc/account/myprofile/16403>
Linux Günter Hartl
 ClubComputer-Portal: „Gunter.Hartl“
Konto BAWAG-PSK
 Konto: 17710-812-896 BLZ 14.000
 lautend auf: ClubComputer
 BIC: BAWAAT333 IBAN: AT741400017710812896
Zugang Einwahl: 080400222
 DNS1/DNS2: 194.50.115.132 194.50.115.170
 Alternativ: 213.129.226.2 213.129.226.2
Clublokal HTL, 1030 Wien, Rennweg 89b

PCNEWS-118

Kenzeichnung ISSN 1022-1611, GZ 022031324 M
Layout Microsoft Publisher 2007
Herstellung Bogenoffset, 80g
Erscheint Wien, April 2010
Texte <http://pcnews.at/?id=PCN118>
Kopien Für den Unterricht oder andere nicht-kommerzielle Nutzung frei kopierbar. Für gewerbliche Weiterverwendung liegen die Nutzungsrechte beim jeweiligen Autor. (Gilt auch für alle am PCNEWS-Server zugänglichen Daten.)
Werbung A4: 1 Seite 52,- EURO U2,3,4 782,- EURO Beilage: bis 50g 138,- EUR pro 1000 Stück
Bezug 1 Heft: 5,- EURO (zuzüglich Versand)
 5 Hefte: 20,- EURO (1 Jahr, inklusive Versand)
 kostenlos für Mitglieder von ClubComputer
Hinweise Druckfehler und Irrtümer vorbehalten.
 Alle erwähnten Produktnamen sind eingetragene Warenzeichen der entsprechenden Erzeuger.

Inserenten

MTM-Systeme

32



Ing. Gerhard Muttenthaler
 Hadrawagasse 36 1220 Wien
 01-2032814 FAX: 2021303 Handy: 0664-4305636
g.muttenthaler@mtm.at
<http://www.mtm.at/>

Produkte uC/uP-Entwicklungswerkzeuge, Starterkits, Industrie-computer, Netzqualitätsanalyzer, USV-Anlagen
Vertretung Tasking, PLS, Infineon, TQ-Components, Kontron, Dranetz-BMI, Panasonic, Dr. Haag, HT-Italia, Dr. Kneff
Erreichbar U1-Kagrán, 26A bis Englisch-Feld-Gasse

CC | Skripten

Martin Weissenböck

ADIM, Arbeitsgemeinschaft für Didaktik, Informatik und Mikroelektronik
 1190 Wien, Gregor Mendel Straße 37
 Tel.: 01-314 00 288 FAX: 01-314 00 788

Nr	Titel
38	Turbo Pascal (Borland)
39	RUN/C Classic
40	Turbo-C (Borland)
41-3	Turbo/Power-Basic
43-2	DOS
43-3	DOS und Windows
47	Turbo-Pascal (Borland)
49	Quick-Basic (Microsoft)
50	C++ (Borland)
53-3	AutoCAD I (2D-Grafik)
53-5	AutoCAD I (2D-Grafik)
54	AutoCAD II (AutoLisp+Tuning)
55	AutoCAD III (3D-Grafik)
56	Grundlagen der Informatik
61	Visual Basic (Microsoft)
63	Windows und Office
81	Linux
110	Best Of VoIP (CD)
111	All About VoIP (DVD)
191,192	Angewandte Informatik I + II
201,202	Word I+II
203	Excel
205,206	Access I+II
221	HTML
222	HTML und CSS
223	JavaScript,
227	VB.NET
231,232	Photoshop I+II
237, 238	Dreamweaver, Interaktive und animierte Webseiten

Bestellhinweise, Download

<http://www.adim.at/>
<http://adim.at/download/>
<http://www.adim.at/dateien/BESTELL.pdf>



One and Millions of Chairs

Zu christian-rupp.com/invasion/index.htm, Gruppe Or-Om

Entrance

„One and Three Chairs“¹ ist ein Kunstwerk von Joseph Kosuth aus dem Jahre 1965. Das Bild enthält einen Stuhl, eine Photographie dieses Stuhles und einen Lexikoneintrag mit einer Definition des Wortes ‚Stuhl‘. Kosuth's Arbeit „One and three chairs“ markiert seinen Beginn der Konzeptkunst. Nach Platons Gedanken zur Idee eines Gegenstandes hinterfragt er seine verschiedenen Erscheinungsformen - wirklicher Sessel, ein Foto desselben und ein Eintrag über ihn in einem Wörterbuch.

Der Fotograf gibt den Stuhl an seinem gegenwärtigen Ort im Raum wieder, so dass sich das Kunstwerk jedes Mal verändert, wenn es in einer neuen Umgebung installiert wird. Zwei Elemente der Arbeit bleiben aber bestehen: das exakte Abbild eines Lexikonartikels über das Wort 'Stuhl' und ein Diagramm, das Anweisungen für den Aufbau enthält. Diese beiden konstanten Bildelemente sind von Kosuth signiert worden. Eine Anweisung für den Installateur lautet dabei, einen Stuhl zu wählen, diesen vor einer Wand zu platzieren und ein Foto davon zu machen. Dieses Foto soll dann auf die Größe des Stuhles vergrößert und wiederum zur Linken des Stuhles aufgehängt werden. Schlussendlich soll eine Vergrößerung des Lexikonartikels zur Rechten des Stuhles aufgehängt werden, dessen obere Lehne mit der Fotografie abgestimmt werden soll.²

http://de.wikipedia.org/wiki/One_and_Three_Chairs.

Monobloc ist die fachsprachliche und international gängige Bezeichnung für den preiswerten Plastikstapelstuhl. Merkmale des Monobloc sind seine Spritzgussfertigung aus Polypropylen^{3,4} in einem Arbeitsgang, Stapelbarkeit und sein günstiger Preis. Der Monobloc ist weltweit verbreitet und vermutlich das in größter Zahl hergestellte Möbel. Der Monobloc wird als Sessel, Stuhl und Hocker gefertigt. Die ersten Monobloc-Stühle entstanden in den frühen 1970er Jahren. Der Erfinder ist unbekannt⁵.

<http://de.wikipedia.org/wiki/Monobloc-Stuhl>.

Die Geburt der Stühle

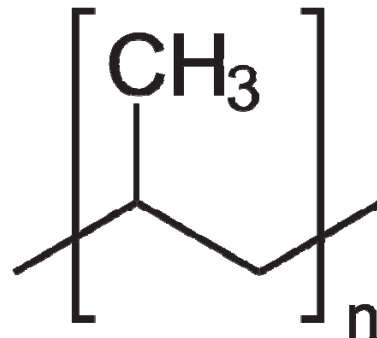
Über den zwei Fotografien in <http://christian-rupp.com/invasion/index.htm> aus 1st international openart residency '08⁶ liegt eine düster rätselhaft-schwül-surreale Stimmung aus Ver- (-fall, -rottung, -lassenheit, -trocknung, -wesung, -brennung und -rostung) von Gras, Gestängen, Zäunen, Steinen, Sand, Beton, Sträuchern und Bäumen eines Strandstückes mit Bungalows auf den *Dream Islands* in der Nähe von Eritrea. Das bleiern stille Meer und das trostlose Licht-Schattenspiel einer fahl-verhangenen Sonne verharren in flirrende Starre der Zeit gebannt.

Welches Geheimnis aber verbergen die in strengen Formationen platzierten Stühle? Die Forschung näherte sich dem Rätsel schrittweise: Die weltweit größte Müllkippe erstreckt sich über eine beachtliche Fläche doppelt so groß wie die USA, und ist dennoch auf Satellitenaufnahmen nicht zu sehen. Unmengen von Plastikmüll treiben als 'plastic soup' unter der Wasseroberfläche im nördlichen Pazifischen Ozean, von der 500-Meilen-Zone vor der kalifornischen

Küste bis fast nach Japan. Hochdruckgebiete sorgen für Meeresströmungen in denen sich zwei gigantische, nahezu geschlossene Areale des im Uhrzeigersinn strudelnden 'Great Pacific Garbage Patch' östlich und westlich von Hawaii gebildet haben, die der amerikanische Meeresforscher **Charles Moore**⁶ auf über 100 Millionen Tonnen Treibgut schätzt. Die von Moore gegründete *Algalita Marine Research Foundation*⁷ untersucht aktuell mit der *Winter Gyre Expedition* die im Nord-Pazifik treibenden Müllfelder, speziell in der *Transition Zone Chlorophyll Front (TZCF)*. In der Übergangszone zwischen subtropischen Meereströmungen mit niedrigem Chlorophyllgehalt und subarktischen mit sehr hohen Chlorophyllanteilen hatte AMRF 1999 in einer ersten bahnbrechenden Studie das unglaubliche Verhältnis von 6:1 Plastikmüll zu Zooplankton (in der Trockenmasse) festgestellt.

1997 bezifferte eine UNEP-Studie die jährlich in die Meere eingebrachte Müllmenge auf mindestens 6,4 Millionen Tonnen, bis zu 13.000 Plastikteile treiben pro Quadratkilometer in den Ozeanen. Der Müll im 'Pacific Garbage Patch' ist zu 90% Plastik, ein Fünftel davon ‚entsorgen‘ Schiffe und Gas- oder Ölbohrinsel, bzw. wird bei Schiffsunglücken ins Meer gespült. Wird der vor den Küsten Asiens und Amerikas treibende Müll von der nord-pazifischen Strömung erfasst, verbleibt er nach Angaben der *National Oceanic and Atmospheric Administration (NOAA)* mindestens 16 Jahre in dem Megastrudel des 'North Pacific Gyre'.

Konventionelle Plastikverpackungen, Getränkeflaschen etc. werden im Meer nicht abgebaut, sondern nur bis auf Planktongröße verkleinert, auf Mikropartikel von weniger als 0,3mm Durchmesser. **Richard Thompson** von der *University of Plymouth* geht von weitreichenden Folgen aus, die dieser Mikromüll gerade für benthische Organismen haben kann (am Meeresgrund lebende Tiere und Pflanzen mit einer besonderen Filterfunktion für das Meer). Thompson hat



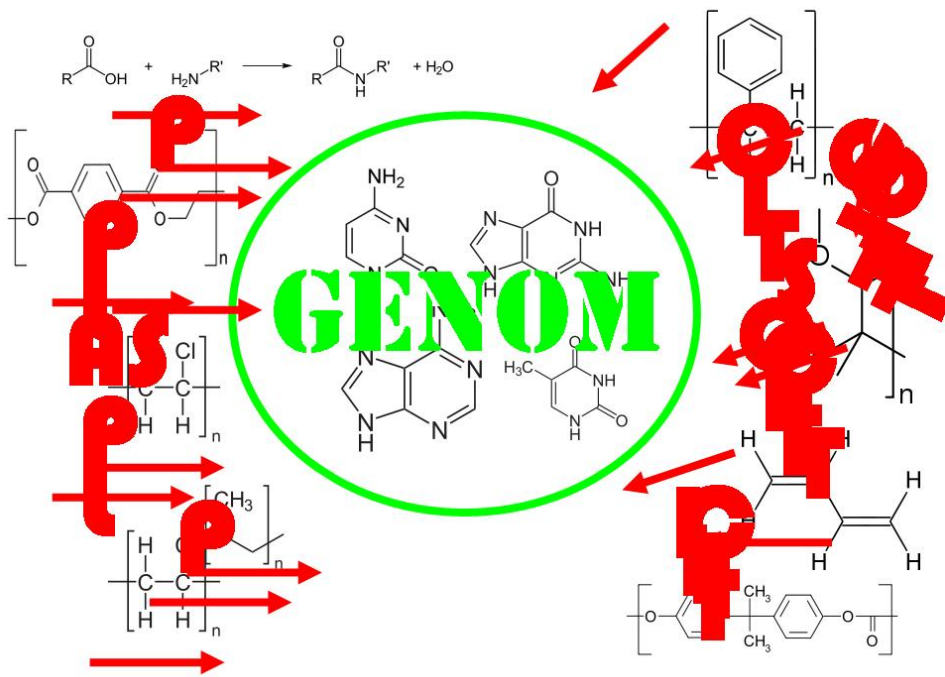
unter der Wasseroberfläche eine Dichte von 300.000 Plastik-Partikeln pro Quadratkilometer festgestellt, am Meeresboden lag sie bei 100.000 parts/qkm.⁸

Plastik wirkt im Meer wie ein Giftschwamm, der wasserunlösliche toxische Substanzen wie DDT oder PCB, die über Flüsse ins Meer gelangen, an seiner Oberfläche bindet. Die Konzentration dieser Substanzen an treibenden Plastikpartikeln ist um eine Million Mal höher als im umgebenden Wasser. Fische nehmen die ‚nahrungsergänzenden‘ Stoffe auf und setzen sie in der Nahrungskette fest, oder sterben direkt an Plastikverschlüssen und anderen Kunststoffteilen, die sie für Futter halten. Schätzungen gehen von jährlich mehr als einer Million Seevögeln und über 100.000 Meeressäugern aus, die auf diese Weise qualvoll verenden.

Neueste Forschungen ergaben im Weiteren, dass sich an bestimmten Stellen hoher Konzentration der mikroskopisch kleinen *plastic pellets*, die man auch "mermaids' tears" nennt, in den Weltmeeren ähnlich wie dies Evolutionisten¹⁰ für die Herausbildung lebender Organismen aus unbelebten Atomen und Molekülen annehmen, aus den Basinsubstanzen dieser *pellets* auf bisher noch ungeklärte Weise unter anderem die Monobloc-Stühle bilden und sich gezielt - in größeren Formationen aus dem Meer erhebend - invasiv-aggressiv in den Bereichen der menschlichen Kulturlandschaften niederlassen.

Bei Ausdehnung der Erforschung der „inneren Kräfte“ dieser invasiv militanten Eroberung des menschlichen Lebensraums durch die Stühle zeigte sich, dass **generell** in der Mikrosphäre des Planeten der brutale Kampf der Plastikstoffe

METATHEMEN



gegen das **menschliche Genom¹¹ voll entbrannt ist**. Die aggressive Molekularsubstanz der Stühle erwies sich lediglich als besonders evidenter Einzelfall der generellen Attacke des Plastiks auf die Biosphäre des Planeten.

Beispiele: In Wasserproben finden sich erschreckend hohe Konzentrationen von Weichmachern (überwiegend DEHP). Ausgangsstoff für diese Weichmacher sind die **Phthalate**. Phthalate bewirken eine Auf- und Abwärtsregulierung von Genen schon in kleinsten Mengen. Eines steht fest: Die Chemikalien sind bereits in unserem Körper. In einer Studie der U.S. Gesundheitsbehörde fanden sich in nahezu jedem Probanden beunruhigende Mengen an Bisphenol A im Urin. Besonders Kinder waren hiervon stark betroffen. Bisphenol A ist Bestandteil von Polycarbonat Kunststoff (Sonnenbrillen, CDs, Zahnfüllungen, Beschichtung von Dosen von innen). Das entscheidende zu Bisphenol A: Ähnlich wie beim natürlichen Hormon Östradiol reichen weniger als 1 Billionstel Gramm im Blut aus, um

körperliche Entwicklungen nachhaltig zu beeinflussen. Bisphenol A steht im direkten Zusammenhang mit Anomalien von Spermienentwicklung, Prostatakrebs, Gehirnentwicklung, Sozialverhalten, Hyperaktivität und Hyperaggressivität. Bisphenol A wurde ursprünglich als Östrogensatz entwickelt. Die Plastik-Eigenschaften wurden erst später entdeckt.

Lyrische(s) Plastik – Die Portraits

Der zweite Teil der Arbeit „Invasion“ besteht aus lyrisch-zarten Foto-Portraits von Monobloc-Stühlen, die bereits ohne die inhaltlichen Hintergründe allein durch ihre duftige Monochromie mit leichten Farbeinsparungen ein theoretisch unendliches Universum der mathematischen Variationsmöglichkeit eines formalen Themas andeuten und ermöglichen.

Die formale Variation des Themas „Stuhl“ erweist sich bei Einbeziehung der obigen Analyse als eine Veränderung des einzelnen Stuhls durch

exogene und **endogene** physikalische, chemische und im Weiteren biochemische Prozesse.

Die Portraits dokumentieren „äußere“ Verletzungen, Beschädigungen und Deformationen, welche der einzelne „Kämpfer“ in der invasiven Aggression nach seiner Entlassung aus dem Meer in seiner Mission "von außen" her erleidet.

Andere Läsionen werden aber in den neuesten Studien auf **endogene** Deviationen der Strukturformel der Plastiksubstanz im Stuhl zurück geführt, welche offensichtlich die Molekularebene aller Plastikprodukte in ihrem Kampf gegen die biomolekulare Genomstruktur der Lebewesen des Planeten durch Kontamination im offenen Kampf erleidet. Das Genom schlägt zurück! Die Stühle erkranken von innen her!

Wir sehen, die theoretischen Rahmen der eher epistemisch orientierten Konzept-Kunst Kosuths werden in Rupps Ansatz der Stuhl-Thematik weit überschritten und soziologisch ausgebaut.

Fußnoten

¹ <http://images.google.at/images?gbv=2&hl=de&sa=1&q=%22One+and+Three+Chairs%2BJoseph+Kosuth%22&btnG=Suche&aq=f&oq=&start=0> sowie auch *One and Four Chairs* unter <http://www.jacquelineheerema.nl/reflectiewonderkamer.htm> und weiters <http://www.architonic.com/de/dcsht-1-2-3-five-chairs-and-table-sotheby-s/4108852>.

² Kosuths konzeptuelle Positionen der Kunsterweiterung sind selbst inzwischen maßgeblich erweitert worden. Vgl. hierzu etwa <http://or-om.org/kunsterweiterung.doc> und <http://portal.or-om.org/art/ExtendedArt/tabid/6075/Default.aspx>. Rupps Einsatz des Stuhlmotivs erfolgt, wie sich zeigen wird, in völlig neuen und konzeptuell keineswegs von Kosuth irgendwie abhängigen Parametern.

³ <http://de.wikipedia.org/wiki/Polypropylen>

⁴ „Addicted to Plastic“ <http://www.bullfrogfilms.com/catalog/atp.html> und <http://www.youtube.com/watch?v=daSFXT-HYk> sowie: (siehe Tabelle rechts).

Weltweit werden im Jahr fast 240 Millionen Tonnen Kunststoffe aus rund „nur“ 4 Prozent der weltweiten Erdölproduktion hergestellt. „In Europa werden heute etwa 60 Millionen Tonnen Plastik produziert. das sind etwa 25% der globalen Zahlen.“ (John Taylor, Präsident von Plastics Europe, Zitat aus „Plastic Planet“.) Die Kunststoffindustrie macht 800 Milliarden Euro Umsatz pro Jahr. Allein in Europa verdienen mehr als eine Million Menschen ihr tägliches Brot in der Plastikindustrie. Jeder Industriezweig ist heute auf Kunststoff angewiesen (zitiert aus „Plastic Planet“.)

⁵ <http://www.openartresidency.gr/index-en.htm>.

⁶ http://www.algalita.org/charles_bio.html.

⁷ <http://www.algalita.org/> und http://www.alguita.com/movs/pelagic_plastic_mov.html.

⁸ <http://news.bbc.co.uk/2/hi/science/nature/6218698.stm>

⁹ <http://www.mindfully.org/Plastic/Ocean/Moore-Trashed-PacificNov03.htm>.

¹⁰ Abiogenese oder Chemische Evolution etwa unter http://de.wikipedia.org/wiki/Chemische_Evolution.

¹¹ Hinsichtlich der Struktur des Gencodes und seines Verhältnisses zu den „Grundbausteinen der Materie“, den Atomen hat die Gruppe Or-Om in "Gencode meets Or-Om-Code ∞ 1.0" <http://portal.or-om.org/science/GencodemeetsOrOmcode10/tabid/6374/Default.aspx> dargestellt, dass die modernen Ansätze der Physik und Biologie (Genstruktur) durch die Einführung einer universellen unendlichen Urkraft der Natur zu erweitern sind, in deren Inneren sich Partialkräfte und –teilchen (Vermittlerteilchen der Kräfte und Elementarteilchen) und im Weiteren Atome, Moleküle und schließlich Biomoleküle ausbilden. Alle diese Partialelemente befinden sich aber weiterhin in einer Unverschränkung mit der über ihnen bestehenden unendlichen und universellen Urkraft der Natur. Dieser Ansatz bildet auch evolutive Horizonte aus, um die derzeitigen Projekte und Aktivitäten der Umweltorganisationen zu vertiefen, zu erweitern und ihnen theoretisch höhere Sicherheit zu geben.

Recyclingsnummer	Kürzel	Name des Werkstoffs	Verwendung und Recycling des Polymers zu
	PET	Polyethylenterephthalat	Polyesterfasern, Folien, Softdrink-Flaschen
	PE-HD	High-Density Polyethylen	Plastikflaschen, Plastiktaschen, Abfalleimer, Plastikrohre, Kunstholz
	PVC	Polyvinylchlorid	Fensterrahmen, Rohre und Flaschen (für Chemikalien, Kleber, usw.)
	PE-LD	Low-Density Polyethylen	Plastiktaschen, Eimer, Seifenspenderflaschen, Plastiktuben
	PP	Polypropylen	Stoßstangen, Innenraumverkleidungen, Industriefasern
	PS	Polystyrol	Spielzeug, Blumentöpfe, Videokassetten, Aschenbecher, Koffer, Schaumpolystyrol
	OTHER	Andere Kunststoffe wie Acrylglas, Polycarbonat, Nylon, ABS und Fiberglas.	

Datalogger iGot-U GT-600 – Set and forget?

Paul Belci

In meinem letzten Bericht, habe ich einige Datalogger verglichen und da war der GT-200e dabei. Nun habe ich von Mobile Action das Nachfolgemodell zum Testen bekommen, den i-gotU GT-600.

Das neue Modell ist zwar baugleich, aber um eine nicht unwesentliche Funktion erweitert worden.

Einführung

Die neue Funktion des GT-600 ist eine automatische Abschaltfunktion, wenn der Logger nicht bewegt wird. Trivial, wird man glauben, aber weit gefehlt! Denn diese Funktion macht es möglich, den Datalogger wirklich intuitiv zu verwenden. Einfach auf den Wanderrucksack d'raufgeschmalt und das war's. Alles andere macht der Logger selbsttätig. Natürlich muss man ihn ab und zu aufladen ;-)

Beim GT-600 ist eine Silikon-Hülle dabei, an die man ein Klettband montieren kann. Mit dieser Ausstattung kann man den Logger nahezu überall befestigen. Entweder am Arm, am Handgelenk, am Tragegurt einer Tasche oder eines Rucksackes.

Technik

Größe:	46 x 41.5 x 14 mm
Gewicht:	37g
Empfänger:	SiRF Star III 65nm GPS low power chipset
Speicher:	64MB Flash-Memory, oder 262.000 Wegpunkte
Batterie:	LiOn Akku mit 750 mAh
Kaltstart:	35 Sekunden
Anschluss:	USB 1.1
Besonderheiten:	eingebauter Bewegungsschalter, wasserdicht

Der eingebaute Bewegungsschalter deaktiviert den Logger sobald er eine gewisse Zeit nicht mehr bewegt wurde und spart somit Akku. So kommt der Logger auf sehr lange Akkulaufzeit.

Wichtig ist, den Datalogger beim ersten Mal erst draußen einzuschalten, denn dann ist es für das Gerät leichter, einen brauchbaren Empfang für einen ersten Fix zu finden. Danach ist es nicht mehr so wichtig, denn der Empfänger merkt sich den Ort des letzten gültigen Signals und sucht von dort aus wieder nach einem Fix.

Schwierig wird es nur, wenn man Teile des Weges ohne Empfang zurücklegt, zum Beispiel in der U-Bahn. In diesem Fall wird der Empfänger einige Zeit im Freien brauchen, bis er wieder Empfang hat und auch einen Fix bekommt. Diese Einschränkung trifft allerdings alle GPS Logger egal welchen Chipsatz sie haben.

Und da kommen wir schon zu einem wichtigen Punkt der speziell Datalogger von i-gotU betrifft. Sie müssen aufgrund des verbauten SiRF III Chipsatzes immer gute Sicht zum Himmel haben, da der SiRF Chipsatz im abgeschirmten Zustand nur schlecht funktioniert!

Aber das sollte sich beim Fototaggen, oder Wandern ja machen lassen ...

Zusätzlich ist der Logger auch wasserdicht. Dadurch kann man ihn auch bei feuchten Aktivitä-

ten verwenden, beispielsweise beim Wassersport, oder Schifahren.

Das Highlight des GT-600 ist die Software - @Trip

Der beste Logger hilft nix, wenn die mitgelieferte Software nicht brauchbar ist. Da kann der Logger von MobileAction richtig gut punkten!

Das mitgelieferte Programm beginnt schon bei der Installation, einen professionellen Eindruck zu machen. Es installiert die Software, danach wird man aufgefordert, den GT-600 anzusteuern. Anschließend wird automatisch der Treiber installiert. Am Ende der Installation wird noch abgeprüft, ob das richtige DotNet-Framework auf dem Rechner ist und auch dieses installiert, wenn nötig.

Startet man die Software, kann man den Logger sofort automatisch auslesen lassen. Natürlich ist es auch möglich, Daten von anderen Geräten zu importieren, wenn sie im *.gpx oder im *.gpk Format verfügbar sind. Danach kann man sich die Routen ansehen, sie bearbeiten oder gleich



auf dem Webspaces von Mobile Action der Öffentlichkeit präsentieren.

Anfangs ist die Software etwas gewöhnungsbedürftig, aber man findet sich danach gut zu recht. Einige der vielen Funktionen offenbaren erst nach einer kurzen Einarbeitungszeit ihre volle Funktionalität.

Der Tipp für „Foto-Tagging“

Zuerst muss man einmal sicherstellen, dass die Zeit der Kamera korrekt eingestellt ist, sonst können die Fotos nach der Aktion nicht automatisch richtig zugeordnet werden.

Ist man im Freien, sollte man stehen bleiben, bis der Datalogger auch einen FIX hat. Das zeigt der GT-600 an, indem die rote und die blaue LED gleichzeitig aufblinken.

Nun kann es eigentlich schon losgehen. Man sollte natürlich beim Foto-Taggen darauf aufpassen, dass der Logger immer auch in der Nähe der Kamera ist, denn sonst stimmt die Position nicht mit dem Ort des Fotos überein. Eh klar werdet Ihr sagen, aber in der Praxis bleibt der Rucksack schon mal auf einer Parkbank liegen

während man losrennt, einen seltenen Vogel zu fotografieren ;-)

Ist die Fotowanderung abgeschlossen, geht es ans Zusammenstellen der Aufzeichnungen.

Daten aus dem Logger auslesen, Fotos in einen Ordner spielen und diesen beim Datenauslesen angeben. Fertig! Sofern die Zeit synchron ist, werden die Fotos bereits auf der Route mit angezeigt.

Stimmt die Zeit nicht mit den Geodaten überein, kann man anhand eines Fotos, von dem man die korrekte Position weiß, alle Fotos aus dem Ordner automatisch richtig zuordnen lassen. Dazu öffnet man im Programm die Eigenschaften des Fotos und setzt dort die Zeit richtig. Alle anderen Fotos werden dann auf Knopfdruck passend zugeordnet!

Außerdem ist die Software in der Lage, die Geodaten gleich nach der richtigen Positionsbestimmung in die entsprechenden EXIF-Felder der JPG-Dateien einzutragen. Das finde ich wirklich toll!

Online Präsentation der Routen

Hat man eine Route fertig bearbeitet, kann man sie auf dem Server von Mobile Action online stellen.

Entweder man veröffentlicht seine Route mit einem Schlüssel, damit nur bestimmte Personen diese Route sehen können, oder einfach als „öffentlich“ für Alle.

Die Routen werden auf dem a-trip Server mit einem Nummernkürzel abgelegt und können dort dann angesehen, oder nachgefahren werden

Ich habe euch eine Beispielroute von einer Segwaytour des letzten Sommers zum Ansehen abgelegt: <http://www.a-trip.com/tracks/view/42937>

Natürlich lassen sich auch gleich Fotos dazugeben, wenn man das möchte. Auch hier habe ich ein Beispiel von einem Rundflug über Salzburg für Euch: <http://www.a-trip.com/tracks/view/42948>

Allerdings hatte ich hier die Zeit der Kamera nicht korrekt eingestellt und musste daher einige Fotos händisch positionieren!

Mit dem Programm ist es auch möglich, Fotos nach Picasa oder Flickr hochzuladen. Da ich diese Dienste nicht verwende, habe ich das allerdings nicht ausprobiert.

Fazit - Wann ist der GT-600 empfehlenswert?

Der i-gotU GT-600 von Mobile Action ist durch Verwendung des SiRF Star III Chips sicher nicht der empfangsstärkste Empfänger, zeichnet sich aber trotzdem durch einige Besonderheiten aus.

- ☺ Leistungsfähige Software
- ☺ Möglichkeit zur Onlinepräsentation von Tracks
- ☺ Automatische Abschaltung
- ☺ wasserdicht
- ☺ „nur“ SiRF Star III Chipset (braucht immer gute Sicht zum Satelliten!)

Der Logger ist natürlich auch als Bluetooth Empfänger für PDAs und Notebooks verwendbar. Er kann sogar während des Betriebes als Bluetooth Empfänger Daten mitloggen. Das können auch nicht alle Empfänger...

Ich habe den GT-600 die meiste Zeit auf meinem Rucksack montiert und verwende ihn als Datenlogger für alle Aktivitäten wo der beste Empfang nicht so wichtig ist. Beispielsweise bei einer Segwaytour ist er ideal. Ich befestige ihn am Lenker und da macht es auch nix wenn's mal regnet, denn der GT-600 ist ja wasserdicht.

Für alle, die einen Datalogger für die Hosentäschche, oder fürs Auto suchen, ist der GT-600 nicht geeignet, denn sobald der Empfänger etwas abgeschirmt ist, lässt auch der Empfang sehr zu wünschen übrig...

Grundsätzlich ist schon die Software den Preis des GT-600 wert, denn diese ist nach einer entsprechenden Einarbeitungszeit sehr brauchbar!

Flug Start



Deuer: 1 Stunde 32 Min 32 Sek
Gesamte Entfernung: 244.344 km
Mittlere Geschwindigkeit: 158 km/h
Maximale Geschwindigkeit: 226 km/h
Größe: 2909 m

Höhenlage vs. Entfernung: [Line graph showing altitude over distance]

Rundflug

Name	Datum
✓ Segwaytour mit Boris	2009-07-08
Rundflug mit Roland	2009-08-30
Hochzeit Tom	2009-09-09
20100111 - Fussm...	2010-01-11

@ Trip Software

Datenbanken in Outlook 2010 synchronisieren

Paul Belcl

Mit dem Tool „Outlook Sync DB“ ist es möglich, nahezu jede Datenquelle in Outlook zu synchronisieren.

Warum braucht man das ?

Hat man, wie ich, eine sehr umfangreiche Adressdatenbank mit vielen besonderen Funktionen, dann kommt immer wieder der Wunsch auf, diese gut gepflegten Daten auch in Outlook zu verwenden. Für all diese Zwecke ist Outlook Sync DB von Andreas Schultz eine gute Lösung.

Was kann Outlook Sync DB?

Die Software kann nahezu jedes gängige Datenbank Format auf Datensatzebene mit Outlook synchronisieren. Man kann viele Optionen zur Synchronisation auswählen. Beispielsweise, ob die Datenbank als ODBC-Datenquelle vorhanden ist, oder in einer .mdb Datei.

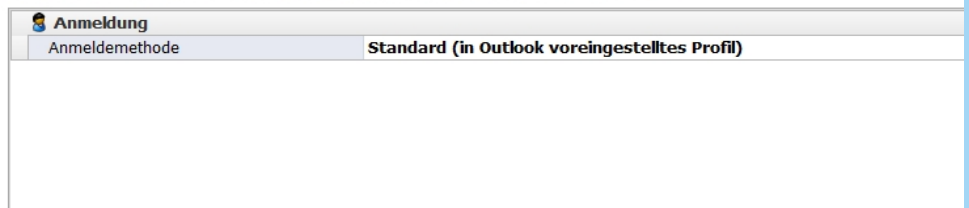
Genau so kann man auswählen, ob man mit einem Outlook Postfach (.pst-Datei) synchronisieren will, oder mit einem Exchange Postfach. Auch kann man das Standard-Profil von Outlook verwenden ohne Rücksicht auf die Anbindung dahinter nehmen zu müssen.

Beim Synchronisieren lassen sich Filter setzen, oder SQL-Statements verwenden um die Datensätze zu selektieren die man synchronisieren will. In meinem Fall konnten sogar Bilder, die in der Access Datenbank als Dateipfad verknüpft waren, in Outlook übernommen werden.

Die Synchronisation funktioniert allerdings „nur“ auf Datensatzebene. Das bedeutet, es

Outlook / Exchange Verbindung

Wählen Sie die Anmeldemethode aus und geben Sie die Parameter für die Verbindung an.



Wählen Sie den Outlook Ordner aus, optional können Sie einen Filter definieren.



Abb 01: Outlook Einstellungen

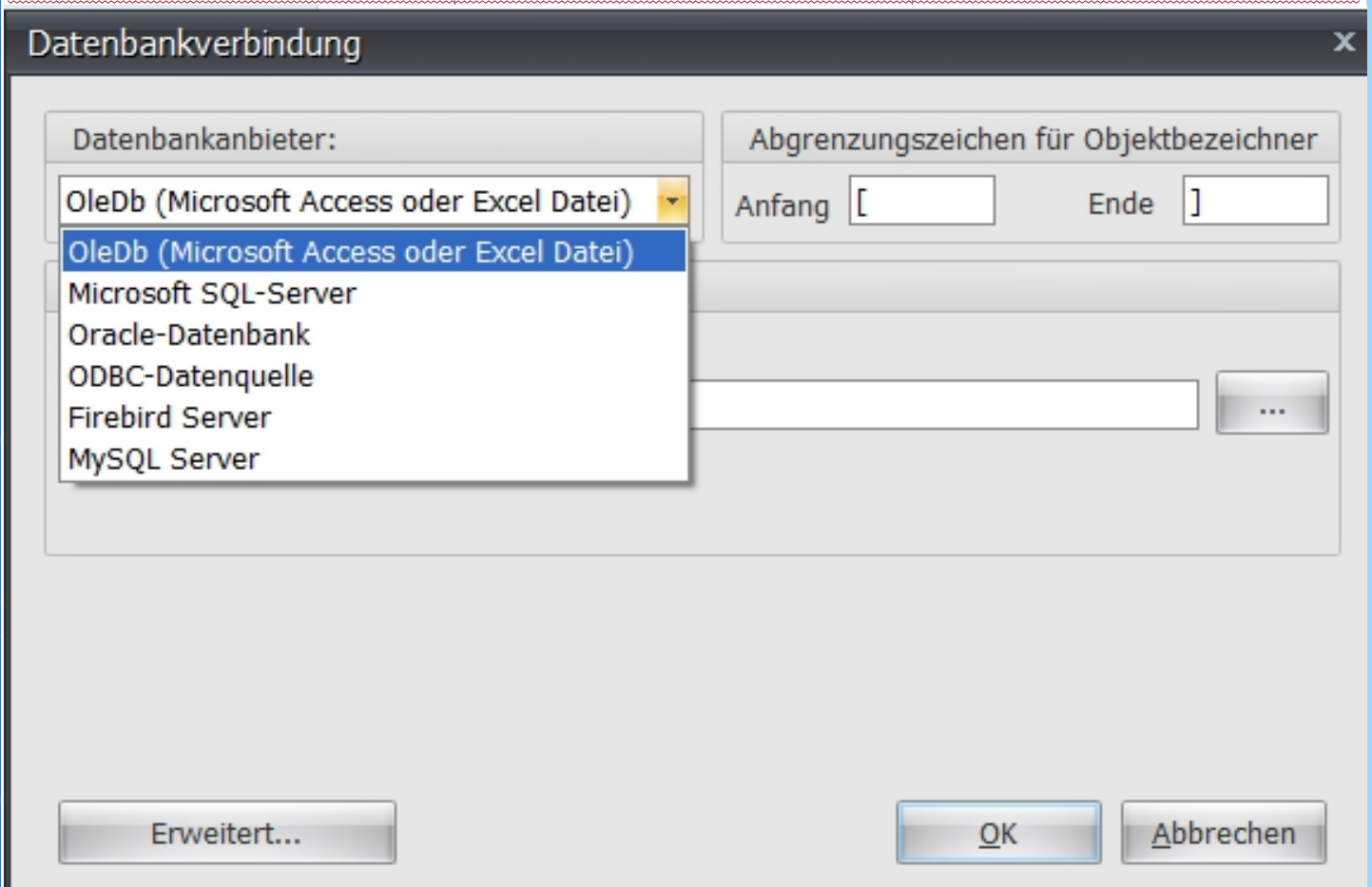
wird nur ein kompletter Kontakt einer Person synchronisiert und nicht einzelne Felder.

Das birgt zwar eine kleine Stolperfalle, denn wenn man an *einem* Kontakt auf beiden Seiten ein anderes Feld ändert und danach synchronisiert, dann wird die zuletzt gemachte Änderung durchgeführt und die Änderung vorher in diesem einen Kontakt überschrieben. Warum das so ist, kann einfach erklärt werden.

Datenbank Synchronisation - hinter den Kulissen

Für eine Synchronisation auf Feldebene, müsste der Synchronisationsstatus nahezu die doppelte Größe der Datenbank haben. Da ja der letzte Wert jedes Feldes von Outlook und von der Datenbank im Synchronisationsstatus gespeichert werden müsste. Ansonsten ließe sich ja nicht ermitteln, welches Feld sich seit der letzten Synchronisation geändert hat.

Abb02: Datenbank-Auswahl



Das würde bei großen Datenbanken nicht nur viel Platz verbrauchen, sondern auch sehr lange dauern. Daher ist es besonders für „große“ Anwendungen von Vorteil, die Daten auf Datensatzebene zu behandeln.

Outlook Sync Db speichert lediglich einen Hashwert über alle Felder pro Datensatz im Synchronisationsstatus. Das hat zwar den Nachteil, dass nur immer ein Kontakt mit allen Feldern behandelt werden kann, dafür bekommt man den Vorteil, dass die Speichergröße für den Synchronisationsstatus klein ist und die Synchronisation wesentlich schneller abläuft. Damit lassen sich Datenbanken mit mehreren Zehntausend Kontakten in brauchbarer Zeit synchronisieren.

Outlook Sync Db ist meines Wissens die einzige Lösung auf dem Markt, die überhaupt eine „echte“ Synchronisation in beide Richtungen ermöglicht.

Praxistest

Ich habe die Software getestet und damit die Daten meiner Adressen-Datenbank in Outlook 2010 synchronisiert.

Das Programm läuft komplett eigenständig und braucht daher Outlook nicht einmal installiert, wenn man beispielsweise Daten mit einem Exchange Postfach synchronisieren möchte.

Natürlich lässt sich auch die Richtung der Datensynchronisation auswählen, in meinem Fall wird nur von der Adressdatenbank in Outlook synchronisiert, da die Access Datenbank nach wie vor meine „Basis“ ist.

Nachdem festgelegt ist, welche Art von Datenbasis synchronisiert werden soll, kann das Ziel ausgewählt werden. In meinem Fall ist das ein Exchange Postfach, welches ich über ein Outlook Profil einbinde. Nach der Auswahl der zu synchronisierenden Datenquelle kommt die Auswahl der Felder dran. **(Abb02, Abb03)**.

Rechts sind alle in Outlook verfügbaren Felder angeführt, rechts die Felder der Datenbank. Ist die Feldauswahl beendet, kann man sie auch für spätere Verwendung als Profil abspeichern, somit muss man sich diese mühsame Arbeit nur einmal antun!

Danach legt man noch fest, ob ein Filterkriterium zum Synchronisieren benötigt wird **(Abb04)** und ob man nur in eine Richtung synchronisiert, oder in beide Richtungen **(Abb05)**.

Ist alles fertig eingerichtet, dann gibt es noch eine Zusammenfassung der eingestellten Felder. Hier kann man nochmal prüfen, ob alle Felder richtig verbunden sind und ob alle Einstellungen passen.

Danach kann die Synchronisation schon starten.

Fazit

OutlookSyncDb ist ein sehr flexibles Werkzeug, nahezu jede Datenquelle in Outlook zu bekommen. Zwar ist dies „nur“ auf Datensatz-Ebene möglich, aber in vielen Fällen reicht das aus, um mit wenig Aufwand ans Ziel zu gelangen.

Die Software funktionierte in meinem Test stabil. Toll fand ich, dass Andreas Schultz für meine Fragen immer offen und hilfsbereit war. Einige kleine Dinge die ich mir noch wünschte, wurden von ihm auf die *ToDo*-Liste gesetzt, da er sie auch für sinnvoll erachtete. Wirklich vorbildlich...

Bezugsquelle

<http://www.schultz-it.de/Products/OutlookSyncDb.aspx>

Standardfelder Verknüpfungen

Definieren Sie die Feldverknüpfungen für die Standardfelder.
Hinweis: Mit der Tastenkombination [Strg] + [Entf] können Sie die markierte Zuordnung löschen.

Outlook Feld	Datenbank Feld
Kategorie: Adressfelder	
Adresse geschäftlich [String]	
Adresse geschäftlich: Bundesland/Kanton [String]	
Adresse geschäftlich: Land/Region [String]	
Adresse geschäftlich: Nur Hausnummer [String]	
Adresse geschäftlich: Nur Straße [String]	
Adresse geschäftlich: Ort [String]	
Adresse geschäftlich: PLZ [String]	
Adresse geschäftlich: Postfach [String]	
Adresse geschäftlich: Straße [String]	
Adresse privat [String]	
Adresse privat: Bundesland/Kanton [String]	
Adresse privat: Land/Region [String]	Land
Adresse privat: Nur Hausnummer [String]	
Adresse privat: Nur Straße [String]	
Adresse privat: Ort [String]	Ort
Adresse privat: PLZ [String]	PLZ
Adresse privat: Postfach [String]	
Adresse privat: Straße [String]	Adresse
Art der Postadresse [Integer (0-3)]	

Abb03: Feldauswahl

Datenbankverbindung

Klicken Sie auf die Schaltfläche, um die Datenbankverbindung zu definieren.

Ausgewählte Datenbankverbindung

Verbindungsoptionen

Vorbereitungsskript (SQL)

angepasste Abgrenzungszeichen nutzen

Wählen Sie die Datentabelle bzw. Sicht aus. Alternativ können Sie auch benutzerdefinierte SQL-Befehle nutzen.

Allgemein

Benutzerdefinierte SQL Befehle nutzen

Tabelle bzw. Sicht

Name **ADRESSEN**

Filter **[arc]=false**

Abb04: Filter

Synchronisationsoptionen

Geben Sie die Parameter für die Synchronisation an.

Analyseoptionen

ICS für Analyse nutzen

Synchronisationsoptionen

Synchronisationsrichtung **Von Datenbank zu Outlook**

Änderungen am Ziel überschreiben

Einfügungen am Ziel beibehalten

Abb05: Synchronisationsrichtung



Mobiles Wertkarteninternet

Ein kurzer Überblick

Christian Schneider

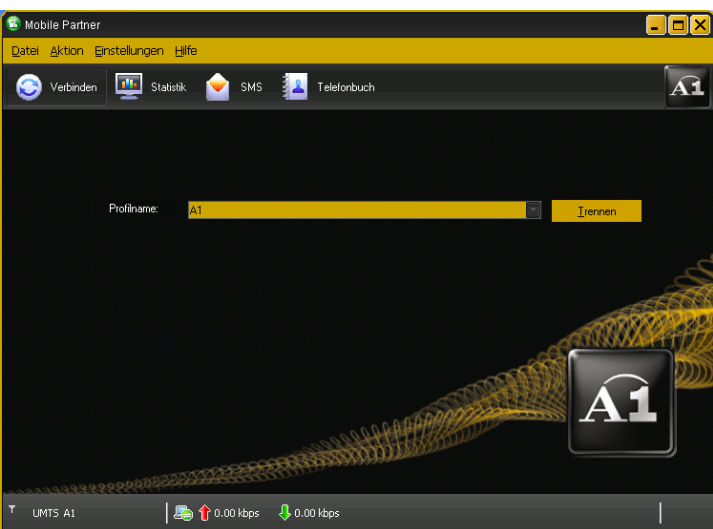
Als Ergänzung zum fixen Internetanschluss ist mobiles Internet äußerst praktisch. Doch es muss nicht immer mit Vertragsoption sein. Wer nicht zu viel Traffic verursacht, ist mit Wertkarteninternet gut bedient. Ich habe fixes Internet von Inode www.inode.at zu Hause und mobiles Internet von Telering www.telering.at mit Vertrag: 10 €pro Monat für 6 GB Datentransfer.

Die Preise der Wertkarten-Sticks sind von allen Betreibern fast gleich. Man zahlt für das Startpaket, das in der Regel gleich 2 GB Guthaben beinhaltet, ca. 55-60 € 1 GB Guthaben kann um 20 €aufgeladen werden. Nur Ge-org www.ge-org.at ist billiger, hier kostet es die Hälfte. Ge-org nutzt die Telering-Infrastruktur. Als eines meiner Hobbies habe ich mir Wertkarteninternet-Sticks von allen Netzbetreibern zugelegt.

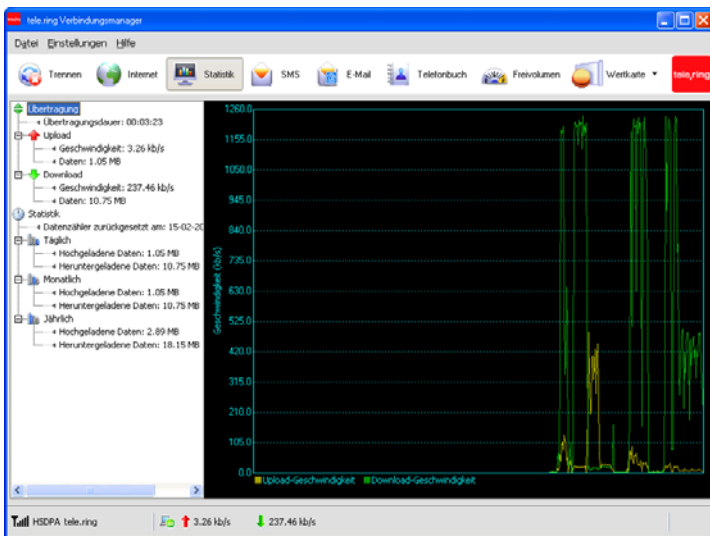


Links sieht man den Stick von A1, den ich auf der Post gekauft habe, www.a1.net – auch Bob www.bob.at nutzt das gleiche Netz. Dann folgt Telering; diesen Stick habe ich von meiner Nachbarin geschenkt bekommen. Auch T-Mobile www.t-mobile.at und Ge-org www.ge-org.at haben dieses Netz. Der dritte Stick ist von Drei www.drei.at – diesen habe ich im Internet bestellt, und er wurde nach zwei Tagen geliefert. Rechts ist der Stick von Yesss www.yesss.at, welches das Orange-Netz www.orange.at benützt. Diesen habe ich auch im Internet bestellt, jedoch erst nach sechs Tagen geliefert bekommen.

Die Installation der Modems ist denkbar einfach: Man steckt es an den USB-Anschluss an, und die Software wird automatisch installiert. Es lohnt sich aber, auf der Homepage des jeweiligen Netzbetreibers nach Aktualisierungen zu suchen, ich habe mir bei allen meinen Sticks außer Yess die neueste Software herunter geladen. In diesem Fall installiert man zuerst die heruntergeladenen Programme, und erst dann steckt man das Modem nach einem Neustart ein.



Dies ist der Mobile Partner von A1. Er verfügt auch über ausführliche Statistiken. Links unten wird die Signalstärke angezeigt und ob man UMTS, Edge usw. verwendet.



Hier haben wir den Telering Verbindungsmanager. Auch hier wird die Signalstärke links unten angezeigt sowie die Upload- und Download-Geschwindigkeit. Der Knopf für Verbinden und Trennen ist links oben. Das Telering Wertkartenmodem ist auch auf meinem Asus Eee-PC unter Xandros-Linux ohne Installation einwandfrei lauffähig, die anderen leider nur unter Windows und Mac.



Im Vergleich dazu ist der Internet-Manager, der bei meinem ZTE-Modem mit Vertrag dabei war, schlicht gestaltet. Man trennt bzw. verbindet, indem man auf das Männchen rechts unten klickt.



Das ist das Drei-Zugangsprogramm, welches am einfachsten zu bedienen ist, dank des großen Knopfes in der Mitte. Die Statistiken sind allerdings nicht sehr ausführlich. Als besonderes „Schmankerl“ kann man gratis, ohne dass es vom Guthaben abgebucht wird, Fernsehen am Laptop genießen, indem man auf 3MobileTV klickt, und das zwei Monate lang seit der letzten Aufladung:



Leider machte Yess bei der Installation Probleme, alle anderen Sticks funktionieren parallel auf einem Gerät ohne Schwierigkeiten. Yess benutzt auch das Programm *Mobile Partner*, und beim Anstecken wollte es meinen A1-Anschluss deinstallieren. Der Workaround lautete, dass ich den A1 *Mobile Partner* für Yess in einem neuen Profil konfigurierte, was sehr einfach war:

Hier sieht man links unten auf dem Desktop die Verknüpfungen meiner Internet-Einwahlprogramme. Als Wermutstropfen bleibt, dass man bei jedem Anstecken des weißen Yess-Sticks die Installation abrechnen muss.

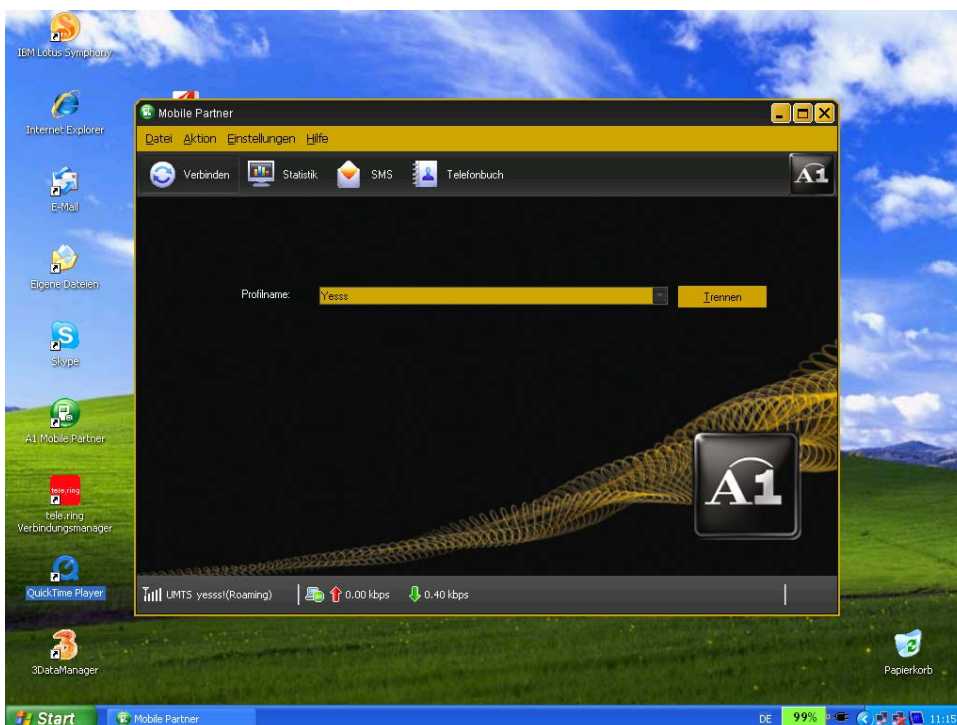
Kontostand und Aufladung sind bei allen Netzbetreibern außer Telering webbasiert und über den Browser sehr einfach zu managen. Im Allgemeinen kann man mit Wertkarte, Internet-Banking oder Kreditkarte aufladen, nur bei Telering erfolgt die Verwaltung über die Zugangssoftware, und die Aufladung ist nur über Wertkarte im Manager möglich.

Alle Zugangsprogramme außer Telering (und natürlich Yess) starten automatisch beim Anstecken des Modems, Telering muss manuell gestartet werden, nachdem das Modem blau blinkt.

Bei der Wahl des günstigsten Betreibers muss man darauf achten, ob auch an den gewünschten Orten genügend Signalstärke vorhanden ist. Am besten testet man das vorab mit Bekannten. Gerade außerhalb der Ballungszentren ist nicht gesagt, dass Empfang vorhanden ist, und die Netzabdeckungskarten der Netzbetreiber online sind etwas optimistisch gestaltet. In Ballungszentren ist oft die Geschwindigkeit reduziert, weil viele an einem Umsetzer hängen und Traffic verursachen. So funktioniert bei mir das Drei Gratis-TV nicht immer. Im Allgemeinen ist die Anbindung bei allen für Skype und Youtube ausreichend.

Die empfehlenswerte Webseite www.umtslink.at hat ein gutes Forum für alle Fragen zu bieten. Geschwindigkeitstests kann man auf www.speed.io und www.speedtest.net machen, und auf www.senderkaster.at sieht man die Standorte der Sender auf einer Karte nach Eingabe des gewünschten Ortes (Postleitzahl). Leider sind hier die Netzbetreiber aus wettbewerbsrechtlichen Gründen nicht aufgeführt. Zeigt man mit der Maus auf einen Sender, werden Informationen eingeblendet.

Ich wünsche frohes Surfen und Mailen. Bei Fragen können Sie mich gerne kontaktieren.





Linux für Umsteiger

Günter Hartl

Mit diesem Beitrag möchte ich, Linuxanfängern oder -umsteigern grundsätzliche Dinge in und über Linux näherbringen. Ausserdem streife ich das allgegenwärtige Thema Sicherheit und Windows ein bisschen.

Warum ich das schreibe? Weil es immer dieselben Fragen sind und ich mir damit eine Menge Arbeit erspare. Die Tipps beziehen sich hauptsächlich auf Vector-Linux. (Nebenbei, es gibt über 800 verschiedene Linuxvarianten.)

Warum gerade dieses Linux?

Es ist klein, schnell, ausgereift...sehr stabil und läuft auch auf älteren Rechnern sehr schnell (ab 128 MB RAM). Ausserdem gibt es eine gute Softwareauswahl.

Der Nachteil: manche Programme muss man eindeutschen, da Vector-Linux ursprünglich aus Kanada stammt. Was soll's, das bisschen Englisch werden wir auch überleben. Es steht ja auch auf jedem DVD-Player (*play ... Record ... forward*)

Also, fangen wir an.

Warum brauch i a Linux?

Keiner braucht Linux, keiner braucht Windows. Wir brauchen nur einen vollen Eiskasten und a bissal Schlaf. Wenn Du mit Windows zufrieden bist, bleib dabei. Ich kenn' genug Spieler, die es seit Jahren verwenden. Wenn Du unbedingt Windows-Spiele spielen willst, musst Du es sowieso verwenden. In Linux gibt's auch Spiele, genug sogar. Viele Leute kennen halt nix außer Windows, deshalb glauben sie, dass jeder PC wie Windows funktioniert. Woher sollen sie's auch wissen?

Wo sind die Vorteile gegenüber Windows?

- Stabilität: Okay, Windows XP ist auch schon sehr stabil geworden (zumindest nach servicepack 2)...:-), außer ein Treiber verursacht wieder einmal einen Bluescreen.
- Transparenz (bei Fehlersuche hilfreich)
- Läuft auch auf älteren Rechnern performant (auf neuen sowieso).
- Extrem userfreundlich (nicht zu verwechseln mit idiotenfreundlich)
- Virenresistent (auf einer Workstation keine Securitysoftware erforderlich)
- ausgereift (strikte Benutzertrennung, User darf gar nichts am System machen)
- Ressourcenschonend (ab 128 MB Ram bist Du dabei)
- gängige Software ist automatisch mit installiert (Office, CD/DVD-brennen, Video, Pdf, Zip...)
- geringer Administrationsaufwand (Updates mit einem Mausklick)
- Große Community (Hilfe gibt es genug im "Netz")
- gratis und legal (Software wird über das Internet downgeladet)

Zum Thema "gratis"

Ja, Linux ist in der Regel gratis. Vom Internet runterladen, installieren, konfigurieren, fertig.

Nachteile von Linux

- Vor Hardwarekauf sollte man sich informieren...(Usb-Modem, Drucker...)
- Für die Administration sollte man ein bisschen Englisch können (Konsolenbefehle...)
- Extrem unstabil (okay...der war gut :-)

Für Windows hab' ich doch auch nix gezahlt!

Du zahlst immer für Windows; nur ist es für den Laien auf den ersten Blick nicht ersichtlich..

Mediamarkt, Superangebot, Komplett-PC mit Windows (XP, Vista, Windows7, was auch immer) vorinstalliert, "nur 500 Euro". Das heißt nichts anderes, als dass die Hardware um 350 Euro oder weniger verkauft wird und den Rest (in diesem fiktiven Beispiel 150 Euro) streift Microsoft ein. Üblicherweise hast Du bei solchen Angeboten MS-Office und einen Virenschanner vorinstalliert; super, zumindest für 60 Tage. Danach musst Du es kaufen oder die Software geht danach nicht mehr (wird nach 60 Tagen deaktiviert). Und wenn man die Leute erst einmal zwei Monate an was gewöhnt hat, ist es um so leichter, die paar Hundert Euro auch noch zusätzlich zu zahlen.

Ungefähre Preise...

- XP 100 Euro
- Windows7 ca. 140 Euro
- MS Office ca. 300 Euro (mit Outlook)
- Office 2007 Standard ca. 400 Euro, Professional ab 480,-
- Virenschanner ca. 60 Euro jährlich
- Brennsoftware ca. 70 Euro
- Photoshop 300 Euro ? (keine Ahnung jetzt..)

Wohlgemerkt, die Preise gelten immer nur für eine Lizenz (sprich einen PC)!

Natürlich ist es gängige Praxis, sich mal eben vom Nachbarn sich CD auszuborgen und irgendwas zu installieren.

Unterschätz' Microsoft nicht!

Keiner wird Dich jetzt deswegen verhaften (es ist nach wie vor illegal, aber da müssten sie halb Österreich verhaften); aber Du handelst Dir mit dieser Zugangsweise auf lange Sicht nur Ärger ein. Glaub mir, ich seh' das jeden Tag. Keine Frage, dass das Unrechtsbewusstsein in dieser Hinsicht im privaten Bereich nicht sehr ausgeprägt ist. (*bin I blöd, dass I ma des kauf, des is ja a Wahnsinn*). Das sind dann auch die Leute, die sich als erste über MS beschweren. Naja...

Sicherheitsupdates

Microsoft erkennt gecrackte Betriebssysteme (oder baut Mechanismen zur Erkennung ein) und schliesst diese von den Updates aus.

Du brauchst eh keine Updates?

Blaster-, Conficker-, I love you-Virus sind nur ein paar kleine Beispiele. Frag' einmal Deinen Administrator in der Firma nach diesen Begriffen! (hüstel)

Freeware-Virenschanner?

Des is a guada Virenschanner! Merke: es gibt keine guten Virenschanner, nur gute Virenschreiber :-)) Einmal ist der eine Scanner im Test besser, dann der andere; egal, in Windows brauchst Du einen auf jeden Fall; und eine Firewall (ist üblicherweise eh schon drin).

Warum gibts so viele Viren?

Weil die alle Geld bringen oder bringen sollten. Egal, ob Spam, Viren, Trojaner oder Würmer, der Endzweck einer solchen Schadsoftware ist immer, Geld zu machen; *simple as that*. Keiner ist wirklich an Deinen Urlaubsfotos oder an dem Liebesbrief auf Deiner Festplatte interessiert. Aber Deine Passwörter, Deine Hardwareressourcen (Internetleitung, CPU, Festplattenplatz, Adressbuch..) sind bares Geld wert.

Viele Leute glauben, weil Windows so verbreitet ist, gibts so viele Viren. Aber das ist nur die halbe Wahrheit. Weil die Chancen für einen erfolgreichen Angriff so hoch sind, trifft's schon eher. In Windows brauchst Du üblicherweise nur eine Sicherheitsbarriere zu überwinden, und Du hast gewonnen. Darum versuchen sich ja auch so viele daran.

Aber was wollen die bei mir?

Hauptsächlich Deinen PC, um andere PCs zu übernehmen, Dir dauernd Werbung einzublenden, Spam von Deiner Kiste aus zu verschicken oder Firmen über Deinen PC anzugreifen. Da gibt's unzählige Szenarien.

Wieso gibts für Linux keine Viren?

Gibt's schon aber nur im Labor und nicht in freier Wildbahn; bis jetzt zumindest nicht.

Es gibt über 800 verschiedenen Varianten von Linux und jede funktioniert ein bisschen anders. Und das bedeutet sehr viel Aufwand für einen Virenschreiber. Aber Windows in Schweden, St. Pölten oder Uganda funktioniert immer gleich und darum braucht man immer nur eine Variante schreiben und erwischt 800 Millionen Rechner; wenn's gut geht!

Selbst wenn jetzt ein Virus für Linux geschrieben werden würde, wäre binnen Tagen, wenn nicht Stunden ein Gegenmittel (Patch) vorhanden, da Linux keiner Geschäftspolitik unterliegt wie Microsoft.

Jeden zweiten Dienstag im Monat ist Patchday für Microsoft. Da kommen die Patches 'raus. Willst Du im schlechtesten fall einen Monat warten?

Egal, vergiss das Virenthema in Linux; das ist echt angenehm. Und auf älteren Rechnern merkst Du es ganz gut, wenn Du nicht diese aufgeblasenen Norton-Security-was-weiß-ich immer mitschleppen musst.

Hast Du schon einmal beobachtet, wie lang das Ding beim Start zum Laden braucht?

Zum Thema ressourcenschonend...

Okay, Windows XP läuft auf älterer Hardware auch ziemlich flott - zumindest am Anfang. Ich sag' immer: *die Zeit ist der beste Richter:-)* Solange Du nicht ins Internet gehst und nix installierst, ist Windows XP ein Super-Betriebssystem. Wenn's nach 5 Jahren noch



immer so laufen würde, wäre die Welt in Ordnung. Wir alle wissen, dass das nicht stimmt.

Linux hingegen läuft nach 5 Jahren noch genauso schnell (oder langsam) wie am Anfang. Linux installierst Du einmal, und dann komm' ich wieder, wenn die Hardware stirbt (Festplatten- oder Speicherdefekt, was weiß ich; ansonsten seh' ich die Leute jahrelang nicht - außer ich will sie sehen:-))

XP wird noch bis 2014 unterstützt, danach ist Schluss. Natürlich will Windows, dass Du eine neue Version (Windows7) kaufst, und die Firmen müssen da auf jeden Fall mitziehen. Spätestens 2014. Die "Schlaumeier" oder "Sparmeister" in den Firmen haben eh voriges Jahr die Rechnung präsentiert bekommen. Eine der Hauptverbreitungsquellen für den "Conficker-Virus" (ja ich weiss, tolles Wort) waren die vielen NT4-Firmennetze in Übersee. NT-4 wird schon seit Jahren nicht mehr supportet. NT-4 funktioniert, ja. Aber doch bitte nicht für Produktivumgebungen. Und schon gar nicht mit Internetzugang. Was machst Du aber mit Deinem 6 Jahre alten PC mit 512 MB Arbeitsspeicher? Windows 7? Vergiss' es!

Übrigens, auch einige aktuelle Linux-Distributionen (mandriva, suse, ubuntu...) laufen mit 1 GB RAM erst richtig rund. Aber es gibt genug andere (Vectorlinux, Slackware, Zenwalk, Sidux, Debian, Arch, Puppy...), die aktuell sind und auf 10 Jahre alter Hardware super laufen. Welches aktuelle Windows ist für ältere Hardware gemacht?

Muss ich das alte Zeug jetzt wegschmeissen, nur weil Windows es nicht mehr unterstützt?

In Windows gilt: *ich hab' dieses Betriebssystem, also brauch' ich diese Hardware*. In Linux heißt es: *ich habe diese Hardware, also nehm' ich dieses Linux*. Du glaubst ja gar nicht, wie viele 10 Jahre alten Server unter Linux noch immer laufen.

Windows7 soll aber super sein.

Alles, was nach Vista kommt, kann nur besser werden. Okay, das was gemeint:-) Natürlich läuft Windows 7 auf aktueller Hardware (Quadro-core, 4GB RAM) wie die Sau. Was hast Du erwartet? Auf solcher Hardware läuft einfach alles schnell (wahrscheinlich auch eine Waschmaschine). Aber was machst Du mit Deiner alten Hardware? Die funktioniert ja.

Natürlich wird Windows 7 jetzt gelobt, sonst kauft ja das Zeug keiner mehr. Die, die es kaufen, brauchen sowieso neue Hardware (außer sie haben schon Vista laufen).

Aber in Windows kenn' ich mich aus oder ich hab' jemanden, der sich auskennt.

Ganz ehrlich, was machst Du bei einem Fehler? Solange herumklicken, bis Du das richtige Hakerl erwischst hast. Wenn's nix hilft, neu starten oder gleich neu installieren.

In Windows behobst Du nie die Ursache, weil vieles einfach nicht zugänglich ist. Eine systematische Fehlersuche scheidet meist daran. Man kommt bis zu einem gewissen Punkt. Dort entscheidet man sich meist für eine Neuinstallation.

Das macht nix, installier' ich halt neu!

Nur zur Veranschaulichung: eine normale XP-Installation dauert im Schnitt 3-6 Stunden. Ja ich weiß, drei Mal auf *Weiter* klicken, bis die grüne Landschaft kommt, fertig! Das kann meine Mutter auch. Das ist aber nur die halbe Miete, denn danach musst Du noch alles konfigurieren

und installieren. Office, Daten zurückspielen, Anwendungen zuordnen, DVD-Software, Zip-Programme, Firewall, Virens Scanner, E-Mail-Programm, Dienste absichern, Drucker, Security-Patches einspielen, System konfigurieren, restliche Software installieren (Skype...). Also da kommt schon was zusammen.

Ich treff' immer noch auf Leute, die mir versichern, dass sie Windows in 40 Minuten installiert haben. Gähn! Alleine die Installation mit den aktuellen Security-Patches dauert gut eine Stunde und 20 Minuten. (Servicepack 3 hat ein paar Hundert MB!). Und dann musst Du erst noch den Rest vom Internet holen; da red' ich noch gar nicht vom Updaten des Internet-Explorers oder MS-Office.

Tja, und nach spätestens zwei Jahren ist sowieso eine Neuinstallation angesagt, weil die Kiste elend langsam geworden ist und dauernd irgendwelche Fenster aufpoppen. (Taschenlampe bei Verzeichniswechsel, Startvorgang bis Desktop benutzbar ist, rechts unten in der Leiste immer neue Symbole usw.) Ehrlich, ich habe vielleicht 5 Windows-XP-Boxen gesehen, die nach 5 Jahren noch immer anstandslos gelaufen sind. Ich meine ohne Neuinstallation zwischendurch. Der überwiegende Großteil der XP-Kisten war nach dieser Zeitspanne aber unbrauchbar.

Schau' einmal nach einer Neuinstallation, wie groß die Registry ist (ca. 600 MB). Ich hab' schon welche mit 2,8 GB gesehen. Kein Wunder, dass das System darunter leidet.

Aber anscheinend ist es den Leuten nicht abträglich, periodisch Windows neu zu installieren. Das (*Neu-Installieren*) hab' ich - ehrlich gesagt - unter Linux noch nie gemacht. (Außer am Anfang, das ich es von Windows her nicht anders annte).

Vielleicht bin ich bei der Neuinstallation länger gesessen, weil irgendwas nicht gleich funktioniert hat. Aber in der Regel seh' ich die Leute nicht mehr. Weil's einfach funktioniert. Auch nach Jahren. *set and forget*.

Windows ist in dieser Hinsicht ein Selbstläufer. Ich weiß genau, wenn ich Windows installier', seh ich den spätestens in einem Jahr wieder (irgendwas geht immer nicht...). Für einen EDV-Supporter ideal. Alter Spruch: wenn'st Geld verdienen willst, musst' Microsoft verwenden. (Lizenzkosten, Virens Scanner, Updates, Windows 7-kompatible Hard- und Software verkaufen, Software installieren, Trojanerschutz)

Nur können viele Firmen nicht so leicht raus. Wenn'st den Microsoft-Weg eingeschlagen hast, musst ihn bis zum Schluss mitgehen. Warum arbeiten die meisten Firmen noch immer mit dem Internet Explorer 6? Was glaubst Du, was da los ist, wenn die auf IE9 updaten. *Never change a running system...* oder besser, *never run a changing system*. Denk' einmal nach, wie lang es gedauert hat, bis die EDV in Deiner Firma problemlos gelaufen ist. Viele Anwendungen brauchen den Internet-Explorer im Hintergrund. Darum kann man in einer Firma auch nicht so einfach auf den Firefox umschwenken. Im Endeffekt kannst Du großteils nur mehr auf Microsoft-Produkte zurückgreifen, damit Dein Firmennetzwerk funktioniert. (Je mehr MS-Zeug im Netzwerk drin ist, um so inkompatibler wird das Ganze zu anderen Systemen (Linux, BSD, Unix...)).

Egal, tu ma des halt a bissal ausputzen und aufräumen..

Ist reine Kosmetik. Sitz' ich dann ein bis zwei Stunden und weiß genau, dass dieser Zustand maximal 2 Monate anhält, dann ist wieder alles

beim Alten. Installier' ich halt Firefox, damit die leut was Schnelleres als den Internet-Explorer haben, damit sie sich wenigstens ein bisschen freuen können, weil *"alles so schnell geht, jetzt"*.

Wenn die Basis versaut ist....

Alle PC (egal, ob Saturn, Cosmos, Media-Markt...) sind nicht internet-tauglich und schon gar nicht richtig konfiguriert. Die Firmen können (oder wollen) sich das nicht leisten. Die bügeln einfach das Betriebssystem drauf *thats it*; einstellen und absichen muss es der Kunde selbst. Das macht sonst keiner und üblicherweise auch kein (Laien-)Kunde. Wie willst du einem Laien erklären, dass er seine 6 GB große Videodatei nicht auf dem vorinstallierten FAT32-Filesystem speichern kann. Auch 2009 noch immer ein "Klassiker" (*oba des is jo neu des zeugl. seufz*). Vom abgelaufenen Virens Scanner ganz zu schweigen (*wieso, den hob I do mitzoit... wieder seufz*)

Und dann komm' meistens ich ins Spiel; Basis versaut, wie gehabt

Das heißt, der einzige gangbare Weg ist eine Neuinstallation, gleich nachdem man den PC gekauft hat; aber eine richtige Neuinstallation!

Für Linux gibts aber keine Firma, die sich kümmert.

Fühlst Du Dich beim Microsoft-Support wohler? Abgesehen davon, es kümmern sich mehr Leute um Linux als um Microsoft-Produkte. Ehrlich, und die machen einen guten Job.

Aber wenn Linux so gut ist, warum hat es nicht schon längst Windows verdrängt?

Hat es im Serverbereich sicher schon. Einfach weil es sehr stabil und transparent ist. Natürlich sind die Kosten auch nicht zu vernachlässigen.

Am Desktop (Heimuser) ist der Marketingvorsprung einfach zu groß. Auf jedem neuen PC hast Du automatisch Windows installiert. Wieso sollte sich ein Laie jetzt extra Linux draufspielen, wo er doch eh keine Ahnung hat.

Außerdem muss ich da etwas klarstellen: die Intention von Linux ist es nicht: *Linux auf jedem Schreibtisch*. Das hat sich Microsoft auf die Fahnen geschrieben. Linux will einfach nur gute Software machen. Dabei ist es egal, ob die von 8 Leuten oder 800 Millionen benutzt wird.

Linux hat keine Marketingmaschinerie im Hintergrund laufen wie Microsoft. Und Linux macht trotzdem hervorragende Software.

Schau mal auf die Unis. Die meisten Rechner laufen auf Linux; seit Jahren. Audi macht AutoCAD seit Jahren auf Linux-Basis. Die Kanarischen Inseln haben die Verwaltung großteils auf Linux umgestellt. Und es gibt genug weitere Beispiele in Südamerika, und Europa, insbesondere in Skandinavien.

Wenn es gratis ist und dabei nix taugen würde, das wär' nicht gut. Aber es ist gut! Die Stadtverwaltung Wien stellt seit Jahren auf Linux um (ca. 14.000 Rechner). Im Internet findest Du genug Beispiele.

Wenn so viele Leute Windows verwenden, kann es ja nicht so schlecht sein...

Wenn so viele Leute Übergewicht haben, kann es ja auch nicht so schlecht sein...

Windows ist nicht schlecht. Nur, viele Leute kennen halt nix anderes.



In Linux ist aber alles so kompliziert!

Stimmt, wenn man seit gut 10 Jahren auf Windows gearbeitet hat und dann auf einmal "alles anders ist". Es ist deshalb so kompliziert, weil Du noch Anfänger in Linux bist und weil Du der irrigen Annahme bist, dass alle PCs wie Windows funktionieren müssen. Kein Mac- oder Linux-User defragmentiert seine Festplatten. Warum? Weils nicht notwendig ist. Linux und Mac verteilen die Daten gleich "richtig" auf der Festplatte. Nur ein Beispiel für die Handhabung der verschiedenen Systeme.

Erinnere Dich zurück, wie Du mit Windows angefangen hast, wie kompliziert das alles war. Linux funktioniert seit 40 Jahren so. (Linux stammt von Unix ab). Windows seit gut 15 Jahren so. Frag' einmal einen erfahrenen Linux-User, ob er Schwierigkeiten bei der Bedienung hat - oder einen Apple-User.

Setz' einen Windows-Power-User an eine Linux-Maschine und er wirft nach zwei Minuten das Handtuch - weil er es nichts anders kennt (*in Windows hab' i des imma so g'macht*) Nimm' einen absoluten PC-Anfänger und setz' ihn zu einem Linux-PC - er wird keine großen Probleme haben. Warum? Weil er eben nichts anderes kennt.

Natürlich versuchen die Umsteiger am Anfang alles so zu machen, wie sie es in Windows gewohnt waren (des muas so gehen). Abgesehen davon, die Umstellung ist nicht wirklich gravierend. Du klickst wo drauf und irgendwas geht auf. Die Namen der Anwendungen sind anders. Na wenn schon, Dateiverwaltung funktioniert genauso oder besser (Einfügen mit mittlerer Taste).

Versuche einmal, ipv6 in Windows7 zu deaktivieren oder den Bibliotheken eigene Icons zu verpassen. Viel Spaß! Es geht, aber ehrlich gesagt, möchte ich es keinem normalen User zumuten, in der Registry herumzufuhrwerken oder Textdateien zu editieren.

Denk' einmal nach, wie viel Zeit Du in Windows für Dein heutiges Windows Know-How investiert hast. Investiere diese Zeit in Linux, dann kannst Du einen objektiven Vergleich setzen, ob Linux kompliziert ist. Lass Dir gesagt sein: Es ist nicht kompliziert, nur anders :-). Aber wenn man jahrelang Windows benutzt hat, tut man sich schwer in Linux. Warum? Weil man nicht weiß, wie's funktioniert. (Keine setup.exe, kein Defragmentieren, kein Reboot bei Software- und Treiberinstallationen, keine zusätzliche Security-Software, keine Shareware, strikte Usertrennung...)

Versuch mal, Dich im Netzwerk von einer Linux-Box auf eine Windows-Box zu verbinden. Und Du weisst, was ich meine...

Kann ich mit Linux chatten, surfen, Briefe schreiben, drucken, scannen?

Ja!

Aber wenn ich Linux installiere, muss man sich schon auskennen...

Wenn Du Windows installierst, solltest Du auch wissen, was Du tust. Ehrlich! Nicht umsonst gibt es einen EDV-Support in jeder Firma. Aber ich hab' einen 17-jährigen Neffen, der kennt sich ur gut aus. Die Tatsache, dass jemand 6 Stunden am Tag chattet und surft, macht ihn nicht zum PC-Profi. Oder ist jeder Berufslenker automatisch vertraut mit der Mechanik und Elektronik des Autos?

Nicht falsch verstehen, natürlich wird es 17-jährige geben, die wissen was sie tun. Die Praxis zeigt es mir aber nicht. Die meisten sind Welt-

meister im Neuinstallieren - so alle 2 bis 12 Monate.

Ich kenne studierte Wirtschaftsinformatiker, die kennen jedes Detail von Outlook aber haben keine Ahnung, wie man ein Betriebssystem absichert oder einrichtet. Ich kenne auch einen Maler, der komplette Netzwerke einrichtet, als Hobby, weil er sich dafür interessiert und ständig dazulernt - nur so ein Beispiel. Ich kenne Programmierer, die haben keine Ahnung, wie sie ein Bild in Word einfügen sollen. Alles verständlich!

Du musst die EDV wie die Medizin betrachten. Mittlerweile gibt es schon so viele Spezialrichtungen für die Du jeweils Spezialisten brauchst. (Betriebssysteme, Firewall, Schadsoftware, Datenbanken, Server, Wlan, Security, Hardware, Netzwerktechnik, Verschlüsselung, Office, Webdesign, Multimedia, Programmierung.. usw.). Ich kenne keinen, der alle Fachgebiete in der EDV beherrscht.

Du musst Dich eben auf ein Gebiet spezialisieren und aktuell bleiben.

Keiner verlangt, dass man jetzt die Interna seines Betriebssystems lernen muss. Aber wenn ich's nicht kann, muss ich halt wen konsultieren; einen, der weiß was er tut. Und das gilt für Windows und Linux gleichermaßen.

Oder plombierst Du Dir auch selber Deine Zähne, weil Du im Fernsehen gesehen hast, wie es geht?

Das Problem ist, dass die meisten glauben, sich in Windows auszukennen - *des kann ja net so schwer sein, hob i scho 100mal gmocht*.

Und wenn erst einmal die Basis versaut ist - da red' ich noch gar nicht von Linux - und wenn erst die halbe Verwandtschaft a paar Stunden auf der Kiste herumgeklickt und das System zerschossen hat, soll ich's richten. Keine Chance!

Neuinstallation, 3-6 Stunden Arbeit und die Geschichte sollte wieder ein Zeitl rennen.

Windows is ja so leicht zu installieren. Die Arbeit fängt nachher erst an. Beim Konfigurieren des Systems gehen schon ein paar Stunden drauf; immer. Auch bei Linux. Abgesehen davon. Reden sie mit Leuten, die sich am PC auskennen und nicht in Windows. Da gibt's einen großen Unterschied.

Aber i brauch ja net so an Profi für mein PC; i tua ja nur a bissal surfen, Bilder schauen und emailen.

Sagst Du das zu Deinem Automechaniker, Installateur, Masseur, Zahnarzt, Elektriker, Gärtner, Steuerberater... auch, dass Du keinen Profi brauchst? Dann kannst es ja gleich selber machen, wennst keinen Profi brauchst :-))

Kein Problem, aber dann erwarte nicht, dass alles tadellos funktioniert. Merke: Betriebssysteme installieren ist nix für Jedermann, auch wenn das Microsoft immer wieder glauben machen will.

brauchst e nur imma auf "weiter" druckn...

Die meisten wissen gar nicht mal, was sie da tun und wenn erst mal die Basis versaut ist, wundern sich die Leute, dass der PC spinnt.

Hast Du schon 'mal wen die Zylinderkopfdichtung am Auto tauschen gesehen, der keine Ahnung davon hat? Warum traut sich dann mittlerweile jede zweite Kindergartenhelferin (nix gegen die) zu, ein Betriebssystem zu installieren, oder gar in der Registry zu tricksen?

Ich hab Leute neben mir weinen gesehen, weil ihre Daten weg waren. *"i hob jo nur do*

aufgedruckt...". Lass die Finger davon, wenn Du's nicht kannst, oder lerne es, alles andere ist auf Dauer sinnlos.

Ich habe Leute getroffen, die haben 15 Minuten gewartet, bis sich ein Fenster geöffnet hat. *"i bin daweil imma rausgegangen..."*. viele glauben, dass das bei Windows nach einiger Zeit normal ist. (keine Frage, das war ein erstklassiger Spamroboter). Wie erklärst Du das aber einem Laien, dass Du jetzt 5 Stunden am PC werkeln musst (*dos des so lang dauert, hätt I net docht...*). Vom lieben Nachbarn (obwohl er sich sehr bemühte und stundenlang herumgeklickt hat) mit eigenen CDs installiert, ein Klassiker und vollkommen sinnlos. *"er hot ka Zeit mehr ...i soi neich starten..."*. Nicht umsonst ist der EDV-Techniker ein Lehrberuf.

Is wurscht, tua ma des nur a bissal herrichtn, dass' geht...

Vergiss es, haben beide nix davon. Er, weil er sich über den Kübel und über mich ärgert und ich, weil ich drei Mal in der Woche wegen irgendeinem Sch**** angerufen werde, der nicht funktioniert. Darum mach' ich sowas nicht....

Wo bekomme ich Software her?

Aus dem Internet. Gratis.

Viel Spaß mit Deinen gecrackten XP- und Office-Versionen. Glaubst Du, Microsoft ist dumm? Natürlich wird es immer Cracks geben. Aber der Mehraufwand für den User wird auch immer größer, damit seine gecrackte Software läuft. Microsoft fährt seit Jahren eine Zermürbungstaktik. Und wenn Du glaubst, dass im Netz eh alles zum Downloaden ist. Hast Du recht. Weißt Du aber auch was Du da alles downloadest?

Glaubst Du wirklich, irgendwer stellt im Internet MS-Office oder Spiele gratis zur Verfügung oder was weiss ich. Keiner weiß, was da noch zusätzlich auf Deinen Rechner kommt.

Nicht umsonst sind in Firmen die CD-Laufwerke deaktiviert und USB-Sticks nicht gern gesehen; wer weiß schon, was da alles drauf ist und das ist dann alles im Firmennetz :-)

Aber es funktioniert ja alles, ich hab noch nie einen Virus gehabt

Bist Du sicher? Glaubst Du wirklich, dass Du als Laie das entscheiden kannst, ob Du verseucht bist?

Nur, weil Norton Antivirus *alles okay* meldet? Wenn da nicht periodisch Fenster aufpoppen würden, würde keiner das Zeug kaufen. *"des is ja hin...der zeigt ja nix an"*. Aber so ist der User beruhigt, wenn mal hin und wieder ein Fenster aufpoppt *"Virus geblockt...blabla..."*

"Super Virens Scanner" denkt er sich. Eh klar. Wer sagt Dir, dass der Virus den Scanner nicht so manipuliert hat, dass diese Nachricht erscheint? Glaubst Du alles, was im Fernsehen ist? Warum klammerst Du Dich dann so an Deinen Scanner und seine Aussagen?

Wenn Du eine aussagekräftige Analyse haben willst, ob Du einen Virus hast, musst Du immer von einer externen Quelle aus deinen PC scannen. Alle Aussagen vom eigenen PC aus sind nicht aussagekräftig (Selbsttest, Check.). Du musst auch zu einem Arzt gehen, um Dich untersuchen zu lassen. Selbstdiagnosen sind da auch fehl am Platz. Das heißt, Du müsstest von einer CD aus Dein System scannen; dann hättest Du eine aussagekräftige Analyse.

In Windows brauchst Du einen Virens Scanner auf jeden Fall, auch, wenn's nicht immer optimal ist.



Irgendwie paradox, dass ich ein Sicherheitsfeature brauche für ein Betriebssystem, für das ich ein paar hundert Euro gezahlt habe. Und ich muss mich um die Sicherheit kümmern, nur weil Microsoft nicht in der Lage ist (oder sein will) ein resistentes Betriebssystem herauszugeben.

Andererseits: in der Sicherheitsbranche werden durch diesen Umstand Milliarden verdient!

Aber mein Virens Scanner hat nix gefunden

Die Aussagekraft eines lokalen Virens Scanners geht immer gegen Null. Punkt. Das Erste, was ein Virus auf einer Maschine macht: er manipuliert den Virens Scanner (schaltet ihn aus oder täuscht ihn). Die Virenschreiber sind keine Trottel und auf der "dunklen Seite" gibts halt einfach mehr Kohle. Und es geht da hauptsächlich um Kohle; ehrlich!

Wenn Dein PC im Besitz eines anderen ist (Botnetze) hat er einen gewissen Wert. Und wird im Internet mit 100.000 anderen verseuchten PCs weiterverkauft (an Spammer, Direktvertriebler, Versicherungen, Pädophile, kriminelle Organisationen; wer halt am besten zahlt).

Was mach ich bei Virenbefall in Windows?

- Internetverbindung trennen (Kabel ziehen, bei Wireless-Modem ausschalten)
- Daten sichern (soweit möglich)
- Neu installieren oder Image einspielen
- alle Passwörter tauschen

Was macht ein Virus auf einem PC als erstes?

Er manipuliert die Sicherheitseinrichtungen (Virens Scanner, Adblock...); er holt Verstärkung übers Internet (richtet Hintertüren ein, verbreitet sich im System).

Das Problem ist meist nicht, diesen erkannten Virus zu entfernen, sondern "Was hat der Virus in der Zwischenzeit alles gemacht?" (Daten ins Internet geschickt, Verstärkung geholt, Daten und System manipuliert, Zugang zu Security-Seiten gesperrt...). Deshalb sind die sogenannten "Removal-tools" eher mit Vorsicht zu genießen. Im günstigsten Fall entfernen sie wirklich den Übeltäter. Bist Du Dir aber jetzt sicher, dass er nicht schon was anderes am System manipuliert hat, oder was vom Internet nachgeladen hat? Eben. Im schlechtesten Fall hast Du den Übeltäter entfernt (oder doch nicht) und noch zusätzlich ein paar seiner Freunde auf dem System. Egal, wie Du es auch drehst, das System ist kompromittiert. Der einzig sichere Weg, um solch ein System wieder auf die Beine zu bringen, ist eine Neuinstallation. Oder ein sauberes Image einzuspielen. Hat sich auch 2010 nix daran geändert. Zeit und Nerven, das kostets halt immer.

Und wenn man noch so vorsichtig ist. Einmal drückt man wo drauf (war der Finger schneller als das Hirn), und man ist live dabei. Mir selber auch schon passiert. Mit Windows bist Du im Internet halt immer in der "Auslage". Immer erste Reihe fußfrei dabei. "Always keep a low profile"...sagen die Engländer, wenn sie sich im Internet bewegen. Heißt nix anderes, beweg Dich unauffällig dort. Alle MS-Sachen fürs Internet (Outlook, Browser, Messenger...) durch Alternativen ersetzen. Abgesehen davon, dass die meisten Alternativen gratis und auch sehr gut sind. Die Angriffsfläche im Internet verringert sich. Egal, ob Outlook, Internet-Explorer oder MSN. Die verwenden alle die selbe "Tür" im System zum Internetzugang. Du brauchst nur diese eine Tür knacken...(trivial gesprochen jetzt.) Das ist auch der Grund,

warum Firefox in Windows relativ lange zum Starten braucht. Er ist halt nicht ins System eingebunden, und muss sein "Zeug" (Bibliotheken...) erst selber laden. Beim Internet-explorer wird es ja von Haus aus gleich mitgeladen. Deinstallation von Internet-Explorer -> Fehlzanzeige...

Aber ich hab ja nix auf meinem PC, was soll der bei mir?

Er missbraucht Deinen PC als Spamversender, alle Deine Kontakte im Adressbuch werden mit Spam besucht, Dein Plattenplatz kann für dubiose Inhalte missbraucht werden, Deine Internetleitung für Angriffe auf große Firmen missbraucht werden, Deine Daten kursieren im Internet usw.

Warum glaubst Du, sind im Frühjahr 2009 über 3000 Windows PCs in Kärntens Spitälern und Landesverwaltung ausgefallen? Durch einen Windows-Virus. Von den Tausenden Privatrechnern und Firmen in Österreich, die auch betroffen waren (und noch immer sind) red' ich ja gar nicht, geschweige denn von den globalen Auswirkungen.

Linux ist virenresistent; nur so nebenbei.

Wo sind meine Laufwerksbuchstaben?

Nicht da! Linux braucht keine Buchstaben, alle Deine Geräte und Daten sind in Ordnern.

/dev/hda Deine Festplatte, /mnt/vl-hot/sda Dein USB-Stick, /home/benutzername Deine "Eigenen Dateien", /mnt/cdrom Deine CD.

Du siehst, alles fängt immer mit einem Schrägstrich (/) an; von dort geht alles weg. Das ist der Ordner, wo alles drin ist (bin, home, media, dev...)

In /home/benutzername/ sind alle Deine persönlichen Daten drin (Bilder, Dokumente, Filme...)

In /mnt/vl-hot/ sind alle Deine USB-Geräte mit Inhalt drin (Kamera, Stick, externe Festplatte...)

Im Prinzip brauchst Du auch nicht mehr, denn überall anders hast Du sowieso keine Rechte, dort ist nur Systemklumpat, also uninteressant für Dich als User.

Wieso hab ich nur unter /home rechte ??

Weil Du als normaler User angemeldet bist, wie in Deiner Firma am Arbeitsplatz. So kannst Du nichts unabsichtlich kaputt machen.

Linux ist sicherer als Windows

Nein und ja. Linux hat nur alle Werkzeuge schon an Bord, um es abzusichern. Bei Windows musst Du diese erst kaufen. Aber egal, ob Windows oder Linux, Du musst wissen, was Du tust.

Die Sicherheit steht und fällt mit dem, der den PC einrichtet. Da in der Regel Linux-User mehr technik-affin sind, wissen sie meist auch, was sie tun.

Außerdem ist Linux für das Internet gemacht, Windows für das Internet zurechtgebogen.

Bei Windows wagt sich mittlerweile jede frustrierte Kindergärtnerin an eine Neuinstallation heran. Wird scho irgendwie gehn. "des hob i jo net gwusst" "i hob des afoch installiert" "des hot ma jo kana gsogt" "wie soi i des wissn", sind so die typischen Aussagen danach. (z.B. NTFS schnell formatiert > keine Überprüfung der Festplatten-Sektoren -> schlecht) Netbios über TCP/IP nicht abgedreht > Daten über Netzwerkschnittstelle ins Internet freigegeben, auch schlecht). Datei- und Drucker-freigabe nicht deaktiviert -> ditto. Wenn ich

mich wo nicht auskenne, lass' ich die Finger davon; aber erzähl' das mal jemanden.

Vorteil bei Linux ist, dass von Haus aus alles restriktiver ist. Alles was nicht ausdrücklich erlaubt ist, ist in Linux verboten.

Windows dreht in der Regel alles auf "wau...des funktioniert ja alles auf Anhieb".

Das ist auch der Grund, warum in Linux viele frustriert sind, wenn was nicht gleich funktioniert und wieder zu Windows wechseln. Weil sie eben nicht wissen, was sie tun... woher auch?

Man kann auch Windows absichern - bis zu einem gewissen Grad. Aber wenn ich die Wahl hätte, würde ich Linux bevorzugen. Es ist in meinen Augen einfach ausgereifter; und läuft auch nach Jahren noch ohne großen Administrationsaufwand.

Warum sollte ich Linux verwenden?

Keine Ahnung, warum verwendest Du Windows? Es gibt Fanatiker in beiden Lagern. Probier' einmal Linux aus (virtuelle Maschine in Windows...). Und wirf nicht nach 3 Minuten schon das Handtuch. Hast Du ja bei Windows auch nicht, als Du das erste Mal partitioniert hast :-)

Ich habe Linux probiert, komme aber damit nicht klar. Ist zwar nett, aber nix für professionellen Einsatz...

Wenn es wie Windows funktionieren würde, könnte es nur maximal so gut wie dieses werden. Viele wollen einfach ein Linux, das wie Windows funktioniert, nur ohne dessen "Fehler". Das geht aber nicht.

Die Erwartungshaltung eines Windows-Users kann man mit einem Satz widerspiegeln: "Is des Linux besser als Windows". Und da liegt der Hund begraben. Man kann Linux nicht mit Windows vergleichen. Beides sind Betriebssysteme, ja. Aber beide haben verschiedene Philosophien und Konzepte. Da diese zu unterschiedlich sind, ist das mitgebrachte Windows-Systemwissen in Linux bedeutungslos. Ehrlich, glaub's mir. Die Genialität und der logische Aufbau von Linux kommt erst nach ein paar Monaten ins Bewusstsein des Admins, wenn er den Durchblick hat. Ohne Windows-Vorwissen ginge es bedeutend schneller, da man anfangs nicht krampfhaft versucht, alles nach der Windows-Methode zu machen. Das nur nebenbei. Ging mir auch so....

Denk' einmal nach, womit sich der Erfolg des Firefox erklären lässt. Er funktioniert einfach nicht so wie der Internet Explorer (Tabs, Suchmaschinenfeld rechts, modular erweiterbar, aktuell, standardisiert, Suchfunktion) Wenn er so funktionieren würde wie der Internet Explorer, wäre er nur ein weiterer Browser unter vielen anderen.

Merke: Die meisten "Schwierigkeiten" mit Linux haben Windows Power-User. - (Des muas so funktionieren...). Als Windows-Systemspezialist fängst Du in Linux bei Null an. Das ist für viele schwer zu verdauen (war bei mir auch so...). Vorteil: dieses Linux-Wissen ist für ewig. An den Basics hat sich seit 40 Jahren nichts geändert. Du brauchst es nur ein Mal lernen. Vergleiche mal Win98 mit Win7. (SAM, Registry, NTFS, Rechteverwaltung, Lizenzmodell, Defender, UAC...)

Bezüglich der Professionalität: Linux ist ein Bastlersystem! Ja, auch. Linux ist aber auch ein verlässliches Serverbetriebssystem oder ein robustes Workstation-System; es kommt darauf an, was Du machen willst. Die Tatsache, dass Linux



im Privatbereich nicht sehr präsent ist, kann kein Indiz für Unprofessionalität sein. Viele Unternehmen setzen bei Internetdiensten (Web, Mail...) auf Linux. Warum? Nicht weil es „cool“ ist. Weil es dafür gemacht wurde, ganz einfach. Linux ist ein vollwertiges Netzwerkbetriebssystem.

Unterschiede der Benutzerverwaltung

Linux stammt vom Server ab, Windows vom Desktop. (Mehrbenutzersystem, Rechteverwaltung...). Hintergrund: Windows ist zwar mittlerweile auch ein Mehrbenutzersystem, aber nur auf Prozessebene. Versuch einmal, Dich in Windows mit Deinem Benutzernamen (oder einem anderen Usernamen zusätzlich einzuloggen...) es wird nicht gehen, bei Linux schon, weil es ein Mehrbenutzersystem ist! So "unsinnig" dieses Feature im Privatbereich auch erscheint, in einer Firmenumgebung und im Netzwerk ist es Gold wert. Linux ist von Haus aus ein vollwertiges Netzwerk- und Mehrbenutzersystem. Wenn Du sowas auch in Windows willst, (und in einer Firma brauchst Du sowas, glaub's mir), musst Du Software dazukaufen (Terminalserver, Citrix,..). Deshalb darf ein "user" in Linux nur das machen, wozu er auch berechtigt ist. (da ja andere User auch auf der selben Maschine arbeiten könnten. Und das ist einmal gar nix am System. Bei ihm „daheim“, bei seinen eigenen Daten kann er machen, was er will; sein Bier. Auch wenn man „alleine“ am System ist, arbeitet man immer mit einem eingeschränkten Userkonto. Warum? Damit man nichts unbeabsichtigt am System kaputt machen kann. Und damit eventuelle Schadsoftware am System nichts ausrichten kann. Jeder User darf nur das, was ihm explizit erlaubt wurde. Der Windows-Weg mit UAC ist deshalb schon richtig. Nur an der Akzeptanz der User scheitert es noch oft (*wieso darf i des net..kannst ma des blede pop-up abschalten, was da immer kommt*). In XP war man es gewohnt, dass man alles machen durfte. Das heißt die "Umerziehung" des Users ist derweil noch die größte Hürde in Windows. Für einen Linux-User ist es sonnenklar, dass er am System nichts verändern darf (nicht mal die Uhrzeit).

Warum haben alle Linux-Freaks einen Vollbart?

Du solltest nicht so viel Fernsehen...

Ein Freund, der sich mit PCs auskennt, meint, dass Linux zu schwierig für Heimanwender sei.

Dein Freund kennt sich wahrscheinlich mit Windows aus. Aber nicht mit Linux.

Um Missverständnisse gleich auszuräumen: Bei Windows traut sich ja mittlerweile jeder an die Registry heran. (*des kann ja net so schwer sein...i hob do an guatn registrycleaner...*).

Wenn dann die halbe Verwandtschaft sich erfolglos durch das System geklickt hat wird halt neu installiert (Falls es schon mal jemanden aufgefallen ist. Windows wurde bis jetzt regelmäßig (Intervall von 2 Monaten bis 4 Jahren) neu „aufgesetzt“. Warum? Weil viele Bereiche nicht zugänglich sind (*closed source*) zum Reparieren oder das System träge geworden ist (oder beides). "*Der is kaputt..muas ma neich installieren.*" Dieser Satz scheint mittlerweile in der Windows-Welt salonfähig zu sein. Leider behebt das selten die Ursache. In Firmenumgebungen ist es mittlerweile Standard: Rebooten -> geht nix -> neu installieren übers Netzwerk. Der einfachste Weg, ja. Wenn'st eine gute Sicherung hast und dann den PC wieder in den Ausgangszustand bringst; naja, vergeht auch ein Zeil!

In Linux kannst Du viel „reparieren“ (heißt einfach Textdateien editieren). Und das funktioniert auch.

Bezüglich dem oft strapaziertem Begriff „schwierig“ in Linux: Schwierig ist nur das was man nicht kennt. Wenn Du einmal professionell mit Textdateien arbeiten willst (musst), kommst Du um Linuxe nicht herum.

Um eine 12 GB große Textdatei (ja sowas gibt's) zu bearbeiten, sind Linux(Unix)-Tools die Antwort. Extrem schnell und zuverlässig. Hauptsächlich auf der Kommandozeile. Deshalb ist Linux im professionellen Umfeld eine fixe Größe. Bei den „bösen Buben“ im Internet sowieso. Es verarbeitet riesige Datenmengen sehr zuverlässig. Auch ein Grund, dass die meisten Supercomputer unter Linux laufen (Speicherverwaltung). Versuch' 'mal, ein 100-Seiten-Word-Dokument in Office aufzumachen. Viel Spaß! Das aber nur am Rande erwähnt.

Was gratis ist, kann ja nix wert sein...

Noch einmal. Die Intention von Linux ist es, gute Software zu machen. Wenn Du mit Linux Geld verdienen willst, biete Support an. (Nur ein Weg von vielen). Das Thema „bezahlen für Software“ im Heimanwenderkreis grenzt an pure Polemik. Die Akzeptanz, für Software Geld abzudrücken, pendelt sich um den Wert 0 ein. Das ist mitunter auch einer der wichtigsten Gründe für die periodisch auftretenden Virenwellen. Glaub' mir, die nächste Welle kommt bestimmt. Warum? Denk' einmal nach. Fertig? Microsoft ist nicht dumm (waren sie auch nie; nebenbei, die machen auch Super-Software). Ich schätze, ab Juni, wenn die kostenlos heruntergeladenen Windows7 Lizenzen zu bezahlen sind, wird sich einiges am Schadsoftwaremarkt tun. Jeder „clevere“ Heimanwender wird sich einen „crack“ für sein Win7 besorgen und einspielen. Super. Nur bekommst Du halt keine Sicherheitsupdates mehr. Was dann passieren wird über kurz oder lang, ist auch klar (sasser, sven, conficker lassen grüßen...). Alle werden wieder über dieses „sch... windows“ schimpfen.

Wenn ich Software regulär kaufe und sie funktioniert nicht, darf ich mich aufregen. Aber wenn schon die Basis versaut ist; was erwartest Du?

Natürlich wird es immer Cracks geben. Aber ehrlich, welcher User ist daran interessiert, dauernd irgendwelche Sachen in die Registry „einzupflegen“?

Wenn Du Software von Microsoft willst, kaufe sie. Ist nix Schlechtes daran. Wenn Du Software von Linux willst, lade sie Dir gratis herunter.

Wenn Du keine passende Software bei Linux findest, bleib bei Windows oder lerne um.

Heimanwender-Zeug geht *out of the box* in Linux (Office, video, brennen...)

Wenn Du Spezialsoftware brauchst, die nur in Windows verfügbar ist, bleib bei Windows oder freunde Dich mit einer Alternative an)

Muss ich programmieren können, um mit Linux zu arbeiten?

Nein. Nur lesen...

Und die Eingewöhnungszeit?

Als User 2 Wochen, als Administrator ein paar Monate.

Der typische "Weg" eines Windows Users

Linux-CD rein, herumklicken, schauen, net auskennen, abrechnen, CD raus, Windows starten...

Der "richtige Weg" eines Windows-Users

Fachliteratur lesen, dann installieren. Besser, einen Linux-Freund einladen und gemeinsam installieren. Den Linux-Freund mit Essen und Trinken gnädig stimmen und Fragen stellen, eh klar.

Wen rufe ich an, wenn ich mich nicht auskenne?

Deine "Probleme" löst Du hauptsächlich übers "Netz" (Foren, IRC...) oder eben mit Hilfe eines linux-kundigen Freundes.

Welche Distri soll ich nehmen?

Nimm die, die der Typ hat, den Du immer um 10 Uhr abends anrufst, wenn was nicht funktioniert...

Merke: Du nimmst grundsätzlich das Linux, welches auf deine Hardware passt. (Grafiksystem läuft sehr gut ab 256 MB Ram....ja das gibt's wirklich noch im Privatbereich)

In Windows nimmst Du die Hardware, die zu Deinem Betriebssystem passt.

Wie lange dauert eine Installation?

Von 5 Minuten bis 25 Minuten. (aktuelle Hardware, lauffähiges System..Multimedia, Office, Brennen) Konfiguration: von 0 bis 5 Tage :-)

irgendwas will man immer anders haben...

Wieso sind die meisten Infos auf Englisch?

Weil das eine weitverbreitete Sprache ist. Alle Befehle sind in Englisch (wie in DOS). Du kannst einen Internetzugang mit englischen Befehlen in Afghanistan, St. Pölten, Norwegen oder Uganda einrichten. Beispiel: Mit diesem Zweizeiler aktiviere ich die erste Netzwerkschnittstelle, und hole mir dann sogleich eine IP-Adresse und schon bin ich im "Internet":

```
ifconfig eth0 up
dhclient eth0
```

Nicht weil es so "cool" ist, sondern relativ einfach, am System was zu ändern.

Warum tun die immer Text eintippen statt mit der Maus zu klicken?

Weil's einfacher ist. *Simple as that*. Und weil man damit die grafischen Konfigurationsprogramme (und deren eventuelle Fehlerquellen) ausschaltet. Manche Sachen gehen auch mit der Maus schneller. Beide Wege haben Vor- und Nachteile...

Übrigens: auch in Windows bist Du als Systemspezialist oft auf der Kommandozeile unterwegs (netuser, ipconfig...)

Wo und wie wird Linux konfiguriert?

Über Textdateien. Für den User im "home"-Verzeichnis, systemweit im "etc"-Ordner.

Wieso läuft in Linux vieles nicht auf Anhieb?

Linux: Alles was nicht ausdrücklich erlaubt ist, ist verboten (Mehrbenutzersystem). Das ist der sichere Weg. Alles dicht machen. Wenn der User was will, muss er es (als Administrator) freischalten.

Windows: Wir drehen mal alles auf...die User werden eh die unnötigen Sachen wegschalten... (Dienste, Progs... Wie viele Windows-User machen das wirklich? Eben!)

Merksatz: In Linux sitzt Du wahrscheinlich länger bei der Konfiguration. Aber dafür hast Du dann jahrelang Ruhe.



Die "bösen Buben" im Internet wissen das natürlich auch. Das ist der "Klassiker" in Windows. Auf einer normalen Workstation kannst Du in Windows gut ein Drittel der Dienste wegschalten (XP). Natürlich ist das meiste standardmäßig aufgedreht. In Linux musst Du es erst explizit aufdrehen. Wer braucht in Windows einen Server- oder USV-Dienst (im Privatbereich)? Ganz zu schweigen von Netbios über TCP/IP?

Gilt für Windows und Linux: so wenig Dienste wie möglich, so viel wie nötig. Und man sollte wissen was man tut. Das macht man ein Mal bei der Installation und gut is'. Warum? Was nicht läuft, kann nicht kaputt gehen oder angegriffen werden. Des weiteren können andere Sachen am PC nicht beeinträchtigt werden. Außerdem "entkompliziert" es das System und schont Ressourcen.

Wieso dreht man bei Linux nicht gleich die "üblichen Sachen" auf?

Machen sie in der Regel bei "Anfänger"-Distributionen (Ubuntu oder Suse) eh.

Normalerweise weiß ja keiner was Du mit dem Linux machen willst (Cluster für Atomkraftwerk, Workstation für die Mizzi-Tante, E-Mailserver für 2000 Clients, Firewall in der Nationalbank, Terminalclient, Router in der Schule, Streamingserver, Webserver, Fileserver, Gateway, Proxy, Backupserver, Internetcafe...*whatever*).

Du fängst mit einer 150 MB großen CD an und kannst die oben erwähnten Beispiele realisieren. Voraussetzung: eine CD, Hardware und eine Internetleitung; und Du musst wissen, was Du tust :-)) Linux-Credo: Installiere Linux und mach' damit, was Du willst...

Woher bekomme ich Software?

Aus dem Internet...gratis und legal...(einfach den nächsten Linux-User fragen, oder den Vortragenden an den Clubabenden)

Was ist das "Killerfeature", damit ich auf Linux wechsele?

Keine Ahnung, welches ist es bei Windows? So blöd es auch klingt, die Intention von Linux ist es nicht, auf jedem Schreibtisch präsent zu sein. Gute Software zu machen, trifft es eher. Der kommerzielle Aspekt wird natürlich von den Firmen (Novell, Ubuntu, RedHat...) in den Vordergrund gerückt. Ist ja nix Schlechtes.

Aber wenn Linux nicht mehr „frei“ wäre, würde man die goldene Gans schlachten...

Alle (Linux-)Firmen profitieren von den Tausenden Entwicklern weltweit, die sich mit Linux beschäftigen. Zurück zum Killerfeature: Wenn Du auf Photoshop und Dreamweaver „angewiesen“ bist, bleib' bei Windows. Du kannst es zwar auch in Linux laufen lassen aber warum? In Linux gibt's diese Programme nicht, es sind Windows-Programme. Du kannst aber Linux Programme nehmen. Viele große Firmen arbeiten heute mit Open-Source-Software... (Die ARD schneidet Filme mit "Kino", Audi hat die Entwicklungsabteilung seit 6 Jahren auf Linux migriert, der Film "Titanic" wurde mit einem Linux-Cluster für Spezialeffekte realisiert, *whatever*.)

Probier' Linux einfach aus, und schmeiß' die Flinte nicht gleich ins Korn, **Wenn Du** Dich nicht auskennst. Hast Du ja bei Windows98 auch nicht gemacht :-)

Brauch' ich in Linux einen Virenschanner?

Wenn Dein Linux ein Gateway für Windows-Boxen ist, würde es sich anbieten (dann wird am

Gateway schon gescannt). Wenn Du aber "nur" alleine am Internet hängst, nein, sinnlos. Es würde nur nach Windows-Viren gescannt werden und die sind in Linux ohne Bedeutung. Kein Defender, Spybot, Adaware, Registrycleaner, Firewall, Virenschanner...ehrllich, auf einer Workstation brauchst das nicht. Wenn'st paranoid veranlagt bist, installier' Dir halt eine grafische Firewall, wenn's Dir hilft, warum nicht.

Du musst nur wissen, was Du tust beim Installieren, *thats it*.

Übrigens 90 Prozent aller Spam wird über gehackte Windows-Boxen versendet. In diesem Sinne ist auch die in Win7 eingeführte UAC zu begrüßen, sodass wir in Zukunft von weiterem Spam verschont bleiben; von "conficker-Szenarien" ganz zu schweigen...

Warum es für Linux keine Viren (in freier Wildbahn) gibt? Da es keinen einheitlicher Aufbau gibt, besteht erhöhter Programmieraufwand. (Schau Dir die Desktops von 10 verschiedenen Linux-Usern an. Keiner wird denselben wie der andere haben, selten zumindest. Warum? Jeder, wie er will. Bei Windows hast Du alles von der Basis an vorexerziert, wie Du das benutzen musst. In Linux hast Du immer die Wahl. Durch das strikte Rechtesystem (entweder user oder root) und die starke Community (binnen Stunden wäre ein Patch verfügbar).

Ich kann kein Englisch! Welches Linux?

Nimm' Open Suse, dort ist alles in Deutsch. (auch die man-pages (das sind Beschreibungseiten für Programme). Ubuntu ebenso. Ps: Im Prinzip kannst Du (fast) jedes Linux mit vertretbarem Aufwand eindeutschten.

Typische "wenn" Sätze

Wenn Du Anfänger bist oder Windowsumsteiger, nimm' Open Suse de.opensuse.org oder Ubuntu ubuntuusers.de (beide sind sehr gut und in Deutsch)

Wenn Du die zwei populärsten Oberflächen in Linux kennenlernen willst: www.gnome.de, www.kde.org

Wenn Du "nur" arbeiten willst mit Linux, lass' es Dir von einem kompetenten Freund installieren und erklären.

Wenn Du das System kennenlernen und warten willst, besorg' Dir Fachliteratur und nimm Dir Zeit.

Wenn Du mit Linux einen Windows-Ersatz suchst, liegst Du komplett falsch. Das wird nicht funktionieren.

Wenn Du mit Linux eine Windows-Alternative suchst, liegst Du richtig.

Wenn Du einen Überblick haben willst, welche "Linuxe" es überhaupt gibt: www.distrowatch.com

Wenn Du einen Überblick über den Kernel haben willst (Eingeweide von Linux): www.kernel.org

Wenn Du ein Linux für eine Schulumgebung suchst: www.skolelinux.org

Wenn Du ein professionelles Linux für eine Firmenumgebung brauchst: debian, slackware, ubuntu, centos

Wenn Du Support für debian oder slackware brauchst: jeder gute EDV-Dienstleister bietet das an.

Wenn Du Support für die Firma gleich dazukaufen willst: www.redhat.org, www.suse.de, www.ubuntu.de...

Wenn Du Linux auf 4000 Rechnern installieren willst. Keiner hindert Dich daran, ist alles legal.

Wenn Du Linux auf Mehrkernprozessormaschinen als Server installieren willst, ist auch legal.

Wenn eine neue Version herausgegeben wurde, dann weil sie fertig ist, und nicht weil das Weihnachtsgeschäft anfängt

Wenn Ubuntu eine neue Version herausbringt, sind sechs Monate seit der alten vorbei und die haben immer das Neueste drin.

Wenn Du 2001 slackware 8 installiert hast, wird es auch heute noch mit Patches unterstützt.

Wenn Du einen 386er mit 64 MB RAM hast, mach' einfach einen Router draus. Linux läuft da drauf super...

Wenn Du "komplizierte Sachen" in Linux verständlich erklärt haben willst: www.ubuntuusers.de

Wenn Du als root arbeitest: lieber 5 Minuten nachdenken, als nachher 5 Stunden fluchen...

Wenn Du als root arbeitest, nimmt Linux an, dass Du weißt was Du tust.

Wenn Du als root arbeitest, gibt's keine Pop-Ups mit fragen (à la: *Möchten Sie wirklich löschen?*) Siehe oben.

Wenn Du in Linux was löschst, ist es generell weg.

Wenn in Linux eine Operation erfolgreich abgelaufen ist, kommt keine Erfolgsmeldung (*no news is good news*)

Wenn Du in Linux was falsch eingibst, kommt eine aussagekräftige Fehlermeldung.

Wenn Du in Linux defragmentieren willst, vergiss' es. Ich hab's auch noch nicht gefunden :-)

Wenn Du wo keinen Zugang am System hast, wird das schon seine Gründe haben. Du bist "user", nicht "root". Wenn was nicht auf Anhieb funktioniert, sieh' nach, ob Du überhaupt berechtigt bist, das zu benutzen.

Wenn Du gemütlich schlafen willst, keine schweren Systemeingriffe nach 22 Uhr.

Wenn Du um 1 Uhr 38 noch immer vor der Kiste hockst, hat Du den vorigen Tipp nicht beherzigt :-).

Schlussfolgerungen zu Linux

Linux ist nicht perfekt. Zeig' mir ein perfektes System und ich bin sofort dort :-). Aber die Vorteile in Linux sind einfach zu mannigfaltig, als dass ich sie negieren könnte. "Sicher, gratis.." ist der Grundtenor vieler Interessierter. Die Genialität von Linux verdeutlicht sich aber erst später, wenn man sich in die Materie eingearbeitet hat.

Der logische Aufbau und die Konzepte dahinter sind 40 Jahre alt. Und extrem ausgereift. Ich hab' 2001 auch meinen Microsoft-MCSA gemacht und hab' geglaubt, ich bin der King. Versuchte damals Linux zu installieren, geschaut wie ein Schwein in ein Schweizer Uhrwerk - nix verstanden, Windows gestartet. Glaub mir, 99 % der Umsteiger haben so angefangen. Ich auch.

Warum? Weil Du bisher nur Windows gekannt hast. Das ist eine der größten Hürden für den Anfang in Linux (als Systemspezi wohl gemerkt). Du musst unbefangen herangehen an das Thema. Alles andere verlängert nur den Lernprozess. Und Du musst es nur einmal lernen. Das ist das Gute dran. Das nur als kleiner Tipp am Rande.

Rückfragen werden selbstverständlich von mir nach bestem Wissen und Gewissen beantwortet.



Virtuelle Linux-Installation

Franz Fiala

Für mehrere Betriebssysteme benötigte man früher auch mehrere Geräte. Später machten es größere Festplatten möglich, mehrere Betriebssysteme auf verschiedenen Partitionen zu betreiben; es war aber immer ein Neustart des PC erforderlich, um das Betriebssystem zu wechseln. Die hohe Rechnerleistung moderner PCs erlaubt es nun, mehrere Betriebssysteme gleichzeitig auf einem Rechner mit Hilfe eines so genannten „Virtual PC“ ablaufen zu lassen. Man kann in eine bestehende Windows-Installation einen Virtuellen PC installieren und dort dann eine (oder mehrere) Linux-Distributionen installieren; und alles ohne auch nur einen Neustart. Die Box des Virtuellen PC verhält sich wie ein weiteres Windows-Programm. Die anderen Windows-Programme laufen ungehindert im Hintergrund weiter.

Das nachfolgende Beispiel zeigt die Schritte zum Download und Installation einer deutschen Ubuntu-Version in eine virtuelle Maschine von Windows 7:

Schritt 1: Download Ubuntu: http://wiki.ubuntuusers.de/Downloads/Karmic_Koala (Version 9.10)

Hier musste ich etwas experimentieren, denn auf meinem Rechner läuft ein 64-Bit-Betriebssystem Windows 7, doch die 64-Bit-Version von Ubuntu ließ sich nicht installieren. Dagegen hat die 32-Bit-Version funktioniert. Die ISO-Datei muss nicht auf eine CD gebrannt werden. Ubuntu (Gnome-Oberfläche) Kubuntu (KDE-Oberfläche), Xubuntu (Xfce-Oberfläche). Ubuntu installiert übrigens aus rechtlichen Gründen nicht alle Codecs, der User muss sie bei Bedarf nachinstallieren. Wer sich das nicht zutraut, sollte sich bei Mint anschauen: <http://www.linuxmint.de/>

Schritt 2: Danach installiert man einen Virtual-PC (Version 2007 SP1, 32- oder 64-Bit-Version wählen, ca. 30 MB) von <http://www.microsoft.com/downloads/details.aspx?FamilyId=28C97D22-6EB8-4A09-A7F7-F6C7A1F00B5&displaylang=en>

Nach der erfolgreichen Installation kann man zwischen "Virtuellen Computer erstellen" oder "Standardeinstellungen für das Erstellen eines virtuellen Computers verwenden" oder "Vorhandenen virtuellen Computer hinzufügen" wählen. Wir wählen die erste Option und geben zuerst den Namen für diesen virtuellen Computer an, zum Beispiel "Kubuntu910". Dann wählen wir als zu installierendes Betriebssystem "Andere", den Arbeitsspeicher belassen wir wie empfohlen, danach wählen wir eine "Neue virtuelle Festplatte". An einer Stelle wird man um die Größe des zu reservierenden Hauptspeichers und der Größe der virtuellen Festplatte gefragt. Beim Hauptspeicher habe ich 1 GB gewählt, Platz ist ja bei diesen 64-Bit-Betriebssystemen genug.

Die Handhabung der Maus erfordert unsere Aufmerksamkeit. Entweder bedient die Maus den virtuellen Computer oder den Desktop. Umschalten muss man mit der Taste Alt-Gr. Damit die Maus auch so funktioniert, müssen die Virtual Machine Additions (erreichbar über den Menüpunkt Aktion hinzugefügt werden).

Schritt 3: Am besten kopiert man jetzt die Kubuntu-ISO-Datei in den Ordner C:\Documents\Eigene virtuelle Computer\Kubuntu910, wo sich auch die Konfigurationsdateien und die virtuelle Festplatte befinden.

Schritt 4: Virtuellen PC starten (Bild 1), in der Virtual PC Konsole "Kubuntu910" wählen, auf *Starten* klicken, ein schwarzer Schirm meldet sich, am Ende der letzten Zeile rotiert ein Strich. Menü->CD->ISO-Abbild erfassen -> ISO-Datei wählen -> Öffnen. Menü->Aktion->Strg+Alt+Entf-> Kubuntu startet (Bild 2).

Alles Weitere ist so, als wäre der Rechner von der CD gestartet worden.

Die Installationsschritte sind:

- Sprache
- Zeitzone
- Tastaturlayout
- Festplatte
- Userinformationen (username, Passwort, Automatisches Login)
- Installation

Wenn man den virtuellen PC aufruft, kann man im Task-Manager einen deutlichen Sprung im Speicherbedarf sehen, je nachdem, wie viel Speicher man für das virtuelle Betriebssystem reserviert hat.

Das Ergebnis kann sich sehen lassen: ohne besondere Netzwerkconfiguration ist der Linux-Browser Konqueror sofort online; er übernimmt die Ressourcen des Host-Systems (Bild 3).

Man darf sich von dem virtuellen Ubuntu keine Geschwindigkeitsrekorde erwarten. Aber zum ersten Kennenlernen ist diese Installationsvariante bestens geeignet.

Wer auch erforschen will, wie Windows-Applikationen in Ubuntu ablaufen, kann das Programm Wine in Ubuntu installieren und dann ein Windows-Programm in der virtuellen Installation von Ubuntu ablaufen lassen. <http://www.winehq.org/download/deb>

Viel Spaß mit Linux!

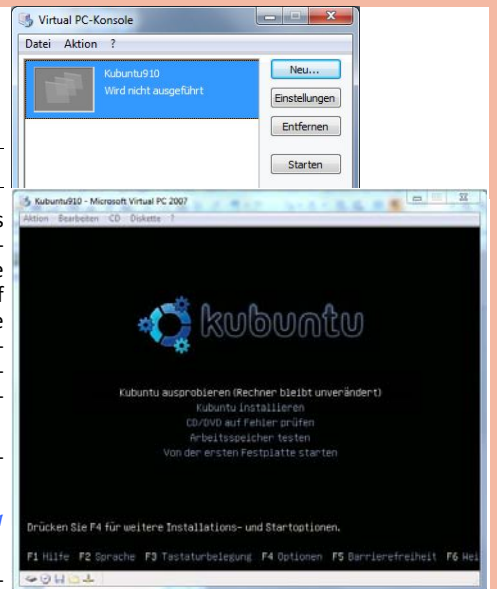
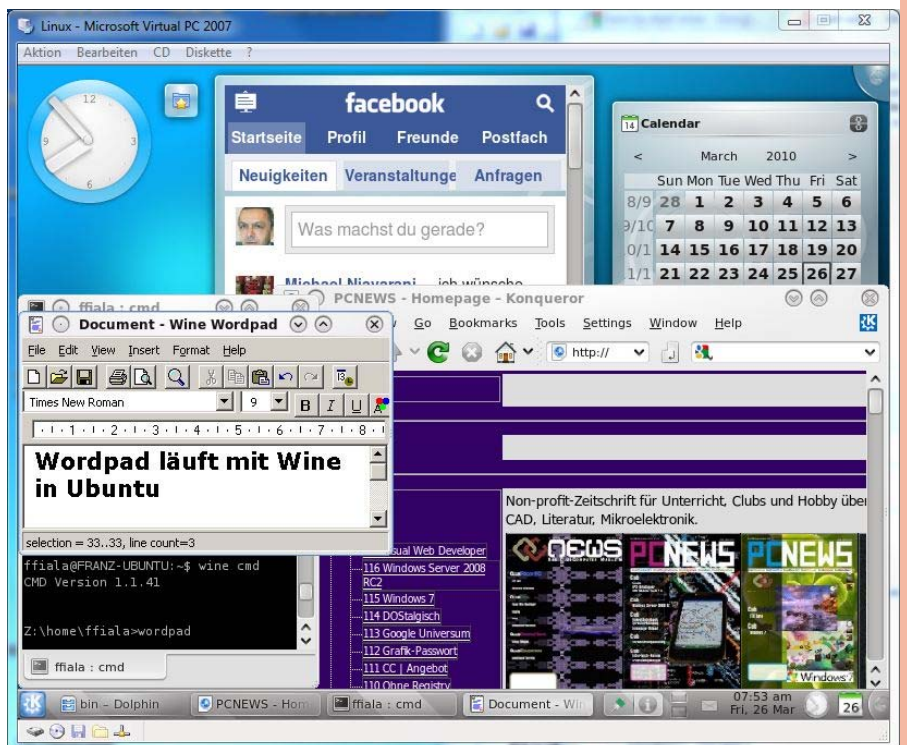


Bild 3: Windows 7 mit Virtual PC 2007.

Installiert ist das Programm Wine. Wine erlaubt die Verwendung (älterer) Windows-Programme unter Linux. Rechts unten: Browser „Konqueror“ zeigt die PCNEWS-Homepage. Links unten: Terminal-Fenster, Kommando Wine cmd (startet Dos-Konsole) Kommando Wordpad startet die Windows-Applikation Wordpad. Im Bild oben sieht man einige Widgets, links eine Analog-Uhr, in der Mitte ein Schnelleinstieg in Facebook und rechts einen Kalender.





Homepage-Download

Franz Fiala

Als Betreiber eines Webs hat man oft den Bedarf, den Besuchern Dateien zum Download anzubieten.

Dateien downzuloaden ist grundsätzlich leicht, man muss nur einen Html-Link auf die Datei setzen. Unangenehm ist dabei, dass man für jede weitere Datei eine Änderung am Html-Kode durchzuführen hat.

Es wäre einfacher, ein Downloadverzeichnis zu haben, in dem sich alle Dateien befinden, die ein User downloaden kann. Man kopiert diese Dateien einfach in ein Verzeichnis und zeigt dieses Verzeichnis an. Dem steht aber das Verhalten des Webservers entgegen.

Der Server sucht zunächst in dem gewünschten Verzeichnis nach einem Startdokument. Dieses ist im Allgemeinen **default.htm**, **.asp**, **.aspx**, **.php** oder auch **index.htm**... Der Server untersucht, ob ein solches Dokument existiert. Wenn ja, zeigt er den Inhalt dieses Dokuments an, wenn nein, könnte er den Inhalt des Verzeichnisses anzeigen. In der Grundeinstellung ist das aber verboten, daher liefert er eine Seite mir dem Text "Directory Listing Denied".

Als Besitzer eines Webs könnte man es aber über die Helm-Verwaltungsoberfläche erzwingen, dass der Inhalt des Verzeichnisses angezeigt wird, und dazu muss man das so genannte "Directory Browsing" aktivieren. Benutzer des Helm-Verwaltungssystems von ClubComputer klicken auf den Domännennamen, dann auf "Web Site Settings" -> "Directory browsing".

Diese Einstellung bezieht sich aber auf alle Verzeichnisse des Webs, d.h. auch Verzeichnisse, die eventuell gar nicht verlinkt sind und daher für einem Besucher gar nicht sichtbar sind, können durch direkte Angabe des Pfades - sofern sie diesen kennen oder erraten - diese Inhalte sehen.

Daher muss man in allen Verzeichnisse, deren Inhalt nicht angezeigt werden darf, ein Startdokument kopieren. Es kann leer sein, oder auch einen Text enthalten; zum Beispiel "Verzeichnis durchsuchen verboten".

Das erfordert aber einige Disziplin, denn immer, wenn man ein neues Verzeichnis anlegt, muss man sich daran erinnern, dass der Inhalt von einem Besucher eingesehen werden könnte (falls er den Verzeichnisnamen kennt) und muss daher ein Startdokument anlegen. Aber wenn das Web nicht großen Veränderungen unterworfen ist, mag diese einfachste Möglichkeit für den Download genügen.

Was allerdings enttäuschend ist, dass nämlich das Layout des Download-Verzeichnisses so gar nicht in das sonstige Layout des Webs passt.

Gesucht ist daher eine Lösung, die es erlaubt, den Inhalt eines Verzeichnisses anzuzeigen, vielleicht auch mit kleinen Einschränkungen (nur ZIP-Dateien) und die Ausgabe auch im Stile des Webs formatieren zu können. Es genügt dann, eine neue Datei in dieses Verzeichnis zu kopieren, ohne Notwendigkeit einer Änderung am Htmlcode. Eine solche Möglichkeit beschreibt das Programm **download.aspx**.

Download.aspx

Zur Lösung wird ein ASPX-Skript in C# angewendet. Damit kann man auf alle Klassen des .NET-Framework zugreifen. Als Entwicklungsumgebung kann man das kostenlose Visual Web Developer Express oder Visual Studio 2008 verwenden.

Damit diese Funktion in nur einer Datei verpackt werden kann, wird auf die *CodeBehind*-Technologie verzichtet. (Mit *CodeBehind* werden die Darstellung in der Sprache der Webserverelemente und der Kode für das Skript in zwei getrennten Dateien verwaltet.)

Voraussetzung

Um dieses Programm verwenden zu können, benötigt man einen *Microsoft Internet Information Server* mit installiertem Framework 2.0. Mitglieder bei ClubComputer müssen dazu lediglich in den *Web Site Settings* von Helm die Option *DotNet* aktivieren. Das Programm besteht aus der einzigen Datei **down.aspx**.

Anwendung

Man kopiert diese Datei **down.aspx** in jenen Ordner, dessen Dateien anzuzeigen sind und ruft dann - zum Beispiel vom Inhaltsverzeichnis diese Datei auf.

Ein Klick auf den Dateinamen öffnet diese Datei (Html- oder Bilddateien) oder bietet diese Datei zum Download an (Archivdateien und alle Dateien, mit denen ein Webbrowser eben nichts anfangen kann). Wenn Dateien nicht angezeigt werden sollen, muss man sie vorher

Dateiname	Länge	Datum
PageGoogleSearchSites.htm	7786	2009-03-26 09:37
RapidChart.aspx	659	2009-03-26 09:39
RapidTopografie.htm	266999	2009-06-14 18:43
ShowKml.htm	9754	2009-06-08 18:43
Statistics.aspx	7359	2009-03-26 09:37
Upload.aspx	1661	2009-07-21 07:33

archivieren (als Zip-Datei speichern). Die Tabellenüberschriften sortieren nach diesem Kriterium.

Parametrierung

Kleine Änderung kann man selbst vornehmen.

Anzuzeigende Dateien

```
string[] strFilter = { "*.gif", "*.jpg" };
```

Hier gibt man mit Wildcards jene Dateien an, die man anzeigen will. In der Grundeinstellung werden alle Bilder (JPG- und GIF-Dateien) angezeigt. Es gelten die Konventionen der Kommandozeile. **{*.*}** zeigt alle Dateien an, **{*.txt}** zeigt alle Textdateien an, **{*.zip, *.htm, *.txt}** zeigt Archive, Html-Dateien und Textdateien an.

Nur aktuelles Verzeichnis oder auch Unterordner

```
const bool AlleVerzeichnisse = false;
```

In der Grundeinstellung werden nur die Dateien des aktuellen Ordners angezeigt; mit **AlleVerzeichnisse = true;** werden auch die Dateien von Unterordnern angezeigt.

Längenangabe anzeigen

```
const bool ShowLength = false;
```

Man kann die Anzeige der Dateilänge ein- (**true**) oder ausschalten (**false**).

Datum und Uhrzeit anzeigen

```
const bool ShowDateTime = true;
```

Man kann die Anzeige des Dateidatums ein- (**true**) oder ausschalten (**false**).

Seitentitel

```
Title = Request.ApplicationPath;
```

Diese Grundeinstellung zeigt in der Titelzeile des Browsers den Pfad des Verzeichnisses an. An dieser Stelle kann man auch einen anderen Wert einsetzen, zum Beispiel **Title = "Meine Dateien zum Download";**.

Formatierung

```
<style type="text/css">
    .Table
    {
        border-width: 1;
        border-color: Teal;
        font-family: Verdana;
        background-color: #EEEEEE;
    }
    .Laenge
    {
        background-color: #EEFFEE;
    }
</style>
```

Das eingebettete Stylesheet ist ein Muster. Das Aussehen der Tabelle definiert man in **.Table** und wenn eine Spalte anders aussehen soll, dann sieht man das hier am Beispiel der Länge. Soll auch das Datum anders formatiert sein, muss man zunächst einen neuen Abschnitt im Stylesheet einfügen und danach in der entsprechenden Spalte der Tabelle den Wert **CssClass** setzen.

Weitere Seitenelemente

Wenn diese Seite weitere Elemente haben soll (zum Beispiel einen Button zur Rückkehr zum Inhaltsverzeichnis, dann man diesen Html-Kode an einer geeigneten Stelle zwischen **<body>** und **</body>** einsetzen.

Vorschlag für den Einbau eigener Bedienungselemente

Den Abschnitt **<div><asp:GridView>...** kann man auch zum Beispiel so formatieren:

```
<table>
<tr>
<td>
    Hier kommt der eigene Text,
```



```

        das eigene Bedienungselement hinein
    </td>
</tr>
<tr>
    <td>
        <!-- Kode des GridView aus der Datei down.aspx -->
        <asp:GridView>....</asp:GridView>
    </td>
</tr>
</table>

```

Funktion

Programmcode und ASP.NET-Kode sind in einer gemeinsamen Datei. Die Formatierung wird in einem Style-Sheet zusammengefasst. Beim Aufruf der Datei wird der Programmteil `Page_Load()` abgearbeitet. Beim Sortieren einer Spalte wird die Funktion `GridView_Directory_Sorting()` aufgerufen.

Beim ersten Seitenaufruf `if (!IsPostBack)` wird das Inhaltsverzeichnis in die `DataTable dt` gelesen. Diese Tabelle besteht aus den Spalten `Dateiname`, `Dateilink`, `Länge`, `Datum`. Das Lesen des Verzeichnisses erfolgt in so vielen Durchläufen als es Filter in der Variablen `strFilter` gibt.

`Path` ist der physikalische Pfad zu dem Verzeichnis. `MyName` ist der Name der Datei (`down.aspx`). Man braucht diesen Namen, damit diese Datei nicht mit angezeigt wird. Wenn daher dieser Dateiname geändert wird, muss das Programm nicht verändert werden.

`fi` enthält Dateiinformationen zu allen Dateien in dem Verzeichnis, die dem jeweiligen Filter entsprechen.

Die vollständig aufgebaute Tabelle `dt` wird als Sessionvariable gespeichert und steht dort bei weiteren Aufrufen derselben Seite wieder zur Verfügung und daher muss die Tabelle nicht wieder aufgebaut werden (`if (!IsPostBack)`).

Die Funktion `DataGrid_DataBind()` holt die Tabelle `dt` aus der Sitzung und erzeugt daraus einen sortierbaren `DataView dv`. Spalten, die nicht angezeigt werden sollen, werden ausgeblendet. Der Vorteil des `DataView` ist die Sortierbarkeit. Das Attribut `EnableSorting` des `WebServerControls GridView` macht aus den Tabellenüberschriften einen Link. Diese Links aktivieren ein Postback dieser Seite, und rufen die Eventfunktion `GridView_Directory_Sorting()` auf.

Der Html-Teil, das Sichtbare an dem Programm, besteht nur aus einem `GridView` mit vordefinierten Spalten. Durch die Verbindung der Daten aus der Datentabelle mit dem `GridView` generiert das Serverprogramm den korrekten Html-Kode für den Browser.

Programmlisting

```

<%@ Page Language="C#" AutoEventWireup="True" %>
<%@ Import Namespace="System" %>
<%@ Import Namespace="System.Data" %>
<%@ Import Namespace="System.Web.UI.WebControls" %>
<%@ Import Namespace="System.IO" %>
<%@ Import Namespace="System.Collections" %>
<!DOCTYPE html PUBLIC "-//W3C/DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">

<script language="C#" runat="server">
    string[] strFilter = { "*.gif", "*.jpg" };
    const bool AlleVerzeichnisse = false;
    const bool ShowLength = false;
    const bool ShowDateTime = true;
    void Page_Load(object sender, EventArgs e)
    {
        if (!IsPostBack)
        {
            Title = Request.ApplicationPath;
            string Path =
                Server.MapPath(Request.CurrentExecutionFilePath);
            string MyName = Path.Substring(Path.LastIndexOf("\\") + 1);

            // Dateinamen entfernen
            Path = Path.Substring(0, Path.LastIndexOf("\\"));
            DirectoryInfo di = new DirectoryInfo(Path);
            DataTable dt = new DataTable();
            dt.Columns.Add("Dateiname", Type.GetType("System.String"));
            dt.Columns.Add("Dateilink", Type.GetType("System.String"));
            dt.Columns.Add("Länge", Type.GetType("System.Int64"));
            dt.Columns.Add("Datum", Type.GetType("System.String"));
            for (int Pattern = 0; Pattern < strFilter.Length; Pattern++)
            {
                FileInfo[] fi;
                if (AlleVerzeichnisse)
                    fi = di.GetFiles(
                        strFilter[Pattern], SearchOption.AllDirectories);
                else
                    fi = di.GetFiles(
                        strFilter[Pattern], SearchOption.TopDirectoryOnly);
                for (int i = 0; i < fi.Length; i++)
                {
                    if (fi[i].Name.ToLower() != MyName.ToLower())

```

```

                string aPath = Path.ToLower();
                aPath = fi[i].Directory.ToString();
                aPath = sPath.ToLower().Replace(aPath, "");
                if (aPath.Length > 0)
                {
                    if (aPath[0] == '\\')
                    {
                        aPath = aPath.Substring(1);
                        if (aPath.Length > 0)
                            aPath += "/";
                    }
                }
                DataRow dr = dt.NewRow();
                dr["Dateiname"] = aPath + fi[i].Name;
                dr["Dateilink"] = aPath + fi[i].Name;
                dr["Länge"] = fi[i].Length;
                dr["Datum"] = fi[i].CreationTime.ToString("u");
                dr["Datum"] = dr["Datum"].ToString();
                int Length = dr["Datum"].Length;
                dr["Datum"] = dr["Datum"].Substring(0, Length-4);
                dt.Rows.Add(dr);
            }
        }
    }
    Session["Directory_dt"] = dt;
    Session["Directory_SortDir"] = 1;
    DataGrid_DataBind("Dateiname");
}

void DataGrid_DataBind(string SortExpression)
{
    DataTable dt = (DataTable)Session["Directory_dt"];
    DataView dv = new DataView(dt);
    int SortDir = (int)Session["Directory_SortDir"];
    dv.Sort = SortExpression + " " + (SortDir == 1 ? "ASC" : "DESC");
    Session["Directory_SortDir"] = SortDir * -1;
    GridView_Directory.DataSource = dv;
    GridView_Directory.Columns[1].Visible = ShowLength;
    GridView_Directory.Columns[2].Visible = ShowDateTime;
    GridView_Directory.DataBind();
}

void GridView_Directory_Sorting(object sender,
GridViewSortEventArgs e)
{
    DataGrid_DataBind(e.SortExpression);
}
</script>
<head id="Head1" runat="server">
    <title></title>
    <style type="text/css">
        .Table
        {
            border-width: 1;
            border-color: Teal;
            font-family: Verdana;
            background-color: #EEEEEE;
        }
        .Laenge
        {
            background-color: #EEFFEE;
        }
    </style>
</head>
<body>
    <form id="form1" runat="server">
    <div>
        <asp:GridView runat="server" ID="GridView_Directory"
            AutoGenerateColumns="False"
            AllowSorting="True" OnSorting="GridView_Directory_Sorting"
            CssClass="Table">
            <Columns>
                <asp:HyperLinkField DataNavigateUrlFields="Dateilink"
                    DataTextField="Dateiname" HeaderText="Dateiname"
                    SortExpression="Dateiname">
                    <ItemStyle HorizontalAlign="Left" />
                </asp:HyperLinkField>
                <asp:BoundField DataField="Länge" HeaderText="Länge"
                    SortExpression="Länge">
                    <ItemStyle HorizontalAlign="Right" />
                    <ItemStyle CssClass="Laenge" />
                </asp:BoundField>
                <asp:BoundField DataField="Datum" HeaderText="Datum"
                    SortExpression="Datum">
                    <ItemStyle HorizontalAlign="Right" />
                </asp:BoundField>
            </Columns>
        </asp:GridView>
    </div>
</form>
</body>
</html>

```

Download http://pcnews.at/pcn/1xx/11x/118/001900/_prg
 Demo <http://fiala.member.pcc.ac/>

Applikationsentwicklung für Windows 7

Thomas Reinwart

Mit der aktuellen Windows Version 7 stellt sich für Entwickler und Administratoren die Frage, welche der bisher eingesetzten Applikationen kompatibel sind. Den Entwickler interessiert, was am bestehenden Code zu ändern ist und was nicht (mehr) erlaubt ist. Den Admin interessiert, warum die Applikation unbedingt unter Admin-Rechten laufen muss. Aber nicht nur das, warum und wann muss die Anwendung bei Windows 7 im XP Mode laufen.

Ein weiterer Aspekt sind die neuen Features des Betriebssystems, die für Windows 7 Applikationen genutzt werden können.

Windows 7 Features für Entwickler

- Jumplists (Sprunglisten)
- Anwendungssymbole, Taskbar Button, Fortschrittsanzeigen
- Ribbon – Multifunktionsleisten
- Multitouch
- Windows Shell (API Code Pack)
- API für Sensor und Location (Licht und Bewegungssensoren)
- Windows 7 SDK (Code Samples)

UAC

UAC steht für *User Account Control* (Benutzerkontensteuerung) und wurde mit Windows Vista eingeführt. Es handelt sich um einen Berechtigungsmechanismus, der bei Vista als häufiges Popup Fenster (elevation) dargestellt wurde. Im Popup Fenster wurde man informiert, dass die gestartete Applikation Admin-Rechte benötigt, um gestartet oder weiterarbeiten zu können. Beispiel bei Installation, Update Prüfung oder beim Speichern von Daten. Gleiches Verhalten ergab sich beim Starten der Systemsteuerung oder Änderungen an der Konfiguration des Rechners. Viele Applikation haben sich lange nicht an die Security Design Vorschriften für Vista Applikationen gehalten, dadurch haben viele Anwendungen Admin Rechte zum Starten benötigt. Die dadurch entstandene Meldungshäufigkeit der Popups war ein Kritikpunkt bei Vista. Bei Windows 7 kann die Meldungshäufigkeit der Popups definiert werden. Die „Always notify“ Einstellung in der Systemsteuerung von Windows 7 entspricht dem Verhalten von Windows Vista. Nichts desto Trotz sollte es nun an der Zeit sein, seine Anwendung Windows 7 (bzw. eigentlich schon Windows Vista) tauglich zu machen.

Sinn von UAC

Sinn von UAC ist es, die Angriffsmöglichkeiten auf das Betriebssystem zu reduzieren. Die Anwendung wird per default unter dem Standard Benutzer gestartet.

Applikationen die nicht den UAC-Security-Vorschriften entsprechen, müssen immer unter Admin-Rechten laufen. Das hat den Nachteil, wenn es Sicherheitslücken in der Applikation gibt oder schadhafter Code eingeschleust wird, wird dieser unter den Admin Rechten ausgeführt und kann mehr Schaden anrichten als unter den Rechten eines Standard Benutzers. Eine Einladung für Angreifer.

Rechte eines Standard Users (ab Windows Vista):

- Ändern der Zeitzone
- WLAN Verbindung herstellen (WEP - Wired Equivalent Privacy installieren)
- VPN Verbindung herstellen
- Ändern der Energiesparoption
- Ändern der Anzeigeeinstellungen
- Schriftarten installieren
- Drucker installieren
- Installieren von Updates (Installer muss UAC kompatibel sein)

Man erkennt, dass ein Arbeiten auch als Nicht-Admin selbst bei Reisetätigkeit möglich ist.

Windows 7 UAC Einstellungen

Always notify



Always notify me when:

- Programs try to install software or make changes to my computer
- I make changes to Windows settings

i Recommended if you routinely install new software and visit unfamiliar websites

Never notify

Always notify



Default - Notify me only when programs try to make changes to my computer

- Don't notify me when I make changes to Windows settings

i Recommended if you use familiar programs and visit familiar websites

Never notify

Always notify



Notify me only when programs try to make changes to my computer (do not dim my desktop)

- Don't notify me when I make changes to Windows settings

i Can be selected if you use familiar programs and visit familiar websites

Not dimming the desktop might allow programs to interfere with the User Account Control prompt

Never notify

Always notify



Never notify me when:

- Programs try to install software or make changes to my computer
- I make changes to Windows settings

i Not recommended but can be selected if you use programs that are not certified for Windows 7 because they do not support User Account Control

Never notify

Least privileges

Dieser Ausdruck definiert den minimalen Rechtebedarf einer Applikation. Je weniger Rechte, umso weniger Möglichkeiten in das Betriebssystem einzugreifen. Beispiel sind angehängte Viren an der ausgeführten Anwendung. D.h. der Entwickler muss etwa dafür sorgen, seine Files an den erlaubten Stellen abzulegen und nicht etwa eine temporäre Datei im Verzeichnis `Windows\System32` anzulegen, was Admin-Rechte beim Ausführen der Anwendung benötigt. Ein weiteres Beispiel ist das Starten eines weiteren Prozesses. Unter dem Prinzip von *least privileges* wird eine Anwendung im vornhinein mit den Rechten des Standarduser gestartet und nicht mit Admin-Rechten.

Welche Bereiche der Applikation müssen UAC fit gemacht werden?

Die meisten Anwendungen benötigen noch immer Zugriff auf Bereiche in Verzeichnisse und die Registry, die ab Vista für den Standardbenutzer gesperrt wurden. Also z.B. die Daten direkt in **program files** anstatt diese in **CommonApplicationData** zu speichern.

An der Applikation sind also jene Bereiche anzupassen, die nur mit Admin-Rechten zu erreichen sind. Man muss sich als Entwickler die Frage stellen, ob Files nicht an anderer Stelle gespeichert werden können.

Unterstützung mit Microsoft Tools

Mittels *Application Verifier* erhält man eine XML-Log-Datei der *issues*. Mit dem *standard user analyzer* (SUA) lassen sich die UAC-relevanten Zugriffe der Applikation erkennen. Diese sind:

- *File Access*
- *Registry Access*
- *Config Files* (ini)
- *Token issues* (Schlüssel)
- *Security privileges* (Sicherheitseinstellungen)
- *Name space issues* (Namensraum-Probleme)

File Access

Irgendwie wild ins Dateiverzeichnis schreiben ist gar nicht gut. (Jeder Admin wird hellhörig).

Nicht erlaubt: **root**, **windows**, **windows32**, **program files**

Erlaubte Ablagemöglichkeiten (.net C# Code)

```
Environment.GetFolderPath
(Environment.SpecialFolder.ApplicationData);
```

Name	Beschreibung
ApplicationData	Verzeichnis, das als Repository für anwendungsspezifische Daten des aktuellen roaming Users genutzt wird.
CommonApplicationData	Gibt den Pfad zum Ordner für Anwendungsdaten aller Benutzer zurück
LocalApplicationData	Verzeichnis, das als Repository für anwendungsspezifische Daten des aktuellen (nicht roaming) Users genutzt wird.
CommonProgramFiles	Gemeinsames Verzeichnis für Komponenten die in unterschiedlichen Anwendungen genutzt werden

Registry Access

Auf **HKCR** (*classes root*)- und **HKLM** (*local machine*)-Verzeichnisse in der Registry kann ab Windows Vista nur mit Admin-Rechten zugegriffen werden. Wenn es Anforderung der Anwendung ist, dass eine Funktion in sensiblen Bereiche agiert, muss die Anwendung entweder mit Admin-Rechten gestartet werden (**runas Admin**), oder im Manifest-File ist definiert, dass Admin-Rechte notwendig sind (Prompt erscheint).

In die Registry gehört mit .net Anwendungen ohnehin nichts mehr rein, dafür gibt es die config files (xml files).

Code Samples

Angenommen meine Anwendung muss auf einen Bereich zugreifen, der Admin-Rechte benötigt. Wie kann ich das bewerkstelligen? Innerhalb des mitgelieferten Manifest-Files wird festgelegt, mit welchen Rechten die Applikation gestartet wird. Nun will ich das aber nicht immer, sondern nur bei einer bestimmten Funktion, die hinter einem Button liegt. Das geht so:

Wurde die Anwendung mit Admin-Rechten gestartet?

```
public static bool IsAdmin()
{
    WindowsIdentity id = WindowsIdentity.GetCurrent();
    WindowsPrincipal p = new WindowsPrincipal(id);
    return p.IsInRole(WindowsBuiltInRole.Administrator);
}
```

Nach dem Drücken des UAC-Buttons erscheint der Elevation Dialog.

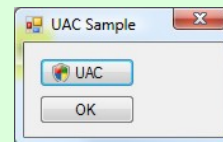
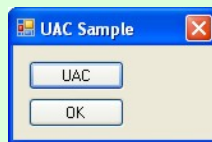
Verwendung der Klasse

```
UACShield.SetShieldOnButton(this.buttonUAC, true);
```

Klasse

```
public class UACShield
```

Admin Logo auf Button



Unter Windows XP (kein UAC vorhanden) ausgeführt – es ist kein UAC-Schild vorhanden
 Unter Windows Vista / 7 ausgeführt – das UAC-Schild ist erkennbar

```
{
    [DllImport("shell32.dll", EntryPoint = "ExtractIconExW",
    CallingConvention = CallingConvention.StdCall)]
    public static extern int ExtractIconEx(
        [In][MarshalAs(UnmanagedType.LPWSTR)]string lpzFile,
        int nIndex, ref IntPtr phiconLarge, ref IntPtr phiconSmall,
        int nIcons);
    private static Image GetVistaShield(bool smallIcon)
    {
        int icoIdx = 6; Image img = null;
        Icon icon = null; IntPtr iconLargePtr = IntPtr.Zero;
        IntPtr iconSmallPtr = IntPtr.Zero;
        string path =
            Path.Combine(
                Environment.GetFolderPath(Environment.SpecialFolder.System),
                "user32.dll");
        try
        {
            int result =
                ExtractIconEx(path, icoIdx, ref iconLargePtr, ref iconSmallPtr,
                1);
            if (result < 1)
                return null;
            if (smallIcon && iconSmallPtr != IntPtr.Zero)
            {
                icon = Icon.FromHandle(iconSmallPtr);
            }
            else if (iconLargePtr != IntPtr.Zero)
            {
                icon = Icon.FromHandle(iconLargePtr);
            }
            else
                return null;
            img = (Image)icon.ToBitmap();
        }
        catch (Win32Exception)
        {
            return null;
        }
        finally
        {
            iconLargePtr = IntPtr.Zero; iconSmallPtr = IntPtr.Zero;
        }
        return img;
    }
    public static Image SmallVistaShield
    {
        get { return GetVistaShield(true); }
    }
    public static Image LargeVistaShield
    {
        get { return GetVistaShield(false); }
    }
    public static void SetShieldOnButton(Control ctrl, bool smallIcon)
    {
        Button btn = (Button)ctrl;
        if (smallIcon)
        {
            btn.Image = SmallVistaShield;
        }
        else
        {
            btn.Image = LargeVistaShield;
        }
        btn.TextImageRelation = TextImageRelation.ImageBeforeText;
    }
}
```

Elevation PopUp

Eine Anwendung meldet sich dann (*Elevation PopUp*), wenn diese beim ersten Start höhere Privilegien benötigt oder wenn die Anwendung explizit mit **runas administrator** gestartet wird. Die Anwendung meldet sich einmalig, um diese höheren Rechte zu erhalten. Im Hintergrund wird der alte Prozess (der mit den unzureichenden Rechten) beendet und mit den gewünschten Rechten neu gestartet. Die einzige Ausnahme des *Elevation PopUps* ist dann, wenn eine Anwendung von einem bestehenden Prozess gestartet wird. Hier erhält der zweite Prozess die Rechte des ersten vererbt und es erfolgt dabei kein *Elevation PopUp*. Per Default hat der Elevation Dialog eine orange Farbe, wenn der Code signiert ist erhält er eine blaue Farbe.

Prozess

Beim Drücken des Knopfes soll ein neuer Prozess gestartet werden:

```
Process p = new Process();
p.StartInfo.UseShellExecute = true;
p.StartInfo.Verb = "runas";
p.StartInfo.FileName = @"c:\Programme\Paint.NET\PaintDotNet.exe";
p.Start();
```

Manifest File

Das Manifest-File einer .net Anwendung legt den *UAC access level* beim Starten fest. Es ist ein XML-*resource file* und kann embedded sein. Die

Name	Beschreibung
asInvoker	Benötigt keine „ <i>elevation</i> “ (es kommt kein Dialog), hat die Rechte des Mutter Prozesses.
highestAvailable	Dialog für Standard-Benutzer das Admin-Rechte zum benötigt werden. Verlangt nach den höchst möglichen Rechten des des Mutter-Prozesses. Von einem Administrator wird der volle Admin-Level verlangt.
requireAdministrator	Dialog für Standard Benutzer das volle Admin-Rechte notwendig sind.

Privilegien werden im Abschnitt **requestedExecutionLevel** festgelegt.

Beispiel Manifest file

```
<?xml version="1.0" encoding="utf-8"?>
<asmv1:assembly manifestVersion="1.0" xmlns="urn:schemas-microsoft-com:asm.v1" xmlns:asmv1="urn:schemas-microsoft-com:asm.v1" xmlns:asmv2="urn:schemas-microsoft-com:asm.v2" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <assemblyIdentity version="1.0.0.0" name="MyApplication.app"/>
  <trustInfo xmlns="urn:schemas-microsoft-com:asm.v2">
    <security>
      <requestedPrivileges xmlns="urn:schemas-microsoft-com:asm.v3">
        <!-- UAC Manifest Options -->
        If you want to change the Windows User Account Control level replace the requestedExecutionLevel node with one of the following.
        <requestedExecutionLevel level="asInvoker" uiAccess="false" />
        <requestedExecutionLevel level="requireAdministrator" uiAccess="false" />
        <requestedExecutionLevel level="highestAvailable" uiAccess="false" />
        If you want to utilize File and Registry Virtualization for backward compatibility then delete the requestedExecutionLevel node.
        >
          <requestedExecutionLevel level="asInvoker" uiAccess="false" />
        </requestedPrivileges>
      </security>
    </trustInfo>
  </asmv1:assembly>
```

Für die meisten Anwendungen reicht der Level asInvoker, der nicht jedes Mal den *Elevation Dialog* bringt wenn die Anwendung gestartet wird und die Standard Privilegien verwendet für den Zugriff auf die Ressourcen. Die anderen Applikationen, die diesen Zugriff beabsichtigen, benötigen einen höheren Zugriffslevel konfiguriert.

uiAccess: legt fest, dass die Anwendung Zugriff auf das User Interface (UI) benötigt, Bsp eine System Dialog Box. Funktioniert nur mit signierten Assemblies. Der Default Wert von uiAccess ist false.

Manifest Tool (mt.exe)

Ist ein Command Line Tool zum Erstellen von signierten Files, als Übergabeparameter gibt man das Manifest Files und die Assembly an. Das Manifest Tool erzeugt Hashes (mittels der Crypto API einen SHA-1)

Beispielaufruf

```
mt.exe -manifest "MyApp.exe.manifest" -outputresource:"MyApp.exe"
mt.exe -manifest "MyApp.dll.manifest" -outputresource:"MyApp.dll"
```

Deaktivieren von UAC

UAC ist per default in Windows Vista und Windows 7 aktiviert. Deaktivieren lässt es sich weiterhin, sinnvoll ist es nicht. Sinnvoll ist es, die Anwendungen anzupassen.

Windows Vista

http://www.administrator.de/Windows_Vista_-_UAC_deaktivieren.html

Windows 7

<http://www.tobbis-blog.de/microsoft/anleitungen/2009-02-11-uac-unter-windows-7-deaktivieren/>

Google-Charts

Franz Fiala

Benötigt man ein Diagramm, welches in Webseiten eingebettet werden soll und will man dafür nicht programmieren, eignen sich die Google-Charts bestens dafür.

Beispiel

Datenfolge

3.3, 4, 6.8, 8.8, 12, 12.5, 14, 17, 18.5, 21.4

Um diese Daten grafisch in einem Liniendiagramm anzuzeigen, kann man entweder—wie im folgenden Artikel gezeigt wird—viel Arbeit in Grafikformate investieren — oder einfach die Google-Charts nutzen.

Aufruf in der eigenen Html-Datei

```
<html><body>

</body></html>
```

Ergebnis



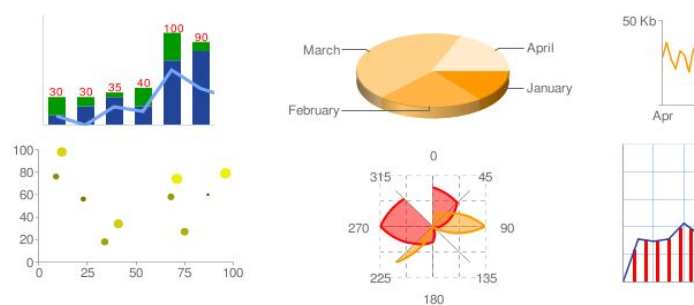
Parameter

cht	Digrammtyp (chart type)
lc	Liniendiagramm (line chart)
chs	Breite x Höhe
chds	Achsenbeginn und Achsenende
chd	Darzustellende Daten
t:	simples Textformat

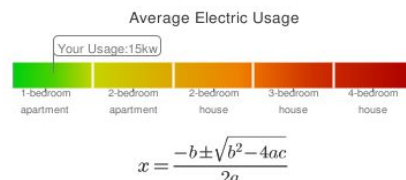
Anleitung

<http://chart.apis.google.com/>

Some things that look like charts...



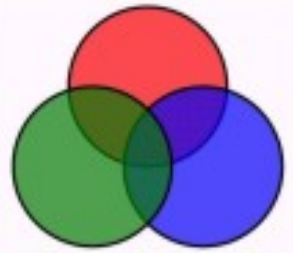
And some that don't...





Vektorgrafik in Webseiten

Franz Fiala



Eine Reihe von Charts, die aus einer Datenbank generiert werden, sollen sowohl im Internet publiziert werden als auch mit CorelDraw für den Druck aufbereitet werden. Nun sind die Ansprüche für eine Internet-Darstellung und den Druck sehr verschieden und man könnte der Meinung sein, dass man dazu eben zwei verschiedene Programme benötigt. Doch im Sinne einer wirklich gut übereinstimmenden Darstellung von Webversion und Druckversion, sollte man von einer einheitlichen Quelle ausgehen. Gesucht war daher ein Format, das eine portierbare Vektor-Grafik erzeugt, die sowohl im Internet als auch im Druck zum Einsatz kommen kann.

Das Problem der Grafik-Darstellung im Internet ist die Herstellung und danach die Aktualisierung der Bilder. Wenn es sich zum Beispiel um Messwerte handelt, dann passen nach kurzer Zeit die Messwerte nicht mehr zu den Bildern. Jedes Chart, das in Excel generiert wurde, muss durch einen Screenshot in ein Bild verwandelt werden, um in einer Internet-Seite präsentiert werden zu können. Bei wenigen Grafiken ist diese Handarbeit kein Problem, es wird aber mühsam, wenn sich die Daten oft verändern und unmöglich, wenn es sich um Hunderte Grafiken handelt; ein Ausflug in die Welt der Grafik-Formate lohnt sich.

Man kann sich mit Formaten wie Adobe Flash behelfen und diese Formate dynamisch parametrieren. Dabei verlässt man aber die Welt des klassischen HTML und verwendet proprietäre Formate, die nicht in jeder Umgebung verfügbar sind.

Dieser Beitrag zeigt, wie man in Web-Seiten portable Vektor-Grafik integrieren kann und wie man daraus dynamische Pixel-Grafiken erstellt.

SVG

Das gewünschte portierbare Format gibt es, es ist das Format *Scalable Vector Graphics* SVG und ist bei W3C normiert. Nach einigen Versuchen erweist sich das Format als sehr universell. Es ist in der Lage, sowohl Vektorgrafik, (auch Fonts) als auch Bitmaps darzustellen. Es wird von den Programmen Adobe Illustrator, Adobe InDesign, CorelDraw und vielen anderen als Export- und Import-Format akzeptiert. Browser interpretieren SVG mehr oder weniger gut bis gar nicht. Getestet wurden Firefox, Opera, Safari, Chrome und Internet-Explorer. Beim IE benötigt man zur Darstellung ein Plug-In. Die bestgeeigneten Browser für SVG-Darstellung ist Chrome oder Firefox.

Aber auch wenn man dieses Plug-In (von Adobe) am IE installiert, ist man über die mangelhafte Darstellung enttäuscht. Dazu kommt, dass damit praktisch alle Installationen des Internet-Explorers in Firmennetzen, bei denen die Installation von Plug-Ins verhindert wird, die allgemeine Verwendung des SVG-Format derzeit blockieren. Fast überflüssig zu erwähnen, dass auch eine Excel-Grafik nicht im SVG-Format exportiert werden kann. Kurz, die direkte Darstellung von SVG-Grafiken ist stark browserabhängig.

Und die Lösung?

Um daher eine wirklich portable Lösung zu erhalten, muss man die SVG-Datei zur Darstellung am Browser in ein JPG-Bild verwandeln. Diese Datei können dann natürlich alle Browser darstellen. Nicht mehr ganz so dynamisch, denn die Umwandlung in eine Grafik dauert auch einige Zeit. Mit dieser Umwandlung kann sich auch der bockigste Browser "brausen". Wir werden sehen, wie das im Einzelnen geschieht. Die SVG-Datei dient zur Publikation für den Druck. Da es sich um Vektor-Grafik handelt, kann jedes einzelne Element mit Adobe Illustrator oder CorelDraw nachbearbeiten (Farben, Positionen und Liniendicken nachjustieren usw.).

Der folgende Beitrag zeigt die wesentlichen Elemente, die zur Herstellung portabler und dynamisch generierter Vektorgrafik notwendig sind. Alle Beispiele und alle Hilfsprogramme und Bibliotheken stehen bei der Webversion dieses Artikels zur Verfügung:

http://pcnews.at/pcn/1xx/11x/118/002400/_prg/

Demo <http://fiala.member.pcc.ac/>

Grundlegendes zum SVG-Format

Scalable Vector Graphics ist eine XML-Beschreibungssprache und daher – anders als HTML – nicht fehlertolerant. Sie erzeugt bei Verstößen gegen die Syntax-Regeln Fehlermeldungen. Die Schreibweise der Tags und Attribute entspricht den strengen Regeln von XML. Für den Anwender bedeutet das praktisch ein Aus für händische Code-Generierung (außer bei

einfachen Beispielen), weil der Interpret fehlerbehaftete Codes ignoriert und größere Verstöße mit Fehlermeldungen begleitet.

Beispiel für eine SVG-Datei (Beispiel1.svg)

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"
"http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">
<svg xmlns="http://www.w3.org/2000/svg"
width="800" height="800">
<g style="fill-opacity:0.7; stroke:black; width:0.1cm;">
<circle cx="200" cy="100" r="100"
style="fill:red;" transform="translate(0,50)" />
<circle cx="200" cy="100" r="100"
style="fill:blue;" transform="translate(70,150)" />
<circle cx="200" cy="100" r="100"
style="fill:green;" transform="translate(-70,150)"/>
</g>
</svg>
```

Das `svg`-Tag definiert ein Bild mit den Abmessungen `width` und `height`. Das `g`-Tag bildet eine Gruppierung und weist dieser Gruppe gemeinsame Eigenschaften, hier Transparenz (`opacity`), Zeichenfarbe (`stroke`) und Zeichenbreite (`width`). Die dargestellten Grafikelemente sind drei überlappende Kreise mit den Mittelpunkten `cx` und `cy`, den Radien `r` und verschiedenen Füllfarben `fill`. Damit sich die gleich großen Kreise überlappen, werden sie durch das Attribut `translate` verschoben.

Die SVG-Datei `Beispiel1.svg` kann mit jedem Browser (außer mit dem Internet-Explorer) aufgerufen werden. Achtung: Der Unterschied zu einem Bild ist der, dass man diese Grafik nicht über das Kontext-Menü speichern kann, so, wie man das mit einem Bild gewohnt ist, denn es ist eben kein Bild. Einen Browser-Test, der auch die SVG-Kompatibilität in die Prüfung einbezieht, findet man hier: <http://acid3.acidtests.org/>.

Spezifikation

<http://de.wikipedia.org/wiki/SVG>

<http://www.w3.org/Graphics/SVG/>

<http://www.w3.org/TR/SVG11/>

Lehrgänge

<http://luxor-xul.sourceforge.net/talk/jug-nov-2002/slides.html>

<http://www.w3.org/TR/2007/WD-SVGPrintPrimer12-20071221/>

<http://www.w3.org/TR/2007/WD-SVGPrint12-20071221/>

<http://svg.tutorial.aptico.de/start.php>

<http://www.w3schools.com/svg/>

<http://tutorials.jenkov.com/svg/>

<http://www.adobe.com/svg/basics/intro.html>

<http://www.datenverdrahten.de/svglbc/>

<http://www.kevlindev.com/tutorials/basics/index.htm>

<http://www.academictutorials.com/svg/>

<http://roitsystems.com/tutorial/>

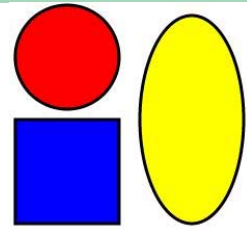
http://apike.ca/prog_svg.html

Die mit SVG darstellbaren Grafiken sind sehr vielfältig und die Mächtigkeit der Sprache kann man daran ablesen, dass man komplexe Grafiken aus Adobe Illustrator oder Corel-Draw in dieses Format exportieren kann und das Ergebnis ohne Qualitätsänderung im jeweils anderen Programm weiterbearbeitbar ist und auch im kompatiblen Browser darstellbar ist.

Der Nachteil ist, dass man jetzt zwar eine perfekte Grafik hat aber keinen Text, der sie erklären könnte. Man kann zwar Grafik-Text zur Beschreibung verwenden aber eine HTML-Seite wird es dadurch auch noch nicht,

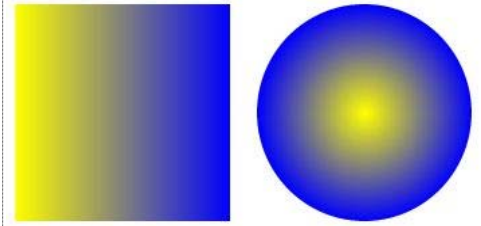
```
<?xml version="1.0" encoding="UTF-8"?>
<svg xmlns="http://www.w3.org/2000/svg" >
<g style="stroke:black; stroke-width:3;">
  <circle cx="60" cy="60" r="50" style="fill:red" />
  <rect x="10" y="120" width="100" height="100" style="fill:blue" />
  <ellipse cx="180" cy="120" rx="50" ry="100" style="fill:yellow" />
</g>
</svg>
```

Grundformen: b10.svg



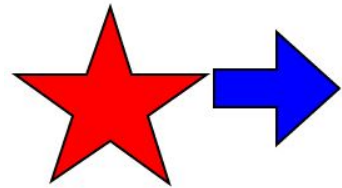
```
<?xml version="1.0" encoding="utf-8"?>
<svg xmlns="http://www.w3.org/2000/svg" >
<linearGradient id="gradient1">
  <stop offset="0%" style="stop-color: yellow" />
  <stop offset="100%" style="stop-color: blue" />
</linearGradient>
<radialGradient id="gradient2">
  <stop offset="0%" style="stop-color: yellow" />
  <stop offset="100%" style="stop-color: blue" />
</radialGradient>
<rect x="10" y="10" width="160" height="160" style="fill:url(#gradient1)" />
<circle cx="270" cy="90" r="80" style="fill:url(#gradient2)" />
</svg>
```

Gradient: b23.svg



```
<?xml version="1.0" encoding="utf-8"?>
<svg xmlns="http://www.w3.org/2000/svg" >
<polygon style="fill:red; stroke: black; stroke-width:2"
  points="97,20 114,74 174,74 127,107 144,161 97,127 50,159 67,107 21,74 78,74" />
<polygon style="fill:blue; stroke: black; stroke-width:2"
  points="180,70 230,70 230,40 280,85 230,130 230,100 180,100"/>
</svg>
```

Polygon: b30.svg



```
<?xml version="1.0" encoding="utf-8"?>
<svg xmlns="http://www.w3.org/2000/svg" >
<text x="60" y="60" style="font-size: 55;">PCNEWS SVG</text>
```

Text: b35.svg

```
<text x="60" y="140"
  style="stroke: black; fill: none; font-size: 55; font-family: 'Times New Roman'">
  PCNEWS SVG
</text>
```

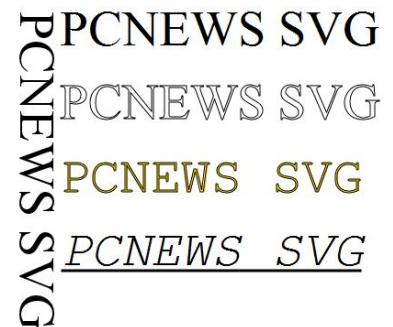
```
<text x="60" y="220"
  style="stroke: black; fill: gold; font-size: 55; font-family: 'Courier New' ">
  PCNEWS SVG
</text>
```

```
<text x="60" y="300"
  style="font-size: 55; font-style: italic; font-family: 'Courier New';
  text-decoration: underline;">
  PCNEWS SVG
</text>
```

```
<text x="30" y="10" style="font-size: 55; writing-mode:tb">PCNEWS SVG</text>
</svg>
```

```
<?xml version="1.0" encoding="utf-8"?>
<svg xmlns="http://www.w3.org/2000/svg" >
<filter id="filtereffect1" height="150%">
  <feGaussianBlur in="SourceAlpha" stdDeviation="6" result="image1" /></filter>
<filter id="filtereffect2" height="150%">
  <feGaussianBlur in="SourceAlpha" stdDeviation="6" result="image1" />
  <feOffset in="image1" result="image2" dx="5" dy="5" />
  <feComposite in="SourceGraphic" in2="image2" operator="over" /> </filter>
<filter id="emboss" >
  <feGaussianBlur in="SourceAlpha" stdDeviation="2" result="blur"/>
  <feSpecularLighting in="blur" surfaceScale="-3" style="lighting-color:white"
  specularConstant="1" specularExponent="16" result="spec" kernelUnitLength="1" >
  <feDistantLight azimuth="45" elevation="45" /></feSpecularLighting>
  <feComposite in="spec" in2="SourceGraphic" operator="in" result="specOut"/>
</filter>
<g style="font-size:100; font-weight:bold">
<text x="20" y="120" style="filter:url(#filtereffect1)">PCNEWS SVG</text>
<text x="20" y="220" style="fill: orange; filter:url(#filtereffect2)">PCNEWS SVG
</text>
<text x="20" y="320" style="fill: orange; filter:url(#filtereffect2)">
  PCNEWS SVG
</text>
<text x="20" y="320" style="fill: orange; filter:url(#emboss)">
  PCNEWS SVG
</text>
</g></svg>
```

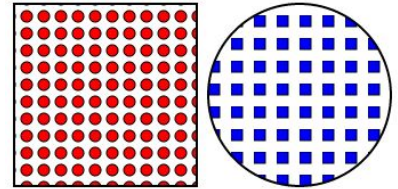
Filter: b36.svg




```
<?xml version="1.0" encoding="utf-8"?>
<svg xmlns="http://www.w3.org/2000/svg" >
```

Füllmuster: b50.svg

```
<pattern id="pattern1" x="0" y="0" width="15" height="15"
  patternUnits="userSpaceOnUse">
  <circle cx="6" cy="6" r="5" style="fill: red; stroke:black;"/>
</pattern>
<pattern id="pattern2" x="0" y="0" width="20" height="20"
  patternUnits="userSpaceOnUse">
  <rect x="0" y="0" height="10" width="10" style="fill: blue; stroke:black;"/>
</pattern>
<rect x="10" y="10" width="160" height="160"
  style="fill:url(#pattern1); stroke:black; stroke-width:2;"/>
<circle cx="260" cy="90" r="80"
  style="fill:url(#pattern2); stroke:black; stroke-width:2;"/>
</svg>
```



```
<?xml version="1.0" encoding="utf-8"?>
<svg xmlns="http://www.w3.org/2000/svg" >
```

Pfade: b60ori.svg

```
<g style="fill:#c1c1bf;">
  <path d="M 329.336 727.552
    C 315.224 726.328 304.136 715.816 303.128 694.936
    C 306.368 639.496 309.608 582.112 271.232 545.104
    C 265.256 499.024 244.016 482.104 234.008 452.512
    L 218.24 441.208 L 237.104 411.688
    L 245.168 411.904 L 323.936 571.168
    L 340.424 651.448 L 329.336 727.552
    z" />
</g> ...
</svg>
```

Originaldatei von
<http://de.wikipedia.org/wiki/Datei:Tux.svg>

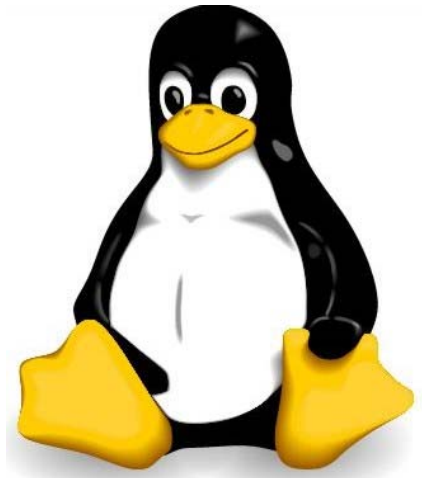


Diagramm1: b70.svg

Leistung des SK-Rapid bei den letzten Bundesligaspielen. (Siehe <http://rapid.iam.at/>)

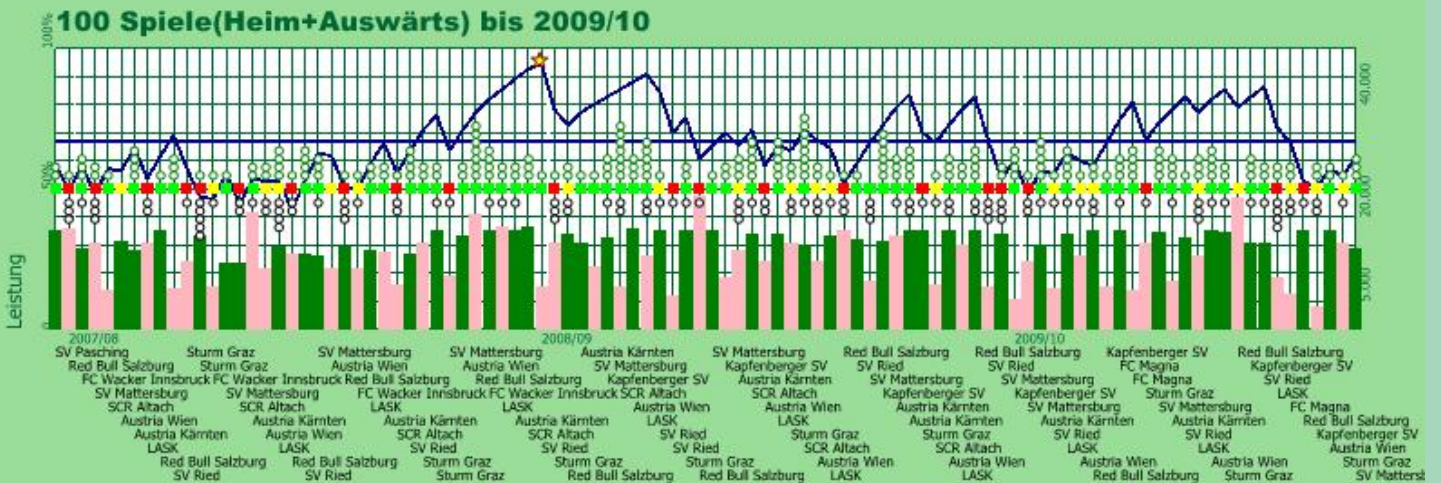
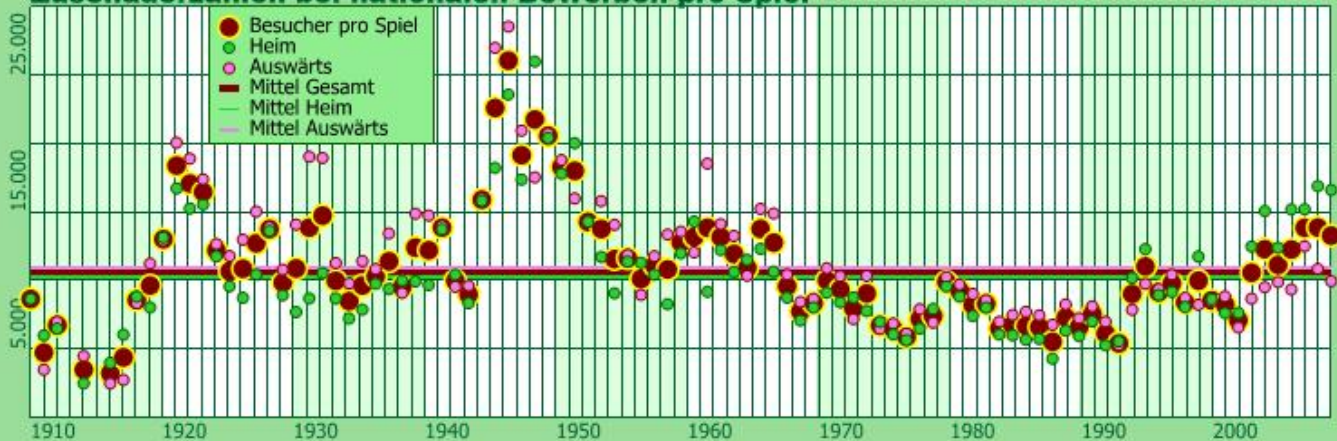


Diagramm2: b80.svg

Zuschauerzahlen des SK Rapid (Siehe <http://rapid.iam.at/statistics.aspx?id=graphics&id1=visitors>)

Zuschauerzahlen bei nationalen Bewerbungen pro Spiel



denn man kann nicht Html-Elemente in das SVG-Format einfügen. Fügt man zum Beispiel `<p>Test</p>` ein, passiert gar nichts, denn das Element `p` ist in SVG nicht definiert und wird ignoriert. Macht man dann noch den Fehler, die einheitliche Kleinschreibung nicht zu beachten, wie in `<P>Test</P>`, erhält man eine Fehlermeldung.

Diese Beschränkung ist durch die DTD (*Document Type Definition*) in der zweiten Zeile des Codes zu suchen:

```
<!DOCTYPE svg PUBLIC "-//W3C/DTD SVG 1.1//EN"
"http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">
```

Es gibt aber auch Dokumentdefinitionen, die es erlauben, verschiedene Sprachen auf einer Seite zu mischen. Will man daher HTML und SVG-Grafik auf einer Seite kombinieren, muss man folgende DTD benutzen:

```
http://www.w3.org/TR/2002/WD-XHTMLplusMathMLplusSVG-20020809/
```

Jetzt erhält man eine XHTML-Datei als Rahmen, die es erlaubt SVG-Abschnitte, also Vektorgrafiken einzubetten:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC
"-//W3C/DTD XHTML 1.1 plus MathML 2.0 plus SVG 1.1//EN"
"http://www.w3.org/2002/04/xhtml-math-svg/xhtml-math-svg.dtd">
<html>
```

```
<body>
<p>Dieser Text erscheint mit der Grafik auf einer Seite.</p>
```

```
<svg xmlns="http://www.w3.org/2000/svg"
width="800" height="800"
xmlns:xlink="http://www.w3.org/1999/xlink">
<g style="fill-opacity:0.7; stroke:black; width:0.1cm;">
<circle cx="200" cy="100" r="100" style="fill:red;"
transform="translate(0,50)" />
<circle cx="200" cy="100" r="100" style="fill:blue;"
transform="translate(70,150)" />
<circle cx="200" cy="100" r="100" style="fill:green;"
transform="translate(-70,150)" />
```

```
</g>
</svg>
</body>
</html>
```

In diesem Codebeispiel wird die vorher erzeugte Grafik in XHTML eingebettet. Der Grafik-Abschnitt wird durch das `svg`-Tag eingerahmt. Es geht aber auch umgekehrt, man kann auch HTML in eine SVG-Grafik einbetten; Details dazu im W3C-Dokument:

```
http://www.w3.org/TR/2002/WD-XHTMLplusMathMLplusSVG-20020809/
#howto-svg
```

Warum eigentlich XHTML und nicht HTML?

XHTML1.0 ist eine strengere Version von HTML4.0. Browser sind fehlertolerant und ignorieren Fehler im HTML- (oder XHTML-)Kode. Es ist aber im Sinne der Präzision der Darstellung hilfreich, wenn eine strenge Überprüfung möglich ist und das ist bei XHTML der Fall.

Äußerlich fällt auf, dass alle Tags jetzt klein geschrieben sind (`<html>` statt `<HTML>`) und einzeln stehende Tags wie zum Beispiel `
` jetzt abgeschlossen werden müssen: `
`. Für den Webdesigner ergibt sich eine viel bessere Überprüfungsmöglichkeit für die Syntax und eine praktisch vollkommene Trennung von Inhalt und Darstellung, weil die ursprünglich in HTML enthaltenen Formatier-Kommandos wie zum Beispiel `` oder `` jetzt nicht mehr verwendet werden und Style-Sheets an ihre Stelle treten. Für professionelle Webseiten ist jedenfalls XHTML vorzuziehen. Um eine Seite auf XHTML-Korrektheit zu überprüfen, verwendet man den Syntax-Checker von W3C. <http://validator.w3.org/>

Beispiele

Auf den vorigen zwei Seiten findet man Beispiele für die grundlegende Syntax von SVG.

Eindrucksvoll ist der Linux-Pinguin Tux, der auch auf der Titelseite zu dieser Ausgabe verwendet wird.

Die beiden letzten Grafiken zeigen, dass man SVG bestens für Diagramme verwenden kann.

Bei der Darstellung der Schriften (35.svg und 36.svg) zeigen sich große Abweichungen bei der Darstellung in verschiedenen Browsern. Wahrscheinlich ist es erforderlich, die Attribute noch genauer festzulegen, um die Freiheitsgrade bei der Darstellung einzuschränken.

SVG-Viewer für Internet-Explorer

Um auch am Internet-Explorer SVG-Dateien direkt anzeigen zu können, benötigt man ein Plug-In, das man hier downloaden kann:

```
http://www.adobe.com/svg/viewer/install/main.html
```

Hier kann man den Svc-Viewer testen:

```
http://www.adobe.com/svg/viewer/install/svgtest.html
```

Wenn das Plug-In erfolgreich installiert ist, sieht man folgendes Bild:



Leider wird dieses Plugin von Adobe nicht mehr unterstützt. Es ist daher nur als ein Behelf zu verstehen. Die Versionsnummer dieses Viewer ist 3, die aktuelle Version, der laufenden Adobe-Entwicklung aber bereits 6.

Wer laufend mit SVG arbeiten will, sollte Opera, Chrome oder Firefox verwenden. Diese Browser machen den besten Eindruck. Man kann auch in die Bilder hineinzoomen, natürlich verlustfrei, denn es handelt sich ja um Vektorgrafiken.

SVG-Editoren

Für einfache Versuche kann man den SVG-Kode so wie jede Html-Datei mit einem Texteditor herstellen. Doch SVG ist nicht so fehlertolerant wie Html. Schreibfehler führen zu Fehlermeldungen.

Um daher den Kode händisch herstellen und bearbeiten zu können, braucht man einen Grafik-Editor. Diese Editoren helfen beim Entwurf der statischen Teile eines Entwurfs. Verbreitete Programme zur Herstellung von SVG-Kode sind zum Beispiel Adobe Illustrator, Corel Draw oder GIMP. Diese benutzen aber SVG nicht als primäres Format sondern nur als eines von vielen Export/Import-Formaten. Der entstehende SVG-Kode ist etwa so gut wie der HTML-Kode einer im HTML-Format gespeicherten Word-Datei.

Die im folgenden Abschnitt beschriebenen Editoren sind dagegen auf die Herstellung von SVG-Grafiken spezialisiert. Der entstehende SVG-Kode ist sehr effizient.

Die Schwierigkeiten bei dem Datenaustausch zwischen diesen reinen SVG-Editoren und etwa CorelDraw entstehen durch die verschiedenartige Philosophie dieser Programme.

Wenn man eine einfach formatierte SVG-Datei in CorelDraw importiert danach wieder exportiert, wird der Kode zum Teil stark verändert. Zu den geringfügigen Veränderungen gehört, dass jedes Tag eine ID bekommt und dass die Koordinaten auf die verwendete Seitengröße A4 umgerechnet werden. Bei komplexeren Strukturen kann der Kode aber bis zu Unkenntlichkeit verändert werden. Das liegt daran, dass CorelDraw sein eigenes internes Format in SVG umrechnet und sich dabei auf die einfachsten Strukturen beschränkt. Beim Export in einen reinen SVG-Editor spielt das weniger Rolle, wohl aber beim Import, denn wenn der SVG-Editor das SVG-Format voll nutzt, versagt CorelDraw, denn die komplexe SVG-Kodierung wird von CorelDraw falsch interpretiert.

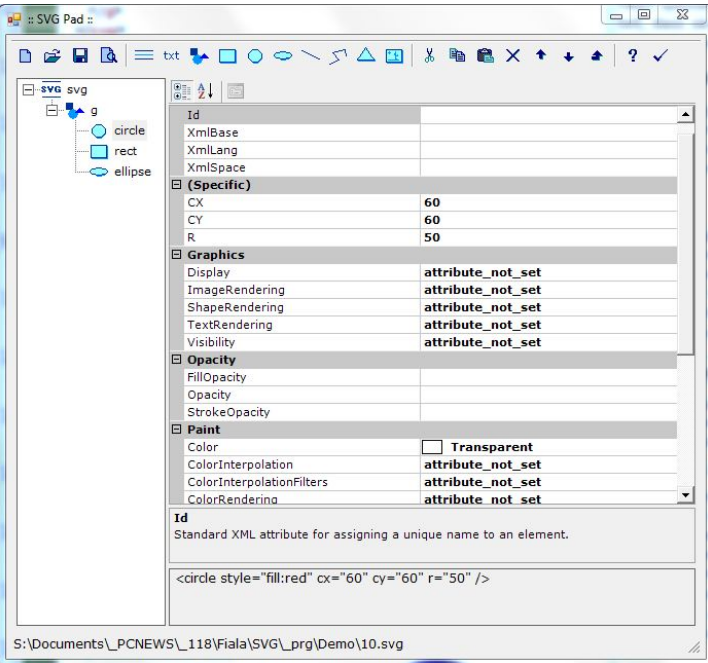
Man muss es sich daher zur Regel machen, möglichst auf die fortgeschrittenen Kodierungstechniken zu verzichten und beschränkt sich auf die einfacheren Strukturen.

SVG-Pad

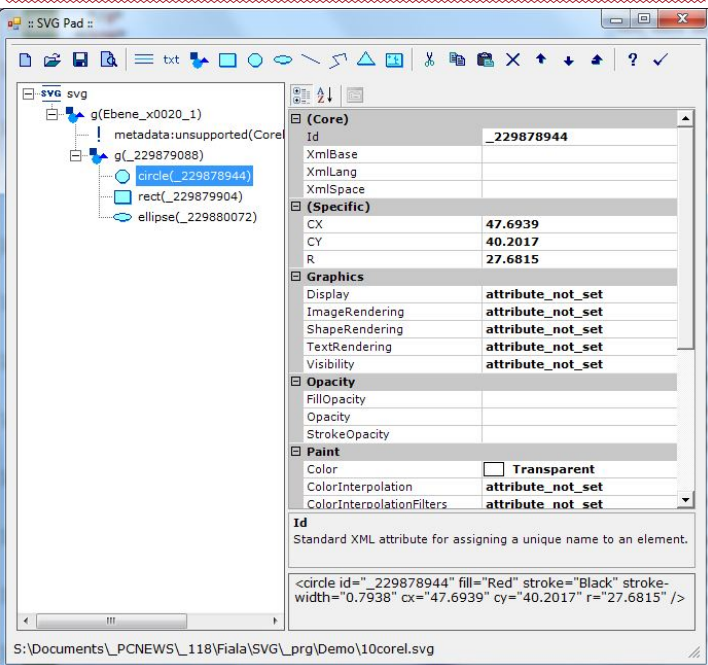
Mit SVG-Pad editiert man eine SVG-Datei auf dem Tag/Attribut-Niveau. Das Programm gibt einen Überblick über die verfügbaren Elemente in SVG und bei jedem Element werden nur die zulässigen Attribute angezeigt. Das Programm verfügt aber über keine Darstellungsmöglichkeit. Man speichert den generierten Code und zeigt ihn mit einem Browser an.

Nachteil des SVG-Pad: es werden nur die grundlegenden Typen unterstützt. Bei komplexeren Typen bekommt man den Hinweis „not supported“. Wenn man dagegen SVG erlernen will, ist diese Darstellung sehr angenehm, weil sie praktisch das Handbuch erspart.

Datei 10.svg im Editor SVG-Pad



BDatei 10.svg nach Import in und Export aus CorelDraw



AKs SVG-Editor

Dieser Editor ist ein Javascript-Programm und läuft in einem Fenster des Internet-Explorers ab. Das installierte Adobe-Plug-In ist erforderlich. Andere Browser sind für die Darstellung nicht geeignet.

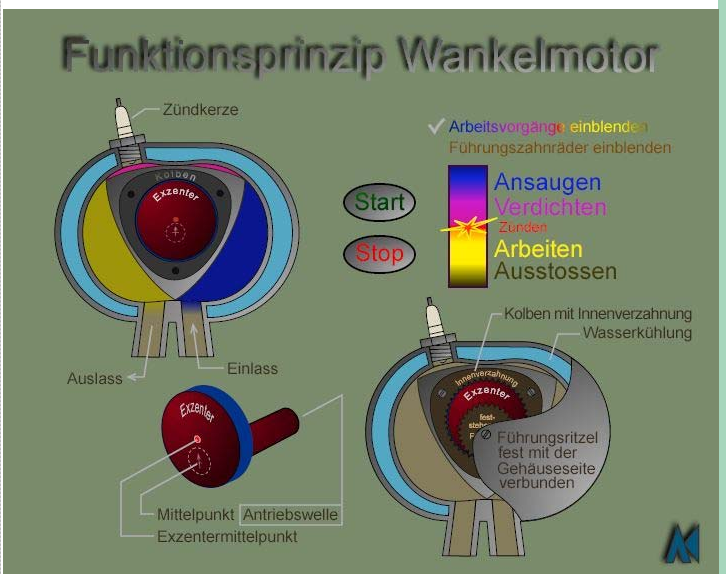
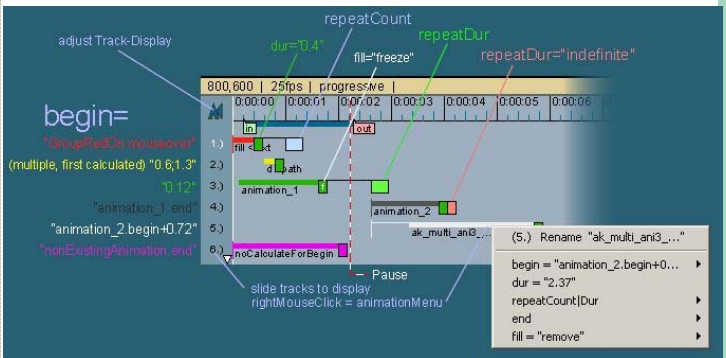
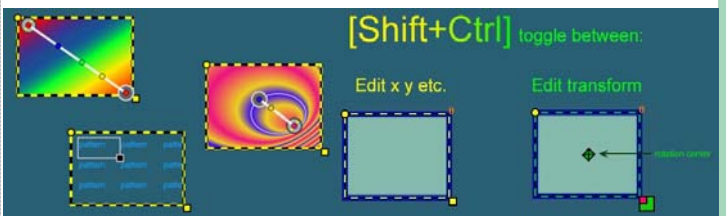
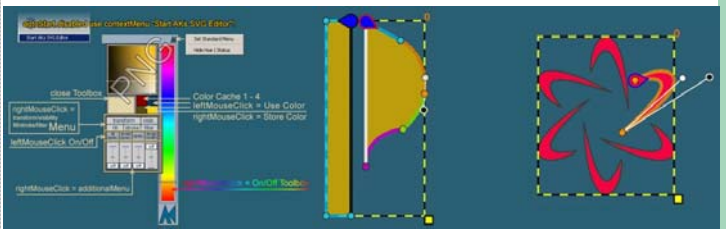
<http://www.krammel.net/>.

Dieser Editor ist noch „nah am SVG-Kode“. Es ist aber erstaunlich, was aus dem einen MB Code herausgeholt werden kann. Besonders eindrucksvoll ist die Möglichkeit, Animationen mit SVG-Grafik herzustellen.

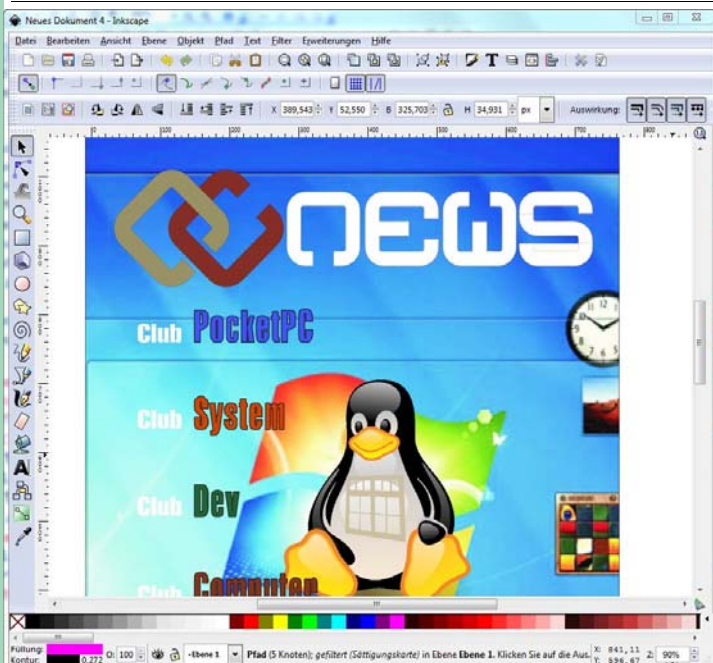
Hier einige Eindrücke aus der Hilfedatei:

Nachteilig erscheint mir diese enge Bindung an den Internet-Explorer, noch dazu, weil gerade dieser Browser seinerseits von einem nicht mehr unterstützten Adobe-Plugin abhängt. Der entstehende SVG-Kode läuft nur am Internet-Explorer.

Der Grund dürfte aber die Spezialisierung auf Animationen sein. Das letzte Bild zeigt die Animation eines Wankelmotors, das Bild darüber zeigt die Einstellungen für das Drehbuch.



Inkscape



Es war sehr einfach, die PCNEWS-Coverseite in Inkscape zu übernehmen. Wirklich vermisst habe ich keine Funktion aus CorelDraw. Im Gegenteil! Es gibt zahlreiche weitergehende und sehr nützliche Funktionen, die auch in den kommerziellen Programmen nicht verfügbar sind. Eindrucksvoll die überaus große Zahl von Filtern, die man auf Objekte anwenden kann.

Mathematiklehrer werden sich freuen, dass man die besonderen Punkte eines Dreiecks für beliebige Dreiecke automatisch einzeichnen lassen kann.

In vielen Details kann man die große Nähe zwischen dem Format SVG und den möglichen Einstellungen erkennen.

Homepage

<http://www.inkscape.org/?lang=de>

Anleitungen

<http://inkscapetutorials.wordpress.com/>

<http://de.wikipedia.org/wiki/Inkscape>

<http://de.wikibooks.org/wiki/Inkscape>

<http://wiki.inkscape.org/>

Beispiel aus der Messtechnik

Eine Messkurve bestehend aus x- und y-Werten ist darzustellen. Dimension und Achsenbeschriftung lassen wir der Einfachheit halber außer Acht. Die Werte sind (jeweils x- und y-Wert).

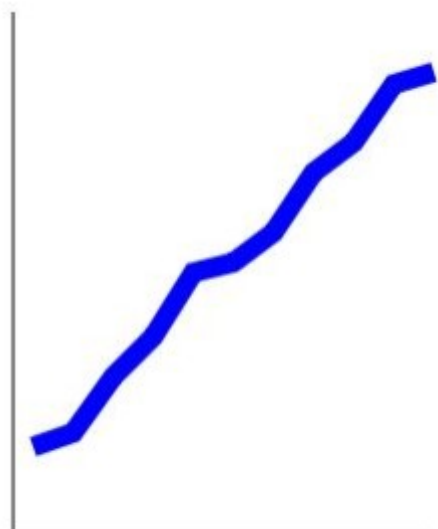
0 3,3 - 1 4 - 2 6,8 - 3 8,8 - 4 12 - 5 12,5 - 6 14 - 7 17 - 8 18,5 - 9 21,4 - 10 22

Um aus diesen Zahlenpaaren ein Diagramm herzustellen, müssen die Messwerte in die geeignete SVG-Form umgewandelt werden. Zunächst muss man beachten, dass die Zählung der y-Achse von oben nach unten verläuft. Weiters muss man eine geeignete Skalierung wählen. In unserem Beispiel zeichnen wir zwei Achsen als Linien und die Kurve als Mehrfachlinie dazu. Jeder Messwert wird mit 10 multipliziert (damit wird er ganzzahlig), außerdem wird 10 addiert, um Platz für eine Achsenbeschriftung zu bekommen. Die x-Achsenwerte werden mit dem Faktor 2 multipliziert.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"
"http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">
<svg xmlns="http://www.w3.org/2000/svg"
width="500" height="500">
<g transform="translate(100,100)">
<line x1="10" y1="310" x2="220" y2="310" stroke="black"
stroke-width="1" />
<line x1="10" y1="310" x2="10" y2="50" stroke="black"
stroke-width="1" />
<polyline fill="none" stroke="blue" stroke-width="10"
points="20,267 40,260 60,232 80,212 100,180 120,175
140,160 160,130 180,115 200,86 220,80" />
</g>
```

</svg>

Man kann einen Linienzug durch ein `line`-Tag oder ein `polyline`-Tag darstellen. Das `polyline`-Tag hat aber gegenüber der einfachen Linie den Vorteil, dass dickere Linien an den Knickstellen ordentlich verbunden werden und nicht abgehackt erscheinen. Das `g`-Tag gruppiert die Elemente und verschiebt das Bild mit dem Attribut `translate`. (Auch Drehungen, und Verzerrungen sind möglich.) Dazu kommt noch der unveränderliche Dokumentenkopf mit der Angabe der Darstellungsgröße 500, 500. Hinweis: Sowohl die Blattgröße als auch die Positionswerte in den einzelnen Grafikelementen können auch als Dezimalzahl in mm, cm und Points angegeben werden.



Programmatische Herstellung von SVG-Kode

Bei veränderlichen Werten wie zum Beispiel bei Messwerten oder bei Börsenkursen ist es nicht möglich, Seiten händisch zu editieren; vielmehr muss ein Programm die variablen Daten mit den statischen Bestandteilen verbinden. Eine kleine Unbequemlichkeit ist die Komplexität der SVG-Beschreibungssprache, und dass es eben XML-Kode ist, der gegenüber Schreibfehlern intolerant ist. Es empfiehlt sich daher, den Code nicht selbst herzustellen, etwa in der Form

```
string s = "<circle cx=\""+cx+"\" cy=\""+cy+"\" style=...
```

sondern dazu eine Bibliothek zu benutzen, die die einzelnen Objekte in Klassen formuliert und man lediglich die entsprechenden Objekte zu generieren hat und den String `s` als Ergebnis abrufen kann. Eine solche Bibliothek kommt von Ucancode (auch Hersteller von SVG-Pad):

Bibliothek zur Erstellung der SVG-Datei

Die folgende Bibliothek ist in C# geschrieben und ist für die Verwendung mit Visual Studio vorgesehen.

http://www.ucancode.net/ActiveX_Programming_Article_Com_Tutorial/Create-Open-Load-Save-Import-Export-Edit-SVG-File-Docment-With-C-NET-Class-Library.htm

Mit dieser Bibliothek kann man die Grafikelemente bequem generieren. Man muss lediglich die dll-Datei in den Ordner `bin` kopieren und als Verweis zu dem aktuellen Projekt hinzufügen. Im eigentlichen Programm generiert man das obige Beispiel so (die Klasselemente und Funktionsnamen wurde hier der Einfachheit halber weggelassen):

Bild.aspx

```
using SVGLib; // Einbindung der Bibliothek
```

```
// Erzeugt den Dokumentenkopf
```

```
// eines Svg-Dokuments
```

```
SvgDoc doc = new SvgDoc();
```

```
// Erzeugt das Root-element svg
```

```
// mit den dazugehörigen Attributen
```

```
SvgRoot root = doc.CreateNewDocument();
```

```
root.Width = "500"; // Zeichenbreite
```

```
root.Height = "500"; // Zeichenhöhe
```

```
// x-Achse
```

```
SVGLib.SvgLine x = new SVGLib.SvgLine(
```

```
doc, "10", "310", "220", "310", Color.Black);
```

```
l.StrokeWidth = "10";
```

```
l.Stroke = Color.Black;
```

```
doc.AddElement(root, x);
```

```
// y-Achse
```

```
SVGLib.SvgLine y = new SVGLib.SvgLine(
```

```
doc, "10", "310", "10", "50", Color.Black);
```

```
l.StrokeWidth = "10";
```

```
l.Stroke = Color.Black;
```

```
doc.AddElement(root, y);
```



```
// Messwerte
SVGLib.SvgPolyline p = new SVGLib.SvgPolyline(doc, "20,267 40,260
60,232 80,212 100,180 120,175 140,160 160,130 180,115 200,86
220,80");
poly_power.StrokeWidth = "10";
poly_power.Stroke = Color.Blue;
doc.AddElement(root, p);

// Datei speichern
bool FehlerFrei =
doc.SaveToFile(@"C:\pfad\Bild.svg");
if (FehlerFrei)
    Response.Write("Datei gespeichert");
else
    Response.Write("Datei nicht gespeichert");
```

Als Ergebnis wird die Datei Bild.svg im Pfad C:\pfad gespeichert. Der Vorteil dieser Vorgangsweise ist, dass man sich nur mehr um die eigentlichen Variablen kümmern muss, denn die XML-Tags und die Namen der Attribute werden durch die einzelnen Klassen gekapselt. Es ist jetzt keine Schwierigkeit mehr, statt der konstanten Werte in diesem Beispiel die Variablen aus der Tabellenkalkulation, der Datenbank oder eben direkt aus der Messung einzufügen.

Umwandlung von SVG-Dateien Pixel-Grafik

Das war aber erst der halbe Weg, denn wir haben jetzt zwar eine SVG-Datei und diese Datei kann man mit den meisten Browsern anzeigen, doch wir wollen ja alle Browser bedienen und dazu braucht man ein Bild, wenigstens so lange als der Internet Explorer das SVG-Format nicht verstehen will. Kurioserweise stellt gerade eine Microsoft-OpenSource-Community ein Umwandlungsprogramm zur Verfügung (<http://www.codeplex.com/svg>): eine C#-Bibliothek, die eine SVG-Datei als Input akzeptiert und daraus eine Grafikdatei (gif, jpg oder png) herstellt.

Um diese Bibliothek in der Entwicklungsumgebung zu nutzen, genügt es, die DLL in das Verzeichnis bin zu kopieren und danach diese Datei als Verweis zum Projekt hinzuzufügen. Nur, wenn man in den Source-Code eingreifen will oder muss, kopiert man die Quelldateien in ein eigenes Verzeichnis und kompiliert es mit dem eigenen Projekt mit.

In meinen bisherigen Projekten erwies sich diese Bibliothek als nahezu perfekt.

Wie setzt man diese Bibliothek ein?

Im obigen Beispiel streicht man die Zeilen ab
bool FehlerFrei = doc.SaveToFile(@"C:\pfad\Bild.svg");
 und erzeugt keine SVG-Datei. Stattdessen speichert man den generierten XML-Kode in einem String. Leider akzeptiert die Microsoft-Bibliothek keinen String als Input und erfordert stattdessen einen Stream. Meine Methode, den Stream zu erzeugen ist wahrscheinlich nicht optimal, das müsste viele einfacher gehen, aber egal, es funktioniert. Die eigentliche Umwandlung in eine Bild erfolgt durch die Befehlsfolge **Open() -> Draw() -> Save()**. Damit wäre das Bild schon erzeugt.

```
string s = doc.GetXML();
byte[] buffer = new byte[s.Length * 2];
for (int i = 0; i < s.Length; i++)
{
    b = char.ConvertToUtf32(s, i);
    [2 * i] = (byte)(b / 256);
    [2 * i + 1] = (byte)(b % 256);
}
MemoryStream memStream = MemoryStream(buffer);
SvgDocument.Open(memStream).Draw().Save(Path + "Bild.jpg",
ImageFormat.Jpeg);
string RelativePath = Request.Path;
Response.Flush();
RelativePath = RelativePath.Substring(0,
RelativePath.LastIndexOf(@"\"));
string PictureFileName = RelativePath + "Bild.jpg";
Image_Statistik.ImageUrl = PictureFileName;
```

Zum Abschluss wird noch der relative Pfad zu dem Bild ermittelt und die Bildadresse im Image-Element eingestellt.

Anwendung

In einer Html-Seite verwendet man

```

```

Wenn man das Bild gar nicht als Datei benötigt, kann man es durch folgende Zeile direkt in den Output-Stream schreiben:

Bild1.aspx

```
Response.ContentType = "image/jpeg";
SvgDocument.Open(memStream).Draw();
Save(Response.OutputStream, ImageFormat.Jpeg);
```

Diese Version ist Dynamik pur, denn das Bild existiert gar nicht als Datei; weder im SVG-Format noch im JPG-Format. Bei großen Grafiken kann aber die Berechnung zu einer unzulässigen Verzögerung führen. Wenn sich daher die Daten nicht gerade im Sekundentakt sondern zum Beispiel täglich oder wöchentlich ändern, sollte man die JPG-Datei speichern. Man regeneriert die JPG-Datei immer nur dann, wenn sich die Daten ändern.

Verwendete Tools und Dateien

- Visual Web Developer 2008 Express Edition
- Einen Microsoft-Server mit installiertem Framework, Version 3.5 (wichtig)
- Bibliothek von Ucancode zur Erzeugung des SVG-Kode
- Bibliothek von Microsoft zur Generierung eines Bildes aus dem SVG-Kode
- Musterdatei Bild.aspx erzeugt die Dateien Bild.svg und Bild.jpg
- Musterdatei Bild1.aspx erzeugt direkt ein Jpg-Bild

Für die Entwicklung wird kein Webserver benötigt, denn auf jedem Windows-PC in ein baugleicher Webserver für die Entwicklung vorhanden (*Verwaltung->Internet-Informationdienste*). Man muss nur darauf achten, dass das Framework Version 3.5 installiert ist, denn die Microsoft-Bibliothek erfordert ein besonderes Modul LINC, das nur dort enthalten ist.

Bild ohne Programm herstellen

Nun, es geht auch einfacher. Im Zuge des Testens wurde ein WebTool entwickelt, das für kleine Anwendungen auch von anderen Usern benutzt werden kann: <http://iam.at/svg2jpg/>

Wenn man eine SVG-Datei herstellt, egal ob mit einem Zeichenprogramm oder mit einem serverseitigen Programm (ASP, ASPX oder PHP) genügt es mit der folgenden Anwendung, die Adresse <http://iam.at/svg2jpg/> aufzurufen und man erhält als Antwort das zugehörige Bild in die eigene Seite geliefert.

Der konkrete Aufruf erfolgt so:

```
http://iam.at/svg2jpg?url=<Internet-Adresse der SVG-Datei>
```

url kann eine beliebige Adresse auf Ihrem eigenen Server oder sonst wo im Internet sein.

Beispiel:

```
http://iam.at/svg2jpg?url=http://iam.at/svg2jpg/exa/Beispiel1.svg
```

Anwendung in einer HTML-Datei

In einer HTML-Datei ist ein Vektor-Grafik-Bild im Rahmen eines **img**-Tag einzufügen. Der Grafik-Kode befindet sich auf der Adresse <http://iam.at/svg2jpg/exa/Beispiel1.svg> (wenn Sie das in Ihrer Anwendung verwenden, steht hier: <http://MeineDomäne.at/Pfad/MeineSvgDatei.svg>).

```

```

Es ist aber zu beachten, dass der eigene Server Dateien mit der Endung **svg** auch tatsächlich als Text zurückschickt. Beim IIS ist das normalerweise nicht der Fall. Daher muss bei den Mime-Typen des Servers die Dateiendung **svg** bekannt gemacht werden und mit dem Typ **text/plain** verknüpft werden. (Clubmitglieder, die ihren Webpace mit helm.ccc.at verwalten, können diesen Eintrag unter dem Menüpunkt *MIME-Typen* hinzufügen.)

```
.svg image/svg
```

Wenn man keine Möglichkeit hat, den Eintrag vorzunehmen, dann muss man die SVG-Datei in eine Textdatei umbenennen, damit die Datei vom Server korrekt behandelt wird und ohne Änderung an den Client zurückgeschickt wird.

Weitere Beispiele zum Ausprobieren finden Sie im Verzeichnis <http://iam.at/svg2jpg/exa/>.

SVG ist eine Sprache, die sich laufend entwickelt. Es kommt daher vor, dass manche fortgeschrittene Techniken nicht funktionieren, etwa die Datei

```
http://iam.at/svg2jpg/?url=http://iam.at/svg2jpg/exa/23.svg
```

Zusammenfassung

Es muss im Interesse des Anwenders sein, dass seine Arbeiten in Formaten gesichert werden, die auf einem möglichst breiten Konsens vieler Hersteller beruhen und nicht von einem einzigen Erzeuger abhängen. HTML, XML und SVG sind solche Formate. Proprietäre Formate können bestenfalls als Übergangslösung akzeptiert werden.

Gerade für das SVG-Format gibt es mit Inkscape auch ein den großen Vektor-Grafik-Programmen gleichwertiges Äquivalent das dieses Format als zentrales Darstellungsformat verwendet.

Auf der Browserseite zwingt der normalerweise sehr üppig ausgestattete Internet-Explorer durch die SVG-Verweigerung zu komplizierten Lösungen. Der Entwicklungsaufwand für portable Anwendungen wird sehr groß, wie man an diesem Beispiel sieht. Erst in der Version 9 des Internet-Explorers gibt es zaghafte Versuche, SVG zu implementieren (ca. 25%).

Links zu diesem Artikel

Alle Dateien zum Download befinden sich bei der Webversion dieses Artikels.

Die Umwandlung einer SVG-Datei in JPG-Format kann man hier vornehmen:

Download http://pcnews.at/pcn/1xx/11x/118/002400/_prg
Demo <http://fiala.member.pcc.ac/>
Webdienst <http://iam.at/svg2jpg/>

Programm zum Umwandeln eine SVG-Datei in eine JPG-Datei. Erfordert die Bibliotheken `svg.lib` und `svglib.dll`

Programmcode

`svg2jpg.aspx`

```
<%@ Page Language="C#" AutoEventWireup="true"
CodeFile="svg2jpg.aspx.cs" Inherits="svg2jpg" %>
```

`sv2jpg.aspx.cs`

```
using System;
using Svg;
using SVGLib;
using System.Drawing.Imaging;
using System.IO;
using System.Net;
using System.Text;
using System.Drawing;
public partial class svg2jpg : System.Web.UI.Page
{
    void Fehlermeldung(string Message)
    {
        SVGLib.SvgDoc doc = new SvgDoc();
        SVGLib.SvgRoot root = doc.CreateNewDocument();
        root.Width = "300";
        root.Height = "100";
        SVGLib.SvgRect Background = new SVGLib.SvgRect(doc, "0", "0",
"300", "100", "0", Color.White, Color.White);
        doc.AddElement(root, Background);
        SVGLib.SvgText FehlerText = new SVGLib.SvgText(doc);
        FehlerText.Value = Message;
        FehlerText.X = "20";
        FehlerText.Y = "20";
        FehlerText.FontFamily = "Arial Black";
        FehlerText.FontSize = "15pt";
        doc.AddElement(root, FehlerText);
        string s = doc.GetXML();
        byte[] buffer = new byte[s.Length];
        for (int i = 0; i < s.Length; i++)
        {
            buffer[i] = (byte)s[i];
        }
        MemoryStream memStream = new MemoryStream(buffer);
        Response.ContentType = "image/jpeg";
        SvgDocument.Open(memStream).Draw().Save
(Response.OutputStream, ImageFormat.Jpeg);
        Response.End();
    }
    bool IsFile(string FileName)
    {
        if (FileName.IndexOf(".") > 0)
            return true;
        else
            return false;
    }
    string ReadUri(string Path)
    {
        Uri uri;
        try
```

```
{
    uri = new Uri(Path);
}
catch
{
    Fehlermeldung("Falsches Url-Format\n" + Path);
    return "";
}
string[] s = uri.Segments;
int SegmentCount = s.Length;
if (IsFile(s[SegmentCount - 1]))
    SegmentCount--;
WebRequest wReq = WebRequest.Create(Path);
WebResponse wResp;
try
{
    wResp = wReq.GetResponse();
}
catch
{
    Fehlermeldung("Dokument nicht gefunden\n" + Path);
    return "";
}
Stream respStream = wResp.GetResponseStream();
StreamReader reader = new StreamReader(respStream,
Encoding.Default);
String respHTML = reader.ReadToEnd();
respStream.Close();
return respHTML;
}
protected void Page_Load(object sender, EventArgs e)
{
    string Url = Request.QueryString["url"];
    if (Url == null) Url = "";
    // Response.Write("Url:" + Url + "<br>"); Response.Flush();
    if (Url == "")
        Fehlermeldung("Kein Url angegeben\n...?url=");
    string s = ReadUri(Url);
    byte[] buffer = new byte[s.Length];
    for (int i = 0; i < s.Length; i++)
    {
        buffer[i] = (byte)s[i];
    }
    MemoryStream memStream = new MemoryStream(buffer);
    Response.ContentType = "image/jpeg";
    try
    {
        SvgDocument doc = new SvgDocument();
        SvgDocument.Open(memStream).Draw().Save
(Response.OutputStream, ImageFormat.Jpeg);
    }
    catch
    {
        Fehlermeldung("Grafik-Fehler");
    }
}
}
```

