



Inhalt

LIESMICH

- 1 Cover
Franz Fiala
- 2 Liebe Leser, Inhalt
Franz Fiala
- 4 Impressum, Autoren, Inserenten

CLUBCOMPUTER

- 2 Clubtermine 2014
- 4 ADIM Skripten
Martin Weissenböck

METATHemen

- 23 Aus für Vorratsdatenspeicherung
Werner Illsinger
- 31 Das Recht auf Vergessen
Werner Illsinger
- 31 Zum Zusammenhang von Motivation und E-Learning
Mattias Hütthaler

CLUBMOBILE

- 22 Sony Xperia Z2 – das bessere Z1?
Paul Belcl
- 24 Sony Smartband SWR10
Paul Belcl

CLUBSYSTEM

- 16 Ubuntu
Günter Hartl
- 30 K&K—oder Karma und Kismet
Thomas Sulak

CLUBDEV

- 5 Titelbild
Franz Fiala
- 6 Wie alles begann...
Franz Fiala
- 8 Codepages in Windows
Franz Fiala
- 9 Am Anfang schuf Gott den ASCII-Code
Franz Fiala
- 10 Unicode
Franz Fiala
- 26 Mikrocontroller-News
Manfred Resel
- 29 DAVE-3
Manfred Resel

LUSTIGES

- 2 Gabalier-CD
Christian Berger
<http://www.karikaturen.guru/>

Liebe Leserinnen und Leser!

Franz Fiala

Digitales Zeitalter

Wenn man sich fragt, was eigentlich die Grundlage unserer Kultur ist, dann lautet meine Antwort: es sind die technischen und vor allem digitalen Systeme, die, getragen von internationalen Normungsgremien den unsichtbaren Unterbau unserer Zivilisation bilden.

Etwa zwischen 1900 und 1960 genühten Elektromechaniker, später Nachrichtentechniker, um die Telefonleitungen, Fernschreiber und Funk-einrichtungen zu betreiben. Mit dem Einzug der EDV in den Alltag ist es aber weit mehr geworden, denn es ist ein konzertiertes Aufgebot von Wissenschaft in vielen Bereichen, die die Entwicklung unserer digitalen Welt vorantreiben. Alle großen Entdeckungen des 20. Jahrhunderts haben überaus praktische Bedeutung für unsere digitale Welt, beginnend bei der Relativitätstheorie, der Quantenphysik, der Chaostheorie und der Kryptografie. Alles wird benötigt, um ein modernes Handy oder einen PC betreiben zu können, so wie wir das gewöhnt sind.

Wie sehr wir von alledem abhängen, merken wir, wenn Rechensysteme ausfallen. Dann stehen nicht nur die Computer still, sondern die gesamte damit verbundene Infrastruktur wie Bahn, Verkauf, Bankgeschäft. Und es gibt keinen Plan-B. Es kann nicht einfach händisch weitergearbeitet werden. Alles beruht auf der korrekten Funktion der digitalen Welt.

Unsere Kultur kann in der heutigen Form Bestand haben, solange es uns gelingt, das technische Know-How auf dem bestehenden Niveau aufrecht zu erhalten und gegen das boomende, oft religiös oder esoterisch begründete Nicht-Wissen zu verteidigen.

Wie viele der Menschen, die täglich Telefon und Computer benutzen, wissen über das Innenleben ihrer Maschinen Bescheid? Wenige, aber das ist nichts Neues, das war immer schon nur eine kleine Gruppe. Und es war immer schon Aufgabe der PCNEWS, dieses Wissen aufzubereiten und weiterzugeben, unter dem Motto von ClubComputer: „Wir begleiten Dich

in die digitale Zukunft“ Aber eigentlich ist diese Zukunft längst zur Gegenwart geworden.

Unsere PCNEWS ist das beste Beispiel für den Übergang von der anfangs analogen zur rein digitalen Herstellung der Druckvorlage. Die Ausgaben 1 bis 30 der PCNEWS wurden im XEROX-Verfahren, teilweise auch im Offset-Druckverfahren aber immer war der Ausgangspunkt eine einfache Papiervorlage. Die Ausgaben 31 bis 70 wurden als Filme der Druckerei übergeben, also immer noch analog. Ab 2001 ist das alleinige Übergabeformat ein PDF-Dokument. Damit hat das digitale Zeitalter der PCNEWS begonnen.

Wir sind also mittendrin, in der digitalen Welt und wollen uns in dieser Ausgabe jener Schicht widmen, die der Verbindung zwischen Technik und Sprache herstellt: dem **Unicode**.

Franz Fiala

ClubComputer

Clubtermine 2014

Di	2014-09-02	Clubabend Android
Mo	2014-09-08	Stammtisch
Do	2014-09-18	Clubabend
Di	2014-10-07	Clubabend
Mo	2014-10-13	Stammtisch
Do	2014-10-23	Clubabend
Di	2014-11-04	Clubabend
Mo	2014-11-10	Stammtisch
Do	2014-11-20	Clubabend
Di	2014-12-02	Weihnachtsfeier

Gabalier-CD

Herbert, ich verlasse Dich. In letzter Zeit fühle ich mich in unserer BEZIEHUNG nicht mehr wohl. Du hast Dich zu einem RÜCKSTÄNDIGEN, PATRIARCHALEN MACHOARSCH entwickelt!



Hau doch ab, verdammte EMANZE. Aber meine GABALIER-CD bleibt da!



METATHemen

MTM

Autoren

Belcl Paul 1966 22,24



Systemberatung und Coaching für mobile Devices;
Direktor für den Bereich Android im Clubcomputer.
Firma BELCL EDV-Koordination & Systemberatung
Hobbies Familienstellen, elektrische Fortbewegung, Fahrradfahren, Fotografieren
pbcl@ccc.at <http://blog.belcl.at/>

Berger Christian 2



Karikaturist und Comiczeichner für Kärntner Zeitungen
Firma Karicartoons
karicartoons@aon.at
<http://www.bergercartoons.com/>

Fiala Franz Dipl.-Ing. 1948 1,2,5,6,8,9,10



Leitung der Redaktion und des Verlags der PCNEWS,
Lehrer für Nachrichtentechnik und Elektronik i.R.
Werdegang BFPZ-Arsenal, TGM Elektronik
Absolvent TU-Wien, Nachrichtentechnik
Privates verheiratet, 1 Kind
franz.fiala@clubcomputer.at <http://fiala.cc/>

Hartl Günter Ing. 1963 16



Wirtschaftsingenieur, Systemadministrator für
Windows Clients und Linux Server in Logistikcenter
Hobbies Krav Maga, Windsurfen, Lesen
ghartl3@gmail.com

Hütthaler Matthias MSc 1984 31



Lehrer und Netzwerkaadministrator an der KPH Wien/
Krems
matthias.huetthaler@kphvie.ac.at

Illsinger Werner Ing. 1968 23, 31



Key Account Manager Financial Services bei Microsoft
Österreich, Präsident von ClubComputer
werner.illsinger@clubcomputer.at
<http://www.illsinger.at/>

Resel Manfred Ing. 1956 26, 29



Lehrer für Technische Informatik und Werkstättenlabor
Schule HTBLA-Hollabrunn, Elektronik-Technische
Informatik
Absolvent TGM, D75
manfred.resel@r.htl-hl.ac.at

Sulak Thomas 30



Unternehmer, IT-Coaching, IT-Consulting, IT-Services
Firma IT-Sulak
Thomas.Sulak@clubcomputer.at

Weissenböck Martin Dir.Dr. 1950 4



Direktor der HTL Wien 3 Rennweg, Leiter der ADIM,
Leiter der ARGE Telekommunikation
martin@weissenboeck.at
<http://www.weissenboeck.at/>

PCNEWS-141

Kennzeichnung ISSN 1022-1611, GZ 02Z031324 M
Layout Microsoft Publisher 2013, GIMP, Inkscape
Herstellung Bogenoffset, 80g
Erscheint Wien, Juni 2014
Texte <http://pcnews.at/?id=PCN141>
Kopien Für den Unterricht oder andere nicht-kommerzielle
Nutzung frei kopierbar. Für gewerbliche Weiterver-
wendung liegen die Nutzungsrechte beim jeweiligen
Autor. (Gilt auch für alle am PCNEWS-Server zugäng-
lichen Daten.)
Werbung A4: 1 Seite 522,- 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.

Impressum

Impressum, Offenlegung

Richtung Auf Anwendungen im Unterricht bezogene Informa-
tionen 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/>
Druck Ultra Print
Pluhová 49, SK-82103 Bratislava
<http://www.ultraprint.eu/>
Versand GZ 02Z031324

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://franz.fiala.cc/>
<http://franz.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: BAWAATWW IBAN: AT741400017710812896
Zugang Einwahl: 0804002222
DNS1/DNS2: 194.50.115.133 86.59.42.66
195.202.152.246
Clublokal HTL, 1030 Wien, Rennweg 89b oder
Gasthaus Kulturschmankerl,
Simmeringer Hauptstraße 152, 1110



Inserenten

MTM-Systeme

Ing. Gerhard Muttenthaler
Hadrawagasse 36 1220 Wien
01-2032814 FAX: 2021313 Handy. 0664-4305636
g.muttenthaler@mtm.at
<http://www.mtm.at/>
Produkte uC/uP-Entwicklungswerkzeuge, Starterkits, Indust-
riecomputer, Netzqualitätsanalyzer, USV-Anlagen
Vertretung Tasking, PLS, Infineon, TQ-Components, Kontron,
Dragnetz-BMI, Panasonic, Dr. Haag, HT-Italia, Dr.
Kaneff
Erreichbar U1-Kagran, 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 ani- mierte Webseiten

Bestellhinweise, Download

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

Das Wort „Alphabet“ in allen Sprachen von Google-Translate. In der voreingestellten Schriftart „Calibri“ können aber nicht alle Zeichen dargestellt werden. Word schaltet in diesen Fällen automatisch auf andere Fonts um, deren Name in der zweiten Spalte angegeben sind.

„Alphabet“	Font	Sprache
Alfabet		Albanisch, Afrikaans, Bosnisch, Dänisch, Indonesisch, Katalanisch, Niederländisch, Polnisch, Rumänisch
字母	SimSun	Chinesisch
الأبجدية		Arabisch
Աբգյուրներն	Sylfaen	Armenisch
əlifba		Aserbaidzhanisch
alfabeto		Baskisch
বর্ণমালা	Vrinda	Bengalisch
Азбука		Bulgarisch, Mazedonisch, Serbisch
Alpabeto		Cebuano, Tagalog
Alphabet		Deutsch, Englisch, Französisch, Javanesisch, Malaysisch, Maori, Mongolisch, Norwegisch, Suaheli, Walisisch
alfabeto		Esperanto, Galizisch, Italienisch, Portugiesisch, Spanisch
Tähestik		Estnisch
aakkoset		Finnisch
აბგუთნო	Sylfaen	Georgisch
αλφάβητο		Griechisch
અલ્ફાબેટ	Shruti	Gujarati
Alfabèt		Haitianisch
Haruffa		Hausa
תּוֹרַת הַבְּרִיאָה		Hebräisch
वर्णमाला	Mangal	Hindi
cov tsiaj ntawv		Hmong
Mkpurɔ Okwu		Igbo
Aibítir		Irish
stafrófið		Isländisch
アルファベット	MS Mincho	Japanisch
טעגסאלפאבעט		Jiddisch
ವರ್ಣಮಾಲೆ	Tunga	Kannada
ஐந்தெழுத்து	DaunPenh	Khmer
알파벳	Batang	Koreanisch
abeceda		Kroatisch, Slowakisch, Slowenisch, Tschechisch
ᠠᠪᠡᠴᠡᠳᠠ	DokChampa	Lao
Abecedarium		Lateinisch
alfabēts		Lettisch
Abécèlè		Litauisch
alfabett		Maltesisch
वर्णमाला	Mangal	Marathisch, Napalesisch
الفبا		Persisch
ਵਰਣਮਾਲਾ	Raavi	Punjabi
Алфавит		Russisch
Alfabetet		Schwedisch
Xarfaha		Somali
நெடுங்கணக்கு	Latha	Tamil
పల్లమాల	Gautami	Telugu
ਭਾਂਡਾਂ		Thailändisch
Alfabe		Türkisch
Алфавіт		Ukrainisch, Weißrussisch
Ábécé		Ungarisch
حروف		Urdu
Bảng chữ cái		Vietnamesisch
Ahbidì		Yoruba
alfabhethi		Zulu

Titelbild

Franz Fiala

Der Hintergrund des Titelbildes ist eine kleine Auswahl der Codebereiche des Unicode, kopiert von <http://wapps.clubcomputer.at/code/> -> [Codepages](#) -> [Vergleich](#)

Über dem Hintergrund schwebt das Wort „Alphabet“ in 51 verschiedenen Schreibweisen, übersetzt mit Google-Translate.

Google Translate

In insgesamt 80 Sprache kann Google-Translate <http://translate.google.com> einen Text übersetzen und kann dabei auch die Sprache, in der ein Text verfasst wurde, erraten.

Hilfreich bei dieser Übersetzung ist natürlich, dass das angewendete Alphabet durch den Unicode eindeutig feststeht und nicht—wie in der Vor-Unicode-Ära—auch noch erraten werden muss.

Die Übersetzung hat aber so ihre Schwächen, weil Englisch als Zwischensprache verwendet wird. Übersetzungen von und nach Englisch sind daher deutlich besser als Übersetzungen zwischen anderen Sprachen.

Beispiel: Im Französischen heißt „naître“ „gebären“, wird aber bei Übersetzung ins Deutsche mit „Frühling“ übersetzt. Wie kommt es zu solchen sonderbaren Übersetzungen? Der Grund ist eine Zwischenübersetzung ins Englische, die „naître“ fälschlicherweise als „spring“ übersetzt wird, das eben im nächsten Schritt zu „Frühling“ führt.

Translator Toolkit

Mit dem Translator Toolkit <https://translate.google.com/toolkit/> kann man Texte hochladen und übersetzen lassen und dabei optional auf professionelle (und kostenpflichtige) Hilfe zurückgreifen. Diese vorher geschilderten Übersetzungsmängel können durch eigene Glossare berichtigt werden.

Website-Übersetzer

Mit dem Website-Übersetzer können Webmaster ihre Website in über 60 Sprachen bereitstellen:

<http://translate.google.com/manager/website/>

Ein Teil der vorliegenden Ausgabe beschreibt den „Weltcode“, den „Unicode“, die *Lingua Franca* oder besser *Alphabetum Franca*(?) des digitalen Zeitalters.

Wie alles begann...

Franz Fiala

Wenn man nach kulturellen Gemeinsamkeiten etwa zwischen Nordkorea und Österreich sucht, wird man mehr Trennendes als Verbindendes finden. Es ist aber erstaunlich, dass trotz aller kultureller und religiöser Unterschiede zwischen den Ländern dieser Welt, man sich—mehr oder weniger unbemerkt—auf ein von allen gleichermaßen akzeptiertes elektrisches Biotop beruft, das die Kommunikation kompatibel macht, trotz aller Inkompatibilitäten der politischen Ideen.

Zu diesem Biotop gehören alle Normen, die die Übertragung und Verarbeitung analoger und digitaler Signale und Informationen ermöglichen, beginnend bei der Definition von dem, was elektrisch Null oder Eins ist, bis hin zu komplexen protokollarischen Fragen im Internet.

Dort, wo einander Technik und Kultur erstmals begegnen, wird es besonders spannend. Denn wozu sollten die vielen Vereinbarungen über Bits und Bytes schon dienen, wenn nicht der Kommunikation zwischen Menschen. Und diese wollen natürlich ihr Alphabet und ihre Sprache bei der Kommunikation mit den elektronischen Systemen verwenden.

Nach den grundlegenden Entdeckungen der Pioniere der Elektrotechnik war der Weg zur zunächst drahtgebundenen Übertragung von Zeichen geebnet. Es begann 1837 mit dem Morse-Code (Bild unten), dessen aktuelle Fassung die ITU-Empfehlung M.1677-1 aus 10/2009

T -	M - -	O - - -	CH - - - -	
			Ö - - - .	
	N - .	G - - .	Q - - . .	
			Z - - . .	
		K - . -	Y - . - -	
			C - . . .	
		X - . . -		
		D - . .	B - . . .	
~~~~~ Morse-Code ~~~~~				
E .	A . -	W . - -	J . - - -	
			P . - - .	
	R . . -	Ä . - . -		
			L . - . .	
		U . . -	Ü . . - -	
				F . . . .
	I . .	V . . . -		
		S . . .	H . . . .	

CCITT-2		
Code	Buchstabe	Ziffer/Zeichen
00011	A	-
11001	B	?
01110	C	:
01001	D	Wer Da?
00001	E	3
01101	F	
11010	G	
10100	H	
00110	I	8
01011	J	Klingel
01111	K	(
10010	L	)
11100	M	.
01100	N	,
11000	O	9
10110	P	0
10111	Q	1
01010	R	4
00101	S	,
10000	T	5
00111	U	7
11110	V	=
10011	W	2
11101	X	/
10101	Y	6
10001	Z	+
01000	Wagenrücklauf	
00010	Zeilenvorschub	
00100	Zwischenraum	
11111	Buchstaben	
11011	Ziffern/Zeichen	
00000		

Zeichenvorrat	Kode- raum	Kodierung (siehe Seite der Codepages)
Zahlen	16	BCD, Excess-3, AIKEN...
Lat. Alphabet, Zahlen	58	Telegrafenalphabet (Baudot-Code, CCITT-2)
Lat. Alphabet, Zahlen, Zeichen, Steuerzeichen	128	ASCII
Lat. Alphabet + westeurop. Zeichenformen, Zahlen, Zeichen, Steuerzeichen	256	EBCDIC, Latin-1, DOS, MAC
Unicode: Viele Alphabete und Zeichensammlungen	253440	UTF-32, UTF-16, UTF-8, UTF-7

ist. Dass man den Morsekode wie eine Sprache erlernen musste, schränkte seine Verwendbarkeit auf Teilbereiche der Kommunikation ein. Heute ist es dank geeigneter Programme durchaus möglich, auch ohne Morsekennntnisse einen „Dialog im Morse-Code“ zu führen.

Der Morse-Code ordnete die häufig verwendeten Zeichen den einfachen Mustern aus Punkt und Strich zu (t, m, o, e, i, s).

Zehn Jahre nach dem Morse-Code wurde der maschinell verarbeitbare Baudot-Code erfunden, später genormt als CCITT-1 und CCITT-2. Dieser Code ist uns in der Form des Fernschreibers und der Telegramme noch in Erinnerung. Der Telegrammdienst war zwischen 1847 und 2005 aktiv. (Bild links)

Dieser Baudot-Code war für die elektromechanische Verarbeitung optimiert, in dem Sinn, dass die Einsen, die die Mechanik in Bewegung zu setzen hatten (bei einer Null bewegte sich einfach nichts) bei häufigen Zeichen möglichst nur einmal im Kodewort vorkamen (a, e, y (englisch!), 1, 2, 3, SP-Zahl, SP-Buchstabe) usw. Weiters beschränkte man den Code auf 5 Bit und schaltete zwischen Buchstaben und Ziffern/Zeichen um, damit die Anzahl der mechanischen Bauteile möglichst gering bleiben konnte,

Mit Beginn des Computerzeitalters hat der Baudot-Code ausgedient, an seine Stelle trat in Teilen des Großrechnerbereichs der EBCDIC-Code und bei den damaligen Minicomputern und in der Telekommunikation seit 1963 der US-ASCII-Code (CCITT-Empfehlung T.50, 9/92).

## Beginn des Computerzeitalters

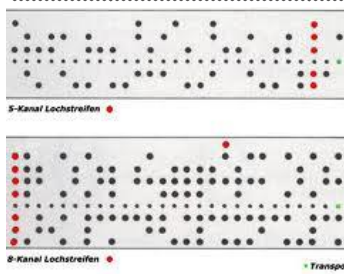
Was gut für die lange Haltbarkeit der Fernschreiber war, war für die elektronische Weiterverarbeitung weniger gut geeignet. Verglichen mit dem Baudot-Code war der ASCII-Code für die elektronische Verarbeitung viel besser geeignet, weil die Buchstaben und Zahlen auf aufeinanderfolgenden Kodeplätzen angeordnet waren und daher ganz einfach nach dem Alphabet sortiert werden konnte. Außerdem gab es endlich auch Kleinbuchstaben (Tabelle nächste Seite).

Das verbreitetste Ein- und Ausgabegerät war der Fernschreiber „Teletype 33 ASR“ mit Eingabe über Tastatur, Ausgabe auf Endlospapier und Ein- und Ausgabe über einen 8-Bit-Lochstreifen, wobei das 8te Bit als Prüfbit verwendet werden konnte (Bild unten).

Auch der ASCII-Code hat noch ein Zeichen, das eben wegen dieser mechanischen Verarbeitung seine typische Stelle 127 (=111.1111) im Koderaum hat. Es war das Steuerzeichen DEL, mit dem man einen Fehler bei der Eingabe korrigieren konnte, denn der Code für DEL bestand aus lauter Einsen und bewegte den Lochstreifen nicht vorwärts sondern überschrieb das falsch eingegebene Zeichen.

Diese Zeit des ASCII-Code war geprägt von Unannehmlichkeiten für Benutzer von Sprachen mit „besonderen Bedürfnissen“, wie eben der Umlaute im Deutschen. Viele der PCNEWS-Leser werden sich noch an die Zeit vor dem PC erinnern, als es einen ASCII-Zeichensatz gab und man den Drucker so einstellen konnte, dass statt der Zeichen [ \ ] { } ~ die Zeichen Ä Ö Ü ä ö ü ausgegeben wurden. Diesen Zeichensatz gibt es immer noch, er hat die Codepage 20106. In der damaligen Enge der 128 möglichen Zeichen musste man für den Druck der Umlaute besondere Sequenzen an den Drucker senden, um diesem einen anderen Zeichensatz einzureden.

## 5-Bit- und 8-Bit-Lochstreifen und Teletype



## Am Anfang war das Chaos

Wegen der grundsätzlichen Beschränkung auf 7 bzw. 8 Bit musste für andere Zeichen der Zeichensatz umgeschaltet werden. Wollte man Zeichen aus einer anderen Codepage verarbeiten, musste man mit diesen Codepages „jonglieren“.

Ein wichtiger Mangel war, dass man es einer Textdatei nicht angesehen hat, welche Codepage anzuwenden war. Wer sich also in mehreren Kulturen bewegte, musste sich in diesen Belangen der Codepages gut auskennen.

Und es braucht ein gewisses „Chaos“, das unsere Arbeit behindert und die uns zwingt, Auswege in einer neuen Ordnung zu suchen.

Mit dem PC kam mit der Codepage 437 (in der Variante 850 und 852 für West- bzw. Mitteleuropa) eine gewisse Entspannung, denn man konnte die häufigsten Zeichen ohne Zeichensatz-Umschaltung verarbeiten, sofern man sich nicht aus dem eigenen Kulturkreis hinausbewegte (Tabelle rechts).

In weiteren Varianten 720 (arabisch) bis 869 (griechisch) konnten bereits sehr viele Sprachen abgedeckt werden. Aber immer musste auch hier die Codepage umgeschaltet werden.

Allerdings war der damalige DOS-Zeichensatz ein gut gemeintes Sammelsurium unvollständiger Zeichensatzteile und keiner dieser Belange war wirklich ordentlich abgedeckt. Den Zeichensatz kann man immer noch erleben, indem man in Windows eine Command-Shell ausführt: `Start -> Ausführen -> cmd.exe`; jetzt mit `type datei.bin` irgendeine Binärdatei ausgeben. Typisch sind die in diesem Zeichensatz auftretenden Grafikzeichen.

Mit Windows kam die Codepage 1252, denn auf die Grafik-Zeichen des DOS-Zeichensatzes wurde zugunsten einer umfassenderen Darstellung der lateinischen Sprachen verzichtet, weil durch die Grafikfähigkeit von Windows die Blockgrafik an Bedeutung verloren hat. Später wurde diese Codepage als ISO 8859-1, „Latin-1“ bekannt. Der Unterschied zwischen Windows-1252 und ISO 8859-1 sind die in Windows-1252 verwendeten Zeichen zwischen den Positionen 128 und 159 (Tabelle rechts).

## Geburtsstunde des Unicode

Die Erfolgsgeschichte der Vereinheitlichung aller Zeichen begann 1991, 154 Jahre nach der Erfindung des Morse-Code, mit der Verabschiedung der ersten Version des Unicode und wurde erstmals in Windows NT 3.0 1993 implementiert und ist seither fester Bestandteil unserer Rechner. Die wichtigste Neuerung des Unicode bestand darin, zwischen der Katalogisierung der Zeichen und dem Kode zu unterscheiden, denn bis dahin gab es einen solchen universellen Zeichenkatalog nicht.

Praktisch alle bisherigen Codepages, inklusive der Blockgrafikzeichen und der Dingbat-Zeichen fanden ihren Platz in der Ebene 0 (Plane 0 = Basic Multilingual Plane). Der US-ASCII-Code findet seinen prominenten Platz gleich am Beginn des Koderaums, was den Vorteil hat, dass es eine 100%ige Kompatibilität für Internet-Protokolle und gleichzeitiger UTF-8-Kodierung des Unicode gibt.

In den nebenstehenden Tabellen sind die neuen CodeBereiche rot angemerkt.

Eigentlich ist die Bezeichnung „Unicode“ etwas irreführend, denn der Unicode definiert in erster Linie die Zeichen, ist also ein Zeichenkatalog; er definiert aber zunächst nicht die Art, wie diese Zeichen in Bytefolgen kodiert werden. Mir gefiele etwas wie „Unialphabet“ oder „Unicodemap“ besser.

Es ist aber interessant, dass diese Trennung zwischen Zeichenkatalog und Kodierung nicht vollständig ist, denn durch die besondere Art der UTF-16-Kodierung wurde es erforderlich, Codebereiche von Zeichen freizuhalten.

Seit dem Jahr 1993 erleben wir ein Schlaraffenland der Zeichen. In der PCNEWS-29, November 1992, berichteten wir auf Seite 37 „Unicode, eine Revolution beginnt“. Das Betriebssystem Windows NT verwendete eine 16-Bit-Zeichen-Darstellung, die es erstmals erlaubte, in einem Dokument praktisch alle Zeichen gleichzeitig zu verwenden und nicht - so wie vor dieser Zeit - bei der Korrespondenz mit dem Ausland immer die richtige Codepage eingestellt zu haben, um auch die verschiedenen Sonderzeichen richtig dargestellt zu bekommen.

<http://de.scribd.com/doc/151672642/PCNEWS-29>

Der Unicode 1.0 war ein 16-Bit-Kode und erlaubte die Verwendung von  $2^{16}=65536$  Zeichen. Dieser grundlegende Zeichenrahmen wird als *Basic Multilingual Plane* bezeichnet.

Ab Unicode-Version 3.0 wurde dieser grundlegende Zeichensatz in einigen Etappen um weitere 17 Planes erweitert. Daher erfordert

## US-ASCII (American Standard Code for Information Interchange)

Hex	_0	_1	_2	_3	_4	_5	_6	_7	_8	_9	_A	_B	_C	_D	_E	_F
0	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2	SP	!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	DEL

## DOS (CodePage 437)

Hex	_0	_1	_2	_3	_4	_5	_6	_7	_8	_9	_A	_B	_C	_D	_E	_F
8	00c0	é	â	ä	à	ç	ê	ë	è	ï	î	ï	Ä	Å		
9	É	æ	Æ	ô	ö	û	ü	ÿ	Ö	Ü	20a0	¥	£	f		
a	ā	í	ó	ú	ñ	Ñ	ª	º	¿	¬	2150	ı	«	»		
b	2500	■														
c	L	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥
d	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥	⊥
e	0370	Γ	π	Σ	σ	μ	τ	Φ	Θ	Ω	δ	∞	φ	ε	∩	
f	2200	≥	≤	∫	∫	÷	≈	°	°	√	n	2	■			

## Windows (CodePage 1252, ~ISO 8859-1)

Hex	_0	_1	_2	_3	_4	_5	_6	_7	_8	_9	_A	_B	_C	_D	_E	_F
8	€	□	,	f	„	...	†	‡	ˆ	‰	Š	‹	ƒ	□	Ž	•
9	‚	‚	„	„	•	—	—	˜	™	š	›	œ	□	ž	ÿ	
a	□	ı	¢	£	¤	¥	¦	§	¨	©	ª	«	¬	®	¯	
b	°	±	²	³	´	µ	¶	·	¸	¹	º	»	¼	½	¾	¿
c	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
d	Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß
e	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
f	ð	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ý	þ	ÿ

die Benennung eines Zeichens in diesen Erweiterungsplanes 21 statt der ursprünglichen 16 Bits des ursprünglichen Zeichensatzes.

Das Unicode-Konsortium <http://www.unicode.org/> arbeitete seit den Anfängen mit der internationalen Normungsorganisation ISO eng zusammen. Der Unicode heißt dort ISO 10646. Unicode enthält in Teilbereichen mehr Zeichendefinitionen als die ISO-Definition. Praktisch alle Staaten der Welt sind über ihre nationalen Normungsinstitutionen Mitglied bei ISO. ISO wieder kooperiert eng mit den UNO-Teilorganisationen IEC und ITU zusammen.

Wenn es also politisch noch keinen Superstaat gibt, auf dem Gebiet der Normung hat der „Weltcode“ bereits seit 1993 Einzug gehalten und hat sich nahezu unbemerkt (und so muss es auch sein, um die Benutzer nicht zu verunsichern) auf allen unseren EDV-Geräten etabliert.

So sehr Religionen und Kulturen bemühen, sich abzugrenzen und Menschen (trotz aller gegenteiliger Beteuerungen) auseinander zu dividieren, so integrativ wirkt Technik rund um den Globus. Unbemerkt für den Benutzer und doch sonderbar verbindend.

Uns fremd erscheinende Kopftuch- oder Turban-Träger sind uns gleich viel vertrauter, wenn sie ihr i-, Android- oder Windows-Phone benutzen. Da scheinen sich die Menschen nämlich einig zu sein. Ohne die Technik, geht gar nichts. Die Technik baut Brücken, die verfeindete Menschen oft nicht einmal wahrnehmen.

# Codepages in Windows

Franz Fiala

Codepages gibt es seit es digitale Kodierung gibt. Egal, welcher Code verwendet wird, es erfordert immer eine Zuordnung zwischen einer Zeichentabelle und dem verarbeiteten Kode.

Vor dem Unicode, also etwa vor 1993 gab es unzusammenhängende Zeichensammlungen, wie zum Beispiel US-ASCII, Latin-1 und andere. Ein bestimmter Code konnte—je nach Codepage—ein anderes Zeichen bedeuten. Einer Textdatei war es nicht anzumerken, welcher Code angewendet wurde, denn eine Textdatei enthielt nur die Zeichen aber keinen Hinweis auf die Kodierungsvorschrift.

Die Zeichentabelle definiert eine Auswahl von Zeichen, die auf den Koderaum abgebildet werden. In unseren Breiten waren das immer 8-Bit-Kodes, im asiatischen Raum waren es immer 16-Bit-Kodes oder Umschaltkodes. Diese Umschaltkodes benutzen zwei Zeichentabellen, eine mit 256 Zeichen und eine mit 65536

Zeichen. Mit reservierten Umschaltzeichen (ESC-Sequenzen) wird zwischen den beiden Zeichenräumen umgeschaltet.

Der Unicode machte mit diesen Mehrdeutigkeiten ein Ende, weil der Versuch unternommen wurde, jedes Zeichen auf diesem Planeten in einem gemeinsamen Zeichenraum zu erfassen.

Der Unicode unterscheidet erstmals zwischen der Definition der Zeichen durch so genannte CodePoints und der Kodierung. Der CodePoint ist eine Hexadezimalzahl, die die Zeichenposition im Koderaum des Unicode beschreibt. Die Kodierung sagt, wie diese Zahl in eine Folge von Bytes übergeführt wird.

Die folgende Tabelle zeigt die in Windows verfügbaren Codepages, die Unicode-Pages sind hervorgehoben.

Die Unicode-Codepages sind also eigentlich Vorschriften zur Kodierung der Unicode-CodePoints.

BR	M	Bit8	Disp	Uni	Bit	SL	WebName	CodePge	Name	-	-	+	+	-	8	-	x-ia5-german	20106	Deutsch (IA5)
-	-	+	+	-	8	-	ibm037	37	IBM EBCDIC (USA-Kanada)	-	-	+	+	-	8	-	x-ia5-swedish	20107	Schwedisch (IA5)
-	-	+	+	-	8	+	ibm437	437	OEM USA	-	-	+	+	-	8	-	x-ia5-norwegian	20108	Norwegisch (IA5)
-	-	+	+	-	8	-	ibm500	500	IBM EBCDIC (International)	-	+	+	+	-	8	-	us-ascii	20127	US-ASCII
+	-	+	+	-	8	-	asmo-708	708	Arabisch (ASMO 708)	-	-	-	-	-	8	-	x-cp20261	20261	T.61
+	-	+	+	-	8	-	dos-720	720	Arabisch (DOS)	-	-	-	+	-	8+16	-	x-cp20269	20269	ISO-6937
-	-	+	+	-	8	-	ibm737	737	Griechisch (DOS)	-	-	-	+	-	8	-	ibm273	20273	IBM EBCDIC (Deutschland)
-	-	+	+	-	8	-	ibm775	775	Baltisch (DOS)	-	-	-	+	-	8	-	ibm277	20277	IBM EBCDIC (Dänemark-Norwegen)
-	-	+	+	-	8	+	ibm850	850	Westeuropäisch (DOS)	-	-	-	+	-	8	-	ibm278	20278	IBM EBCDIC (Finnland-Schweden)
+	-	+	+	-	8	+	ibm852	852	Mitteeuropäisch (DOS)	-	-	-	+	-	8	-	ibm280	20280	IBM EBCDIC (Italien)
-	-	+	+	-	8	-	ibm855	855	OEM Kyrrilisch	-	-	-	+	-	8	-	ibm284	20284	IBM EBCDIC (Spanien)
-	-	+	+	-	8	-	ibm857	857	Türkisch (DOS)	-	-	-	+	-	8	-	ibm285	20285	IBM EBCDIC (UK)
-	-	+	+	-	8	-	ibm00858	858	OEM Multilingual Lateinisch 1	-	-	-	+	-	8	-	ibm290	20290	IBM EBCDIC (Japanisch Katakana)
-	-	+	+	-	8	-	ibm860	860	Portugiesisch (DOS)	-	-	-	+	-	8	-	ibm297	20297	IBM EBCDIC (Frankreich)
-	-	+	+	-	8	-	ibm861	861	Isländisch (DOS)	-	-	-	+	-	8	-	ibm420	20420	IBM EBCDIC (Arabisch)
+	-	+	+	-	8	-	dos-862	862	Hebräisch (DOS)	-	-	-	+	-	8	-	ibm423	20423	IBM EBCDIC (Griechisch)
-	-	+	+	-	8	-	ibm863	863	Französisch, Kanada (DOS)	-	-	-	+	-	8	-	ibm424	20424	IBM EBCDIC (Hebräisch)
-	-	+	+	-	8	-	ibm864	864	Arabisch (864)	-	-	-	+	-	8	-	*)	20833	IBM EBCDIC (Koreanisch, erweitert)
-	-	+	+	-	8	-	ibm865	865	Nordisch (DOS)	-	-	-	+	-	8	-	*)	20833	*) x-ebcdic-koreanextended
+	-	+	+	-	8	-	cp866	866	Kyrrilisch (DOS)	-	-	-	+	-	8	-	ibm-thai	20838	IBM EBCDIC (Thailändisch)
-	-	+	+	-	8	-	ibm869	869	Griechisch, modern (DOS)	+	+	-	+	-	8	-	koi8-r	20866	Kyrrilisch (KOI8-R)
-	-	+	+	-	8	-	ibm870	870	IBM EBCDIC (Multilingual Latin-2)	-	-	-	+	-	8	-	ibm871	20871	IBM EBCDIC (Isländisch)
+	+	+	+	-	8	-	windows-874	874	Thailändisch (Windows)	-	-	-	+	-	8	-	ibm880	20880	IBM EBCDIC (Kyrrilisch, Russisch)
-	-	+	+	-	8	-	cp875	875	IBM EBCDIC (Griechisch, modern)	-	-	-	+	-	8	-	ibm905	20905	IBM EBCDIC (Türkisch)
+	+	-	-	-	8+16	-	shift_jis	932	Japanisch (Shift-JIS)	-	-	-	+	-	8	-	ibm00924	20924	IBM Lateinisch-1
+	+	-	-	-	8+16	-	gb2312	936	Chinesisch vereinfacht (GB2312)	-	-	-	+	-	8	-	euc-jp	20932	Japanisch (JIS 0208-1990 und 0212-1990)
+	+	-	-	-	8+16	-	ks_c_5601-1987	949	Koreanisch	-	-	-	-	-	-	-	x-cp20936	20936	GB2312-80 Chinesisch (vereinfacht)
+	+	-	-	-	8+16	-	big5	950	Chinesisch traditionell (Big5)	-	-	-	-	-	-	-	x-cp20949	20949	Koreanisch Wansung
-	-	+	+	-	8	-	ibm1026	1026	IBM EBCDIC (Türkisch, Lateinisch-5)	-	-	-	+	-	8	-	cp1025	21025	IBM EBCDIC (Kyrrilisch, Serbisch-Bulgarisch)
-	-	+	+	-	8	-	ibm01047	1047	IBM Lateinisch-1	-	-	-	+	-	8	-	*)	21866	Kyrrilisch (KOI8-U)
-	-	+	+	-	8	-	ibm01140	1140	IBM EBCDIC (USA-Kanada-Europäisch)	+	+	+	+	-	8	-	iso-8859-1	28591	Westeuropäisch (ISO)
-	-	+	+	-	8	-	ibm01141	1141	IBM EBCDIC (Deutschland-Europäisch)	+	+	+	+	-	8	+	iso-8859-2	28592	Mitteeuropäisch (ISO)
-	-	+	+	-	8	-	ibm01142	1142	IBM EBCDIC (DK, N, EUR)	+	+	+	+	-	8	+	iso-8859-3	28593	Lateinisch 3 (ISO)
-	-	+	+	-	8	-	ibm01143	1143	IBM EBCDIC (F, S, EUR)	+	+	+	+	-	8	-	iso-8859-4	28594	Baltisch (ISO)
-	-	+	+	-	8	-	ibm01144	1144	IBM EBCDIC (I, EUR)	+	+	+	+	-	8	-	iso-8859-5	28595	Kyrrilisch (ISO)
-	-	+	+	-	8	-	ibm01145	1145	IBM EBCDIC (ES, EUR)	+	+	+	+	-	8	-	iso-8859-6	28596	Arabisch (ISO)
-	-	+	+	-	8	-	ibm01146	1146	IBM EBCDIC (GB, EUR)	+	+	+	+	-	8	-	iso-8859-7	28597	Griechisch (ISO)
-	-	+	+	-	8	-	ibm01147	1147	IBM EBCDIC (F, EUR)	+	+	+	+	-	8	-	iso-8859-8	28598	Hebräisch (ISO-Visual)
-	-	+	+	-	8	-	ibm01148	1148	IBM EBCDIC (INT, EUR)	+	+	+	+	-	8	-	iso-8859-9	28599	Türkisch (ISO)
-	-	+	+	-	8	-	ibm01149	1149	IBM EBCDIC (ISL, EUR)	+	+	+	+	-	8	-	iso-8859-13	28603	Estnisch (ISO)
+	-	-	+	+	16	+	utf-16	1200	Unicode	+	+	+	+	-	8	-	iso-8859-15	28605	Lateinisch 9 (ISO)
-	-	-	+	+	16	+	utf-16be	1201	Unicode (Big-Endian)	+	+	+	+	-	8	-	x-europa	29001	Europa
+	+	+	+	-	8	+	windows-1250	1250	Mitteeuropäisch (Windows)	+	+	+	+	-	8	-	iso-8859-8-i	38598	Hebräisch (ISO-Logical)
+	+	+	+	-	8	+	windows-1251	1251	Kyrrilisch (Windows)	-	-	-	+	-	8	-	iso-2022-jp	50220	Japanisch (JIS)
+	+	+	+	-	8	+	windows-1252	1252	Westeuropäisch (Windows)	+	+	+	+	-	8	-	csiso-2022-jp	50221	Japanisch (JIS, 1 Byte Kana erlaubt)
+	+	+	+	-	8	-	windows-1253	1253	Griechisch (Windows)	-	-	-	+	-	8sw	-	iso-2022-jp	50222	Japanisch (JIS, 1 Byte Kana erlaubt - SO/SI)
+	+	+	+	-	8	-	windows-1254	1254	Türkisch (Windows)	+	+	-	-	-	8sw	-	iso-2022-jp	50222	
+	+	+	+	-	8	-	windows-1255	1255	Hebräisch (Windows)	-	-	-	-	-	8sw	-	iso-2022-jp	50222	
+	+	+	+	-	8	-	windows-1256	1256	Arabisch (Windows)	-	-	-	-	-	8sw	-	iso-2022-kr	50225	Koreanisch (ISO)
+	+	+	+	-	8	-	windows-1257	1257	Baltisch (Windows)	-	+	-	-	-	8sw	-	x-cp50227	50227	Chinesisch vereinfacht (ISO-2022)
+	+	+	+	-	8	-	windows-1258	1258	Vietnamesisch (Windows)	-	+	-	-	-	8+	-	euc-jp	51932	Japanisch (EUC)
-	-	-	-	-	16	-	johab	1361	Koreanisch (Johab)	-	+	-	-	-	8+	-	euc-cn	51936	Chinesisch vereinfacht (EUC)
-	-	+	+	-	8	+	macintosh	10000	Westeuropäisch (Mac)	-	-	-	-	-	8+	-	euc-kr	51949	Koreanisch (EUC)
-	-	-	-	-	16	-	x-mac-japanese	10001	Japanisch (Mac)	-	+	-	-	-	8+	-	hz-gb-2312	52936	Chinesisch vereinfacht (HZ)
-	-	-	-	-	16	-	*)	10002	Chinesisch traditionell (Mac)	+	+	-	-	-	8+	-	gb18030	54936	Chinesisch vereinfacht (GB18030)
-	-	-	-	-	16	-	x-mac-korean	10003	*) x-mac-chinesetrad	+	+	-	-	-	8+	-	x-iscii-de	57002	ISCII Devanagari
-	-	-	-	-	16	-	x-mac-arabic	10004	Koreanisch (Mac)	-	-	-	-	-	8	-	x-iscii-be	57003	ISCII Bangla
-	-	-	-	-	16	-	x-mac-hebrew	10005	Arabisch (Mac)	-	-	-	-	-	8	-	x-iscii-ta	57004	ISCII Tamil
-	-	-	-	-	16	-	x-mac-greek	10006	Hebräisch (Mac)	-	-	-	-	-	8	-	x-iscii-te	57005	ISCII Telugu
-	-	-	-	-	16	-	x-mac-cyrillic	10007	Griechisch (Mac)	-	-	-	-	-	8	-	x-iscii-as	57006	ISCII Assamesisch
-	-	-	-	-	16	-	*)	10008	Kyrrilisch (Mac)	-	-	-	-	-	8	-	x-iscii-or	57007	ISCII Oriya
-	-	-	-	-	16	-	*)	10008	Chinesisch vereinfacht (Mac)	-	-	-	-	-	8	-	x-iscii-ma	57008	ISCII Kannada
-	-	-	-	-	16	-	*)	10008	*) x-mac-chinesesimp	-	-	-	-	-	8	-	x-iscii-gu	57009	ISCII Malayalam
-	-	-	-	-	16	-	x-mac-romanian	10010	Rumänisch (Mac)	-	-	-	-	-	8	-	x-iscii-pa	57011	ISCII Gujarati
-	-	-	-	-	16	-	x-mac-ukrainian	10017	Ukrainisch (Mac)	-	-	-	-	-	8	-	*)	65000	Unicode (UTF-7)
-	-	-	-	-	16	-	x-mac-thai	10021	Thailändisch (Mac)	-	-	-	-	-	8	-	*)	65001	Unicode (UTF-8)
-	-	-	-	-	16	-	x-mac-ce	10029	Mitteeuropäisch (Mac)	+	+	-	-	+	8(7)	+	utf-7		
-	-	-	-	-	16	-	x-mac-icelandic	10079	Isländisch (Mac)	+	+	-	-	+	8-32	+	utf-8		
-	-	-	-	-	16	-	x-mac-turkish	10081	Türkisch (Mac)	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	16	-	x-mac-croatian	10082	Kroatisch (Mac)	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	32	+	utf-32	12000	Unicode (UTF-32)	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	32	+	utf-32be	12001	Unicode (UTF-32 Big-Endian)	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	16	-	x-chinese-cns	20000	Chinesisch traditionell (CNS)	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	16	-	x-cp20001	20001	TCA Taiwan	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	16	-	x-chinese-eten	20002	Chinesisch traditionell (Eten)	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	16	-	x-cp20003	20003	IBM5550 Taiwan	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	16	-	x-cp20004	20004	TeleText Taiwan	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	16	-	x-cp20005	20005	Wang Taiwan	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	16	-	x-ia5	20105	Westeuropäisch (IA5)	-	-	-	-	-	-	-	-	-	-

Diese Codepages werden von Windows verarbeitet.  
**BR** kann von einem Browser im Rahmen einer Html-Seite verarbeitet werden  
**M** kann vom SMTP- oder NNTP-Protokoll verarbeitet werden  
**Bit8** Ein 8-Bit-Kode  
**Disp** kann von wapps.clubcomputer.at/CODE angezeigt werden  
**Uni** eine Kodierung des Unicode  
**Bit** Bit pro Zeichen  
**SL** häufig verwendete Codepage  
**Webname** diese Bezeichnung wird in Html-Dateien verwendet (z.B. „charset=iso-8859-1“)  
**CodePage** diese Zahl wird in Programmen zur Einstellung der CodePage verwendet





# Unicode

Franz Fiala

## Die Welt unter einem Dach

Die Weltkarte am unteren Seitenrand zeigt die Verbreitung der verschiedenen Alphabete.

Das Bild rechts zeigt den Koderaum des Unicode. Der Koderaum besteht aus 17 Ebenen (Planes), die jede für sich 65536 (also  $2^{16}$ ) Zeichen (CodePoints) enthalten kann.

Die wichtigsten Alphabete dieser Welt sind in der „Basic Multilingual Plane“ enthalten, die mit einem 16-Bit-Code vollständig adressierbar ist.

Praktisch alle unsere Texte, die in Textverarbeitungssystemen produziert werden, enthalten Zeichen aus der BMP, der ersten Ebene (Plane) des Unicode.

Die Tabelle rechts gibt eine numerische Übersicht über den gesamten Unicode-Raum. Der Koderaum enthält demnach noch 11 derzeit unbenutzte Planes.

## Kodierung

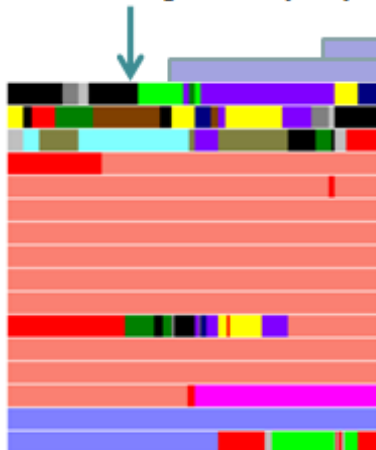
Der Unicode definiert Zeichen und deren zahlenmäßige Zeichenpositionen (CodePoints).

Die Kodierung legt fest, wie diese Zeichenpositionen in Bytes abgebildet werden.

Für die lineare Adressierung sind 21 Bit erforderlich. Die Kodierung, die eine solche lineare Adressierung ermöglicht, ist der UTF-32-Code.

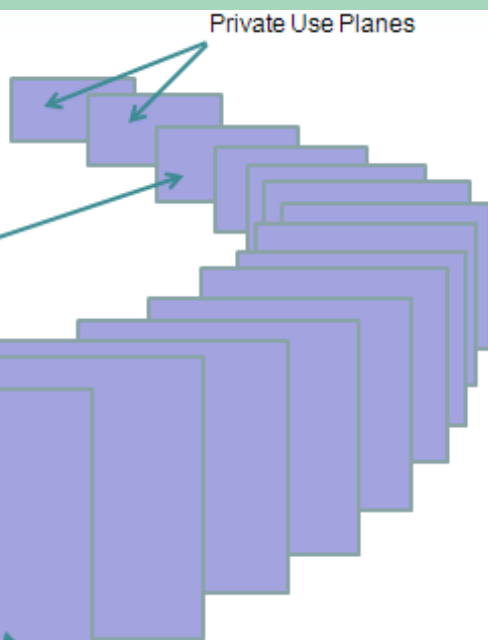


## Basic Multilingual Plane (BMP)



65,536 code points

## Supplementary Special Use Plane

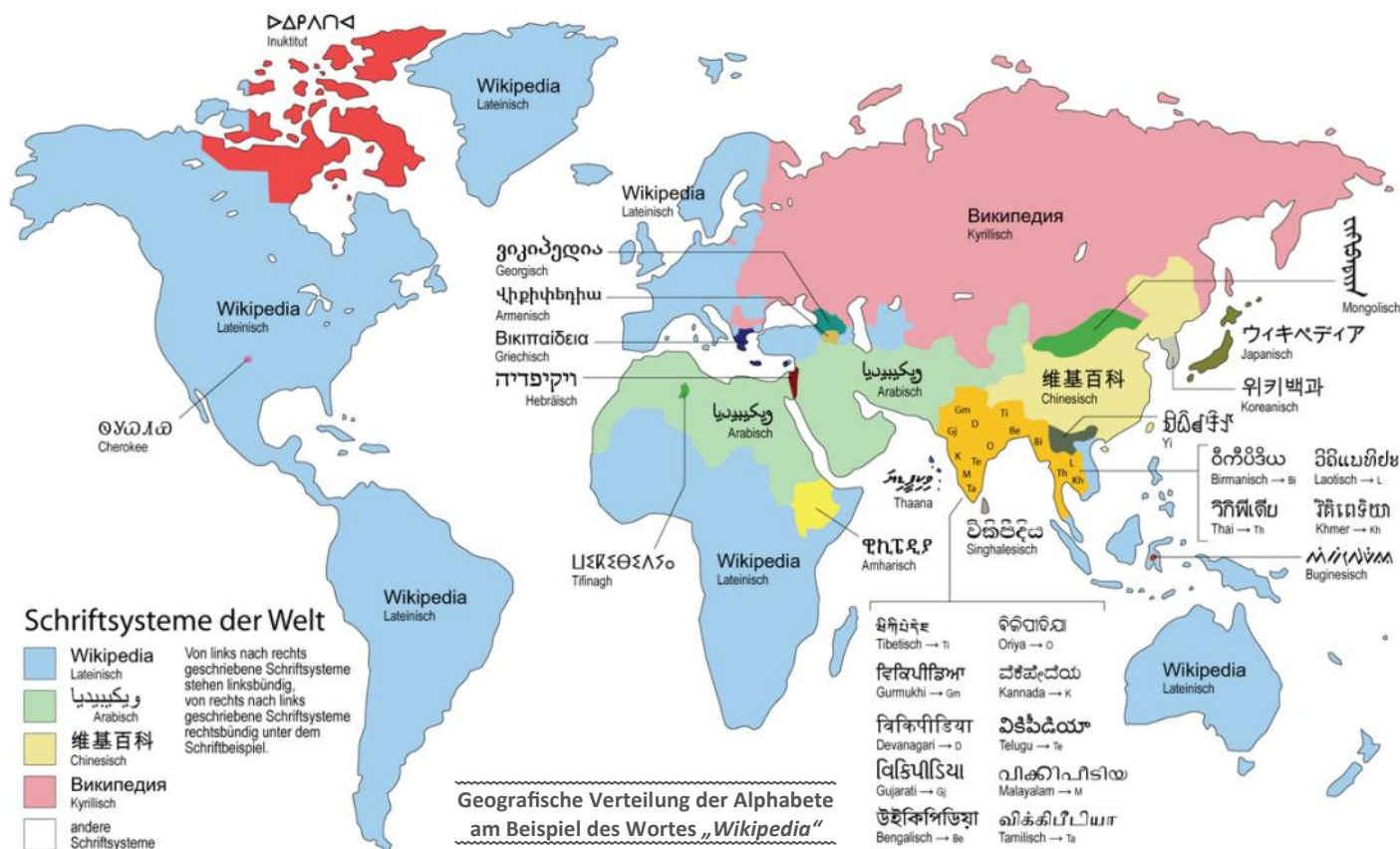


Die 17 Ebenen (Planes) des Unicode

SIP (Supplementary Ideographic Plane)

SMP (Supplementary Multilingual Plane)

Basic		Supplementary			
Plane 0	Plane 1	Plane 2	Plane 3-13	Plane 14	Plane 15-16
Basic Multilingual Plane	Supplementary Multilingual Plane	Supplementary Ideographic Plane	Unassigned	Supplementary Special Purpose Plane	Supplementary Private Use Area
BMP	SMP	SIP	SIP	SSP	S PUA A/B
0000 - FFFF	10000 - 1FFFF	20000 - 2FFFF	30000 - DFFFF	E0000 - EFFFF	F0000 - 10FFFF
0000-0FFF 1000-1FFF 2000-2FFF 3000-3FFF 4000-4FFF 5000-5FFF 6000-6FFF 7000-7FFF	8000-8FFF 9000-9FFF A000-AFFF B000-BFFF C000-CFFF D000-DFFF E000-EFFF F000-FFFF	10000-10FFF 11000-11FFF 12000-12FFF 13000-13FFF 14000-14FFF 15000-15FFF 16000-16FFF 17000-17FFF	18000-18FFF 19000-19FFF 1A000-1AFFF 1B000-1BFFF 1C000-1CFFF 1D000-1DFFF 1E000-1EFFF 1F000-1FFFF	20000-20FFF 21000-21FFF 22000-22FFF 23000-23FFF 24000-24FFF 25000-25FFF 26000-26FFF 27000-27FFF	28000-28FFF 29000-29FFF 2A000-2AFFF 2B000-2BFFF 2C000-2CFFF 2D000-2DFFF 2E000-2EFFF 2F000-2FFFF





## Plane 0 (Basic Multilingual Plane) und Plane 1 (Supplementary Multilingual Plane)

Ein CodeBlock ist ein zusammenhängender Zeichenbereich, in dem Zeichen eines konkreten Sprachraums oder einer konkreten Anwendung (Mathematik, Lautschrift usw.) zusammengefasst sind, meist übernommen von einem früher definierten Zeichensatz.

Die folgenden Bilder zeigen farbkodiert auf einer etwas detaillierteren Landkarte des Unicode wie die einzelnen „Codeblocks“ in den beiden ersten Planes untergebracht sind. Jedes Kästchen enthält 256 (=2⁸) Zeichen. Im Hinblick auf eine effiziente Kodierung ist der erste Codeblock 00 identisch mit dem früheren ASCII-Kode, gleich gefolgt vom so genannte Latin-1-Supplement, sodass dieser Codeblock dem bisher in Windows verwendeten Schema entspricht.

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	
10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F	
20	21	22	23	24	25	26	27	28	29	2A	2B	2C	2D	2E	2F	
30	31	32	33	34	35	36	37	38	39	3A	3B	3C	3D	3E	3F	
40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	
50	51	52	53	54	55	56	57	58	59	5A	5B	5C	5D	5E	5F	
60	61	62	63	64	65	66	67	68	69	6A	6B	6C	6D	6E	6F	
70	71	72	73	74	75	76	77	78	79	7A	7B	7C	7D	7E	7F	
80	81	82	83	84	85	86	87	88	89	8A	8B	8C	8D	8E	8F	
90	91	92	93	94	95	96	97	98	99	9A	9B	9C	9D	9E	9F	
A0	A1	A2	A3	A4	A5	A6	A7	A8	A9	AA	AB	AC	AD	AE	AF	
B0	B1	B2	B3	B4	B5	B6	B7	B8	B9	BA	BB	BC	BD	BE	BF	
C0	C1	C2	C3	C4	C5	C6	C7	C8	C9	CA	CB	CC	CD	CE	CF	
D0	D1	D2	D3	D4	D5	D6	D7	D8	D9	DA	DB	DC	DD	DE	DF	
E0	E1	E2	E3	E4	E5	E6	E7	E8	E9	EA	EB	EC	ED	EE	EF	
F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	FA	FB	FC	FD	FE	FF	
100	101	102	103	104	105	106	107	108	109	10A	10B	10C	10D	10E	10F	
110	111	112	113	114	115	116	117	118	119	11A	11B	11C	11D	11E	11F	
120	121	122	123	124	125	126	127	128	129	12A	12B	12C	12D	12E	12F	
130	131	132	133	134	135	136	137	138	139	13A	13B	13C	13D	13E	13F	
140	141	142	143	144	145	146	147	148	149	14A	14B	14C	14D	14E	14F	
150	151	152	153	154	155	156	157	158	159	15A	15B	15C	15D	15E	15F	
160	161	162	163	164	165	166	167	168	169	16A	16B	16C	16D	16E	16F	
170	171	172	173	174	175	176	177	178	179	17A	17B	17C	17D	17E	17F	
180	181	182	183	184	185	186	187	188	189	18A	18B	18C	18D	18E	18F	
190	191	192	193	194	195	196	197	198	199	19A	19B	19C	19D	19E	19F	
1A0	1A1	1A2	1A3	1A4	1A5	1A6	1A7	1A8	1A9	1AA	1AB	1AC	1AD	1AE	1AF	
1B0	1B1	1B2	1B3	1B4	1B5	1B6	1B7	1B8	1B9	1BA	1BB	1BC	1BD	1BE	1BF	
1C0	1C1	1C2	1C3	1C4	1C5	1C6	1C7	1C8	1C9	1CA	1CB	1CC	1CD	1CE	1CF	
1D0	1D1	1D2	1D3	1D4	1D5	1D6	1D7	1D8	1D9	1DA	1DB	1DC	1DD	1DE	1DF	
1E0	1E1	1E2	1E3	1E4	1E5	1E6	1E7	1E8	1E9	1EA	1EB	1EC	1ED	1EE	1EF	
1F0	1F1	1F2	1F3	1F4	1F5	1F6	1F7	1F8	1F9	1FA	1FB	1FC	1FD	1FE	1FF	

■

 Lateinische Schriften und Symbole

■

 Lautschriften

■

 Andere europäische Schriften

■

 Nahost- und Südwestasiatische Schriften

■

 Afrikanische Schriften

■

 Südasiatische Schriften

■

 Südostasiatische Schriften

■

 Ostasiatische Schriften

■

 CJK-Ideogramme

■

 Kanadische Silben

■

 Symbole

■

 Diakritika

■

 UTF-16-Surrogates und privater Nutzungsbereich

■

 Verschiedene Zeichen

■

 Nicht belegte Codebereiche

■

 Linear B

■

 Ancient Numeric Systems

■

 LTR alphabets and syllabaries

■

 RTL alphabets and syllabaries

■

 Brahmic Scripts

■

 Egyptian Hieroglyphs

■

 Other logographies

■

 Shorthands

■

 Notational Systems

■

 Game Symbols

■

 Mathematical Alphabets and Symbols

■

 Enclosed characters

■

 Emoji and Pictographs

■

 Reserved

■

 Cuneiform scripts

■

 Unallocated code points



## Benennung der Unicode CodePoints

Die CodePoints der BMP (Base Multilingual Plane) werden durch

U+hhhh

benannt, wobei hhhh der Hexwert des betreffenden Zeichens ist. Für die höherwertigen Planes wird die Hexzahl der betreffenden Plane vorangestellt, also

U+1hhhh

für ein Zeichen in der Plane 1.

Es gibt 17 Ebenen (Planes) à 65536 Zeichen, daher insgesamt 1.114.112 CodePoints. Davon sind aber nur ca. 250.000 definiert und mehr als 860.000 noch verfügbar.

U+0000 erster CodePoint in Plane 0

U+100000 letzter CodePoint in Plane 17

Beispiele für Unicode-Codepoints:

A	U+0041	lateinisches A
A	U+0391	griechisches A
A	U+0410	kyrillisches A
א	U+05D0	hebräisches A (Alef)

## Diakritische Zeichen

Diakritische Zeichen sind besondere Auszeichnungen von Buchstaben, die eine abweichende Aussprache des Grundbuchstabens andeuten. Viele dieser Zeichen mit diakritischen Ergänzungen sind bereits als eigene Zeichen im Unicode definiert.

Etwa finden wir ab U+0080, dem Latin-1 Supplement, an der Stelle U+00C4 das Zeichen Ä, Umlaut-A. Aber nicht alle dieser möglichen Zeichenkombinationen sind auch als eigene Zeichen ausgeführt. Daher kann man diese fehlenden Zeichen durch eine Kombination aus dem Basiszeichen und einem der diakritischen Zeichen komponieren. Im Falle des Umlaut A wäre das das Basiszeichen A (U+0041) und das Zeichen „Combining Diacresis“ (U+0308).

Zeichen Kode

Ä C4 00

Ä 41 00 08 03

Die Bytefolge 41 00, die üblicherweise das Zeichen A darstellt, auch einmal ein Umlaut-A sein kann, wenn nämlich die Byte-Folge für das diakritische Zeichen 08 03 folgt. Ohne den Zusammenhang im Bytestrom zu kennen, kann man daher nicht eindeutig sagen, welches Zeichen mit der Bytefolge 41 00 gemeint ist.

## Kodierung

Eine Kodierung legt nun fest, wie ein Zeichen als eine Folge von Bytes im Speicher oder in einer Datei gespeichert wird. Jede dieser Kodierungen wird als CodePage definiert.

Die naheliegendste Version ist natürlich die, dass man aufeinanderfolgenden Zeichen des Zeichenvorrats auch dieselben Nummern in einem Koderaum zuordnet, damit die bisherigen Nummerierungen, etwa die des ASCII-Kode und anderer Alphabete unverändert dargestellt werden.

In der ersten Definition des Unicode gab es eine Ebene (Plane) und daher 65536 Zeichen. Daher war es naheliegend, für ein Zeichen zwei Bytes zu verwenden. Damit war genau diese Zeichenzahl  $2^{16}$  darstellbar.

## UTF-16

Diesen Code nannte man UTF-16 (*Universal Multiple-Octet Coded Character Set (UCS) Transformation Format*)

Schon bei der Definition des Unicode 1991 war bekannt, dass es weitere Planes geben wird und

daher war auch schon bei der Definition von UTF-16 die Verarbeitung der höherwertigen Planes zu berücksichtigen. UTF-16 kann daher mit zwei Bytes alle Zeichen der BasePlane darstellen und mit einem besonderen Kodierungsverfahren auch alle anderen Zeichen der höherwertigen Planes. Alle Zeichen außerhalb der Base Plane (d. h. U+10000 bis U+10FFFF) werden durch vier Bytes dargestellt.

Der Algorithmus für CodePoints über ffff ist wie folgt:

Der CodePoint hat folgenden höchsten Wert (21 Bit):

U+10ffff

Von diesem Wert wird 10000h (=65536) abgezogen und das nunmehr 20-stellige Ergebnis in zwei Blöcke zu je 10 Bit aufgeteilt.

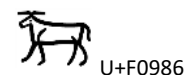
Dem ersten Block wird die Bitfolge 1101.10 (=D8) und dem zweiten die Bitfolge 1101.11 (=DC) vorangestellt.

Alle so entstandenen Codes befinden sich im Zeichenbereich zwischen D8xx (*High Surrogate*) und DCxx (*Low Surrogate*).

Diese Codierung hat zur Folge, dass alle *Code Points* aus den höherwertigen Planes durch zwei Bytes aus dem Bereich D8xx-DCxx kodiert werden. In diesem Bereich der Base Plane sind daher keine Zeichen kodiert.

Beispiel:

Ein ägyptischer Apis-Stier (einer von vielen) hat folgende Hieroglyphe (nur sichtbar mit dem kostenlosen Font Aegyptus):



Wir rechnen:

```
F0986
+10000
-----
E0986 = 1110 0000 10-01 1000 0110
1101.1011 1000 0010 = DB82
1101.1101 1000 0110 = DD86
```

Dieses Zeichen wird daher im UTF-16-Kode durch die zwei Hexzahlen DB82 und DD86 kodiert.

Es ist bemerkenswert, dass diese Codes mit keinem anderen Kode der BasePlane verwechselbar sind, weil ihre Zeichenraum bereits in der BasePlane genau für diesen Zweck reserviert worden ist. Die Definition des Zeichenraumes und der späteren Kodierung hängen also eng zusammen. (siehe Bild unten)

Die UTF-16-Kodierung ist daher eine Kodierung mit ungleicher Länge für Zeichen der Plane 0 und der anderen Planes,

## UTF-32

Mit der Hinzunahme der höherwertigen Planes definierte man den (wegen seines Platzbedarfs eher selten verwendeten) Kode UTF-32, der für die Kodierung von einem CodePoint 4 Bytes=32 Bit benötigt. Dieser Kode erlaubt einen wahlfreien Zugriff auf irgendein Zeichen eines Textes, weil jedes Zeichen mit derselben Anzahl von Bytes kodiert wird.

Leider ist auch das nicht ganz richtig, denn es gibt eine große Zahl diakritischer Zeichen, die erst in Kombination mit einem Basiszeichen ein neues Zeichen ergeben.

Wenn daher diakritische Zeichen verwendet werden, gibt es nicht einmal bei einer UTF-32-Kodierung wahlfreien Zugriff. Wenn daher in einem Text zum Beispiel 10 diakritische Zeichen verwendet werden, dann ist die Gesamtzahl der dargestellten Zeichen um 10 kleiner als es der Datenlänge entspricht.

## UTF-8

Da alle Internet-Protokolle auf dem 7-Bit-ASCII-Zeichensatz aufbauen, wäre es eine Bandbreitenverschwendung, würde man diese Protokolle mit einem platzhungrigen Kode wie dem UTF-16 oder gar UTF-32 abwickeln.

Andererseits reicht zwar ASCII allein für die Protokoll-Elemente aus, nicht aber für die Inhalte. Gesucht ist daher ein Kode, der sich für die Protokollelemente wie ein platzsparender 8-Bit-Kode verhält und für die Inhalte für die häufig benutzten Zeichen mit möglichst wenigen Bytes auskommt und nur für die sehr seltenen Zeichen auch einmal mehr Bytes verbraucht. Ein solcher Kode ist UTF-8.

Bits	CodePoint	Kodierung
7	0...127	0xxxxxxx
11	80...7FF	110xxxxx 10xxxxxx
16	800...FFFF	1110xxxx 10xxxxxx 10xxxxxx
21	10000...10FFFF	11110xxx 10xxxxxx 10xxxxxx 10xxxxxx

## UTF-7

Der UTF-7-Kode war dafür gedacht, den Unicode-Zeichenraum über 7-Bit-Kanäle zu übertragen. Diese Kodierung hat sich nicht durchgesetzt.

## Speicherung der Kodes

Bei allen Kodierungen, die mehr als nur ein Byte pro CodePoint benötigen (UTF-16, UTF-32), kann die Speicherung mit dem niederwertigen Byte beginnen (*Little Endian*) oder mit dem höherwertigen Byte (*Big Endian*) beginnen. Wie der Rechner das handhabt, hängt von seiner inneren Verarbeitung ab. Auf Windows-Rechnern wird die Variante *Little Endian* verwendet.

Um beide Möglichkeiten anbieten zu können, gibt es die Kodes UTF-16 und UTF-32 in beiden Varianten mit jeweils anderer Codepage-Nummer.

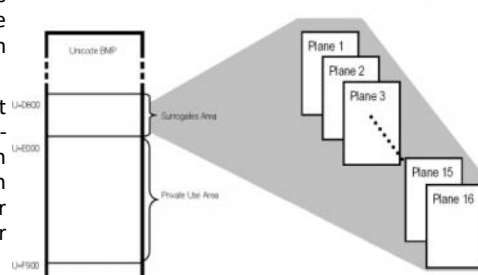
CodePage	Kodierung	Zeichen	Kode
65000	UTF-7	A	41
65001	UTF-8	A	41
1200	UTF-16	A	41 00
1201	UTF-16-BE	A	00 41
12000	UTF-32	A	41 00 00 00
12001	UTF-32-BE	A	00 00 00 41

## Wie speichert die Zeichen ein Rechner?

Windows benutzt intern zur Zeichenablage UTF-16-LE, Unix verwendet UTF-8.

Hier scheint Windows im Vorteil zu sein, denn der Zugriff zu einem Zeichen scheint rascher

Alle Zeichen der höherwertigen Planes werden bei UTF-16-Kodierung mit zwei Bytes in dem dafür reservierten Kodebereich D800-DCFF, der so genannten *Surrogates* projiziert.







## Reiner Text

Eine Datei, die nur Textzeichen enthält, ist „reiner“ Text. So „rein“ ist aber der Text gar nicht, denn es gibt einige Möglichkeiten, „reinen“ Text zu speichern.

### Reiner Text im Notepad

Notepad ist ein Programm, das solche einfachen, unformatierten Texte verarbeitet. Im Bild rechts sind einige Zeichen eingegeben worden. Speichert man einen solchen Text, gibt es vier Speicheroptionen.

- ANSI
- Unicode
- Unicode Big Endian
- UTF-8

Wählt man die Option „ANSI“ und als Namen „demo-ansi.txt“, enthält das Dokument Zeichen, die in dieser Speicheroption nicht darstellbar sind, bekommt man eine Fehlermeldung (Bild rechts oben) zu sehen, die darauf hinweist. Am Bildschirm bleiben dabei die Zeichen unverändert. Würde man aber die gespeicherte Datei neu in den Editor laden, fehlen diese Zeichen.

Man sieht, nicht alle Zeichen, die im Editor darstellbar sind, können in diesem „ANSI“-Format auch gespeichert werden, denn die ANSI-Kodierung kann nur 256 Zeichen darstellen.

Im Beispieltext wäre das „Große scharfe s“ betroffen.

In den anderen Optionen (Unicode und UTF-8) gibt es dieses Problem nicht, diese verfügen über einen praktisch unbeschränkten Zeichenraum.

### Reiner Text in Word

Würde man diesen „reinen“ ANSI-Text in Word laden, hat Word die Schwierigkeit, diesem Text eine Kodierung zuzuweisen. Da man das nicht so ohne Weiteres erraten kann, fragt Word in einem eigenen Dialog nach, wie die Kodierung des Textes einzustellen ist und zeigt dabei die ersten Zeilen an und schlägt eine Kodierung vor.

Im ersten Beispiel wurde dem Text die Kodierung „MS-DOS“ zugewiesen und statt der Umlaut-Zeichen sieht man jetzt Block-Grafik-Zeichen.

Wählt man „Windows-Standard“, schaut das Ergebnis besser aus. Mit „Andere Kodierung“ könnte man auch noch entlegene Kodierungen wählen.

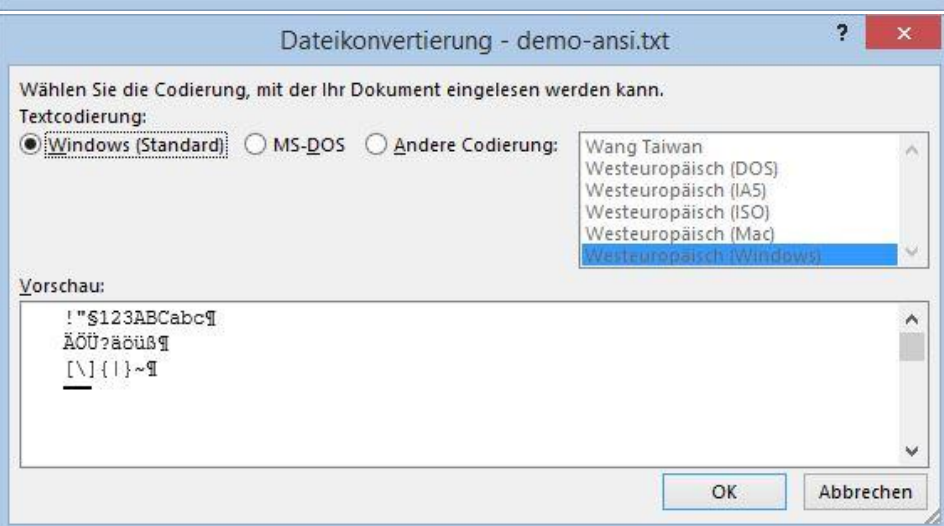
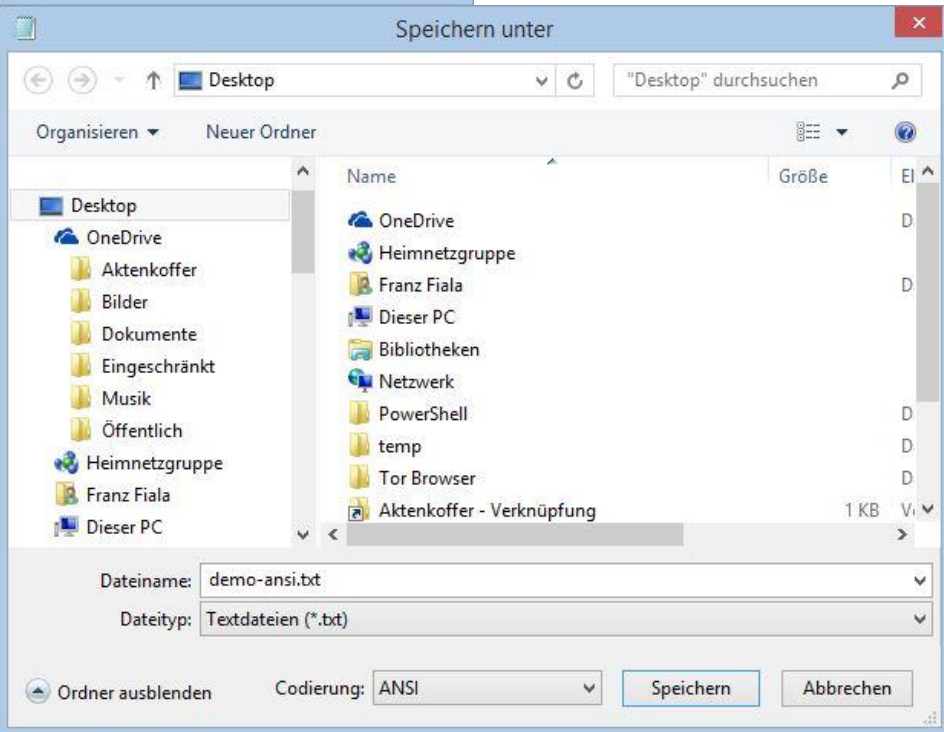
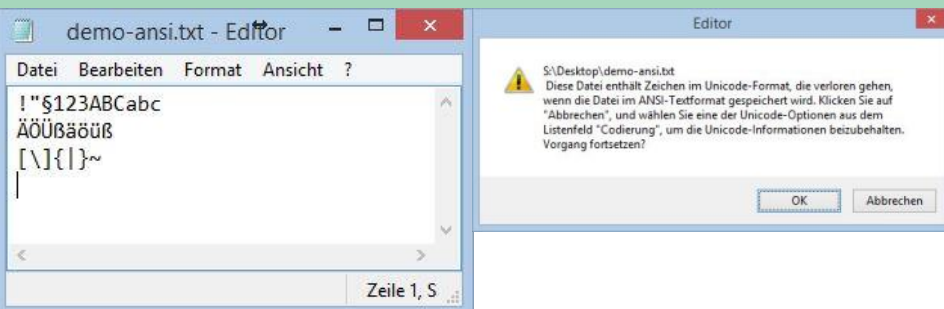
Diesen Dialog sehen wir daher immer dann, wenn Word die Kodierung eines „reinen Textes“ nicht bestimmen kann.

Bei allen alten „reinen Texten“ ist nicht an jeder Kodeposition dasselbe Zeichen, wobei mit „alt“ die Zeit vor 1995 gemeint ist. Heutzutage wird dieses Zuordnungsproblem hauptsächlich beim Umgang mit alten Texten auftreten. Die Kodierung, die man zuweisen muss, wird auch Codepage genannt.

Bei neueren „reinen“ Textdokumenten, die mit dem Notepad generiert worden sind, besteht dieses Verständnisproblem nicht, sofern sie mit den Optionen Unicode, Unicode BE oder UTF-8 gespeichert worden sind.

Alle diese Texte haben auch die Endung .txt aber die Editoren können an deren besonderen Aufbau die Kodierung eindeutig erkennen.

Wie, das sieht man im Kasten auf der folgenden Seite.



## Welche Kodierung hat eine Textdatei?

Wie kann man nun unterscheiden, welche Kodierung in einer Textdatei gespeichert ist? Interpretieren kann man die gespeicherten Zeichen offenbar auf viele verschiedene Arten. Als 8-Bit-Datei ist überhaupt vieles unklar, denn man muss immer dazu sagen, um welche Codepage es sich handelt. Im Zweifel muss man mit den Codepagenummern experimentieren, bis man die richtige gefunden hat.

Um die Anzahl der Möglichkeiten einzuschränken gibt es bei den Codepages für Unicode-Daten die optionale Möglichkeit, den Text mit einer besonderen Byte-Kombination einzuleiten, damit ein lesendes Programm die Kodierung automatisch einstellen kann. Diese Bytekombination heißt Byte Order Mark (BOM). Es gilt:

```
2b 2f 76 38 Unicode UTF-7
2b 2f 76 39 Unicode UTF-7
2b 2f 76 2b Unicode UTF-7
2b 2f 76 2f Unicode UTF-7
ef bb bf Unicode UTF-8
ff fe Unicode UTF-16-LE (Windows)
fe ff Unicode UTF-16-BE
ff fe 00 00 Unicode UTF-32
00 00 fe ff Unicode UTF-32-BE
```

[http://de.wikipedia.org/wiki/Byte_Order_Mark](http://de.wikipedia.org/wiki/Byte_Order_Mark)

Diese BOMs sind nicht zwingend vorgeschrieben, werden aber häufig verwendet.

Es kommt vor, dass man bei Ausgabe einer Textdatei in einem Cmd-Fenster (DOS-Box), diese Byte-Order-Marks sieht, denn die Konsole in einem Cmd-Fenster interpretiert diese Zeichen nicht als Unicode und gibt sie byteweise aus.

Wenn daher aktuelle Textdokumente mit diesem BOM gespeichert sind, besteht Eindeutigkeit über die verwendete Kodierung. Stammt das Textdokument dagegen aus den Anfangszeiten von DOS oder Windows, muss man die zutreffende Codepage suchen, bevor man den Text in die Unicode-Welt konvertiert.

In Word äußert sich das so, dass bei Dateien ohne BOM zuerst einmal eine Fehlermeldung kommt und man sich danach für eine Codepage entscheiden kann. Öffnet man in Word eine Textdatei mit BOM, gibt es den Fehler nicht aber man wird zur Sicherheit dennoch auf die Codepage-Auswahl umgeleitet und die festgestellte Kodierung wird voreingestellt.

## Nachteile des BOM

Es gibt wenigstens drei Probleme beim Umgang mit dem BOM (Byte Order Mark):

- Dateien, die keinen Text enthalten, sind nicht mehr leer, weil sie ja eine mehr-bytige Kennzeichnung haben.
- Dateien, die ASCII-Texte enthalten sind keine ASCII-Dateien mehr, weil das BOM voreingestellt ist. Daher kann es bei Programmen, die reines ASCII als Input erwarten, zu Schwierigkeiten kommen.
- Man kann Textdateien nicht mehr verketteten, weil an der Nahtstelle das BOM der nächsten Datei eingefügt wird und daher die resultierende Datei fehlerhaft ist. Das Betriebssystem hat diese Neuerung der Kennzeichnung von Text-Dateien nicht mitgemacht. Wünschenswert wäre daher eine Option beim Kopierbefehl, der die Berücksichtigung des BOM erzwingt.

## Die vier Speichermethoden des Notepad

Wir geben im Notepad folgenden Mustertext, ein vereinfachtes Alphabet, ein.

```
!"$123ABCabc
ÄÖÜßäöüß
[\]{}|~
```

Der Text enthält in drei Zeilen jeweils drei Zeichen aus einem Zeichenbereich. Es sind insgesamt 33 Zeichen. Das „große scharfe s“ kann mit Alt-Gr Shift ß eingegeben werden. Wenn das nicht funktioniert, die Windows-Zeichentabelle verwenden und das Zeichen 1e9e im Unicode-Feld eingeben und auf „Auswählen“ klicken.

Es gibt vier Möglichkeiten zu speichern: ANSI, Unicode, Unicode-BE und UTF-8.

### ANSI (=Windows Codepage 1252 „Westeuropäisch“)

```
21 22 a7 31 32 33 41 42 43 61 62 63 0d 0a c4 d6 !" $123ABCabc . ÄÖ
dc 3f e4 f6 fc df 0d 0a 5b 5c 5d 7b 7c 7d 7e 0d Ü?äöüß. . [\]{}|~.
0a
```

Die Datei hat exakt gleich viele Bytes wie der ursprüngliche Text Zeichen hat. Klar, es handelt sich um eine 8-Bit-Kodierung. Jedes Zeichen wird durch ein Byte dargestellt. Was fehlt, ist das große Scharfe-S, weil dieses Zeichen nicht im Zeichenraum 0..255 enthalten ist. Daher steht an dieser Stelle ein Fragezeichen (3f).

Man sieht es dieser Datei nicht an, welche Codepage zur Anwendung kommt. Würde man daher diese Seite in einer DOS-Box anzeigen, sieht man, wegen der dort auf 850 geänderten Codepage folgendes:

```
!"°123ABCabc
-İ ?ö÷³■
[\]{}|~
```

Man sieht, dass man bei den 8-Bit-Codepages unbedingt wissen muss, welche Kodierung verwendet wurde, sonst gibt es Datensalat.

Um diese Mehrdeutigkeiten auszuschalten, wird bei den Kodierungen für den UniCode jeder Textdatei optional ein *Byte Order Mark* vorangestellt.

### UTF-16 (UCS-2) Unicode LE

```
ff fe 21 00 22 00 a7 00 31 00 32 00 33 00 41 00 !" $123A
42 00 43 00 61 00 62 00 63 00 0d 00 0a 00 c4 00 BCabc. . Ä
d6 00 dc 00 9e 1e e4 00 f6 00 fc 00 df 00 0d 00 ÖÜßäöüß.
0a 00 5b 00 5c 00 5d 00 7b 00 7c 00 7d 00 7e 00 . [\]{}|~
0d 00 0a 00 ..
```

Die UTF-16-Kodierung verwendet zwei Bytes pro Zeichen und speichert (in der in Windows verwendeten Variante „*Little Endian*“) das niederwertige Byte auf die niederwertige Adresse. Daher sollte die Datei 66 Zeichen lang sein. Sie ist aber 68 Zeichen lang. Den Grund sieht man in den beiden ersten Bytes „ff fe“, die eine Kennzeichnung für UTF-16 darstellen. Diese Kennzeichnung hat den Namen *Byte Order Mark* (BOM). Sie ist nicht verpflichtend, hilft aber beim Einlesen von Texten, die richtige Kodierung zu finden.

### UTF-16 (UCS-2) Unicode BE

```
fe ff 00 21 00 22 00 a7 00 31 00 32 00 33 00 41 !" $123A
00 42 00 43 00 61 00 62 00 63 00 0d 00 0a 00 c4 BCabc. . Ä
00 d6 00 dc 1e 9e 00 e4 00 f6 00 fc 00 df 00 0d ÖÜßäöüß.
00 0a 00 5b 00 5c 00 5d 00 7b 00 7c 00 7d 00 7e . [\]{}|~
00 0d 00 0a ..
```

Die Variante „*Big Endian*“ dreht die Speicherreihenfolge der Bytes in einem Zeichen und auch des BOM um, verhält sich ansonsten aber wie die Windows-Variante.

### UTF-8

```
ef bb bf 21 22 c2 a7 31 32 33 41 42 43 61 62 63 !" $123ABC
0d 0a c3 84 c3 96 c3 9c e1 ba 9e c3 a4 c3 b6 c3 . . ÄÖÜßäöü
bc c3 9f 0d 0a 5b 5c 5d 7b 7c 7d 7e 0d 0a B. . [\]{}|~. .
```

Wer viel englischen oder deutschen Text bearbeitet, verliert in der UTF-16-Kodierung viel an Speicherplatz, weil die meisten Zeichen aus dem Zeichenraum der ersten 256 Zeichen stammen und die 16-Bit-Kodierung für alle diese Zeichen zusätzlich „00“ speichert. Die UTF-8-Kodierung spart viel von diesem Platz, allerdings zu dem Preis, dass die Zeichen verschieden viele Bytes beanspruchen. Unser Beispiel mit 33 Zeichen beansprucht in UTF-8 46 Bytes. Die Speicherreihenfolge ist immer dieselbe. Das BOM besteht aus den drei Bytes: ef bb bf.

**Anmerkung:** Bei einer Notepad-Datei kann man es sich aussuchen, wie der Text gespeichert werden soll. Man kann wählen: ANSI, Unicode, Unicode Big Endian und UTF-8.

Eigentlich ist aber nur die letzte Bezeichnung „UTF-8“ korrekt, denn „Unicode“ ist keine Kodierung. Unicode ist ein Alphabet. Die Kodierung sagt, wie die Buchstaben zu Bytes werden. Gemeint ist im Notepad mit „Unicode“ die Kodierung UTF-16, bzw. UTF-16-BE. Auch „ANSI“ ist sonderbar, denn es gibt keine Kodierung dieses Namens. Um herauszufinden, was damit gemeint ist, speichert man die Bytes 30H bis 255H und vergleicht mit den Zeichen der verschiedenen Codepages. Demnach ist „ANSI“ die Windows Codepage 1252 „Westeuropäisch“.



# Ubuntu

Promiskuitätsrate explodiert!

Günter Hartl

Okay, ich musste es schreiben. Falls Du es nicht gemerkt hast, mit diesen 2 Wörtern habe ich dich marketingmäßig extrem ausgefinkelt an den Artikel gefesselt. Im Fachjargon auch unter dem Terminus „Bondageartikel“ bekannt. Ich verspreche Dir, dass Du nicht enttäuscht werden wirst. Falls doch, könnten wir eine ergebnisoffene Diskussion anstoßen.

April 2014 war ein „wichtiges“ Datum für die meisten Linuxjünger. In diesem Monat kam die neueste Ubuntu-Version mit Langzeitsupport raus.

„Ubuntu 14.04“ heißt demnach die aktuelle Version. Die letzte war „Ubuntu 12.04“.

Wie unschwer zu erraten ist, beziehen sich die Ziffernfolgen auf die Jahres- und Monatszahl der Ubuntu-Distribution. fünf Jahre dauert der Supportzeitraum. Also noch bis 2017, respektive 2019 bei 14.04.

Die Charakteristik von LTS (*Long Term Support*) Versionen ist eher auf Stabilität, denn auf die Implementation neuer Features ausgelegt. Also für Leute, die die vorige Ubuntu-Version (13.10) installiert haben, wird sich nichts Dramatisches geändert haben.

Merke: Alle sechs Monate kommt eine „neue“ Ubuntu-Version raus. Man spricht auch öfters von der „Ubuntu-Karawane“, die alle sechs Monate „weiterzieht“. Sprich, alle sechs Monate werden neue Features und Kernel eingepflegt.

Wer allerdings „Ruhe“ haben will, der greift einfach zur LTS-Version. Läuft stressfrei. Das kann mitunter auch bei den „Zwischenversionen“ (12.10, 13.04, 13.10....) so sein, muss aber nicht. Diese sind eher für Leute gedacht, die „immer die neuesten Features“ und die augenkrebszerzeugenden Effekte haben wollen.

Für den Produktiveinsatz empfiehlt sich aber immer die LTS-Version. Wie sieht das Glump aus? So: Bild 1.

Nix aufregendes. Rennt. Die Oberfläche (Desktopumgebung) von Ubuntu heißt seit einigen Jahren *Unity*. Wie üblich, spaltet auch diese die Gemüter. Aus dem einfachen Grund: Jeder Mensch ist anders. Und das ist auch gut so. Die Softwareentwickler konzipieren immer neue Sachen mit dem Menschen im Mittelpunkt. Dort steht er aber jedem nur im Weg herum :-).

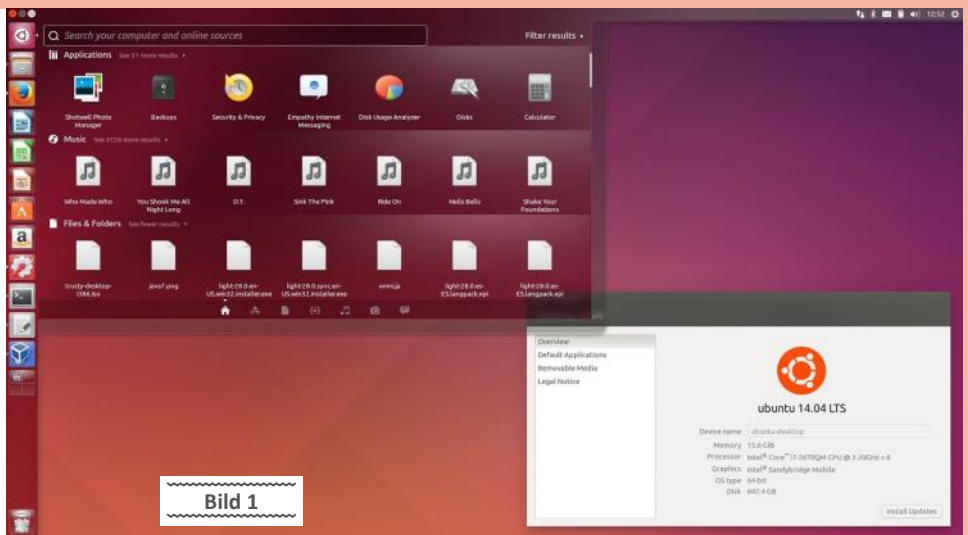
Das größte Plus (und gleichzeitig die größte Schwäche) von Linux sind die schier unglaublichen Auswahlmöglichkeiten. Ein Anfänger wird förmlich erschlagen von diesen. Bei Windows gibt's sowas nicht. Da hast Du eine Oberfläche und das war's.

Schau mal her:

Ubuntu • Kubuntu • Xubuntu • Lubuntu • Ubuntu-Gnome • Ubuntu-Mate

Das sind alles Betriebssysteme. Mit Ubuntu „unter der Haube“. Das einzige, was diese sechs Distributionen unterscheidet, ist deren Desktopumgebung. Also das, was der User am Bildschirm sieht. Man kann auch alle sechs Desktopumgebungen auf einer Maschine installieren und nach Belieben wechseln.

In der Regel entscheidet man sich aber im Vorfeld für eine „Oberfläche“. Diese Entscheidung



ist für Mac- und Windows-User natürlich Neuland.

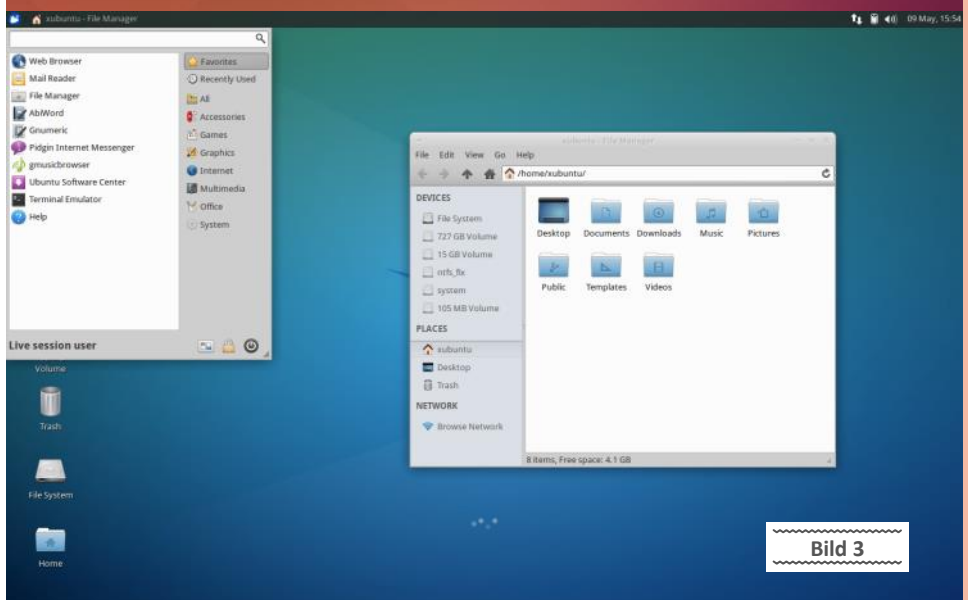
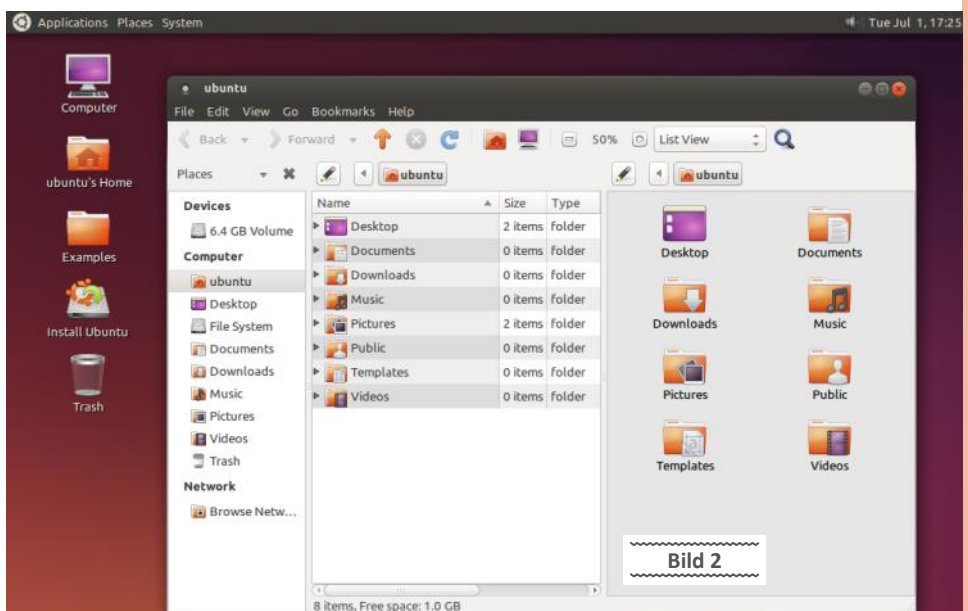
Ich persönlich bevorzuge die „Kubuntu“-Variante. Einfach deshalb, weil hier schier unglaublich viele Einstellmöglichkeiten am System vorhanden sind.

Noch mal, das ist meine persönliche Wahl. Bei meinem Vater habe ich „Ubuntu-Mate“ installiert. Einfach deshalb, weil auf seinem 11 Jahre alten Rechner mit 1 GB Ram diese Oberfläche

sehr flüssig läuft. Außerdem ist sie extrem einfach gehalten und selbsterklärend. Siehe Bild 2.

Einer Trainerkollegin habe ich das normale „Unity“ installiert und sie liebt es im Gegensatz zu mir. Such Dir einfach die Oberfläche aus, die Dir am besten zusagt.

Wie? Ganz einfach, starte das entsprechende Ubuntu im „live-Modus“. So kannst Du gleich ausprobieren, was Dir liegt, ohne am Grundsys-



tem (deinem Laptop oder PC) irgendwas zu verändern.

Ich erspar' mir hier eine Screenshot-Tour, wie man ein Live-Medium herstellt. Google ist dein Freund.

Kleiner Tipp: Poweruser nehmen gerne Xubuntu. Dessen Oberfläche (xfce) ist extrem schnell, sehr gut konfigurierbar und außerordentlich funktionell. Siehe Bild 3.

Hier noch mein persönlicher Senf zu den verschiedenen Distributionen und deren Empfehlungen:

**Ubuntu:** Für Enduser, die sowohl mit Maus als auch mit der Tastatur arbeiten wollen. Gibt sehr praktische Shortcuts (Windows-Metastaste) und eingebaute Suchfunktionen. Moderner Look. Leiste fix auf der linken Seite (kommt den heutigen Bildschirmformfaktoren sehr entgegen). Diese erinnern mich immer an meine Hausmeisterin. Schön breit und gestaucht). Leicht zu konfigurieren.

**Kubuntu:** Vom Aufbau kann man eine gewisse Windowsähnlichkeit nicht verleugnen. Panel unten, Startmenü links unten. Hat sich seit den 90ern nicht viel geändert am Design. Siehe Bild 4.

Sehr granular konfigurierbar in jeder Hinsicht. Alleine bei den Einstellungen im Dateimanager (Dolphin...in meinen Augen der beste Filemanager, den es gibt) kannst Du dich stundenlang „spielen“. Im Gegensatz dazu ist der in Ubuntu (Nautilus) fast als spartanisch zu bezeichnen.

Nicht falsch verstehen, Kubuntu ist nach der Installation sofort einsatzbereit. Aber jeder will's halt ein bisschen anders haben. Sehr komfortable Oberfläche (Datumsanzeige mit Feiertagen je nach Land, Android-Connector, Zweifensteransicht, bis zu 20 Arbeitsflächen, übersichtlicher Netzwerkmanager ..)

**Lubuntu:** Ist noch einen Tick ressourcenschonender als Xubuntu. Eignet sich auch hervorragend für Netbooks. Einfacher Aufbau. Ist Geschmackssache, ob man Xubuntu oder Lubuntu wählt. Wie gesagt, nur meine Meinung. Siehe Bild 5.

**Ubuntu-Mate:** Auch so ein typischer „einfacher“ Desktop. Oma-tauglich. Uneingeschränkt. Sehr schnell und von schlichter Eleganz. Siehe Bild 2.

**Ubuntu-Gnome:** Läuft mittlerweile auch sehr performant. Siehe Bild 6. Hat ein modernes Design, das sehr gut mit der Tastatur ansteuerbar ist. Ich persönlich habe es noch nie produktiv verwendet.

Warum? Keine Ahnung. Ich muss auch zu meiner Schande gestehen, dass ich noch nie Kapitän auf einem atombetriebenen Flugzeugträger war. Weißt Du, auf was ich hinaus will?

Du hast unter Linux immer die Wahl. Wenn Du „Deine Oberfläche“ gefunden hast, bleib dabei. Du musst nicht alle durchprobieren, kannst es aber machen. Viel Spaß...es gibt ungefähr 20 verschiedene Desktopoberflächen. Wobei in meinen Augen die genannten sechs für den Heim-User am weitesten verbreitet sind.

Ich kenne aber auch Leute, die arbeiten am liebsten ohne Oberfläche und nur im Terminal. Speziell bei Servern sehr vorteilhaft. Auch gut. Du hast die Wahl. Das ist der große Unterschied zu Mac und Windows.

Die Mac-Oberfläche mit seinen Features ist seit gut 15 Jahren unverändert. Zumindest ist keine Anlernzeit bei Versionsupdates einzuplanen. Und bei Windows? Alleine die 8er Version (8.0, 8.1.1, 9er wieder mit Menü) trägt nicht wirklich zu einer „geraden“ Designlinie bei.

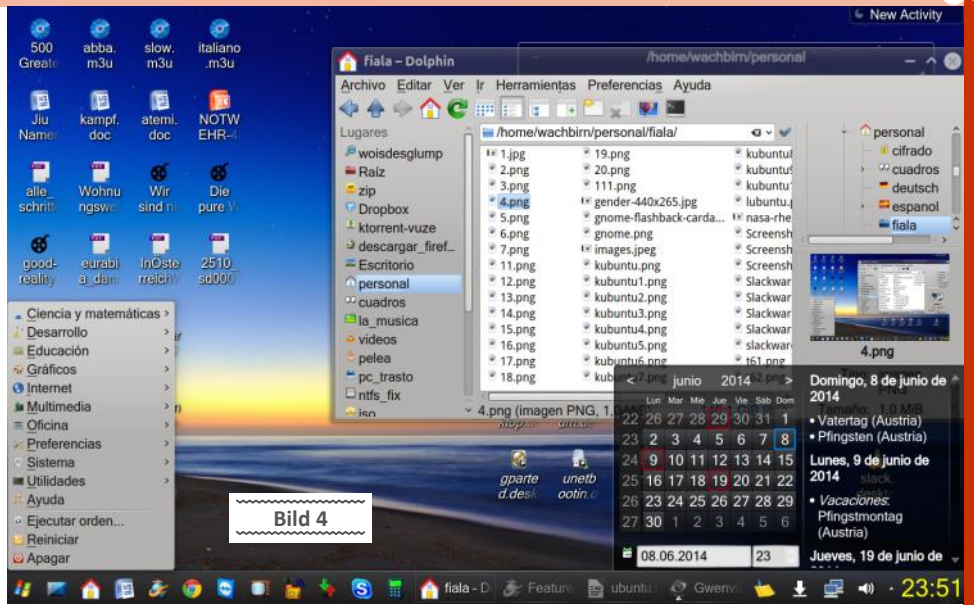


Bild 4

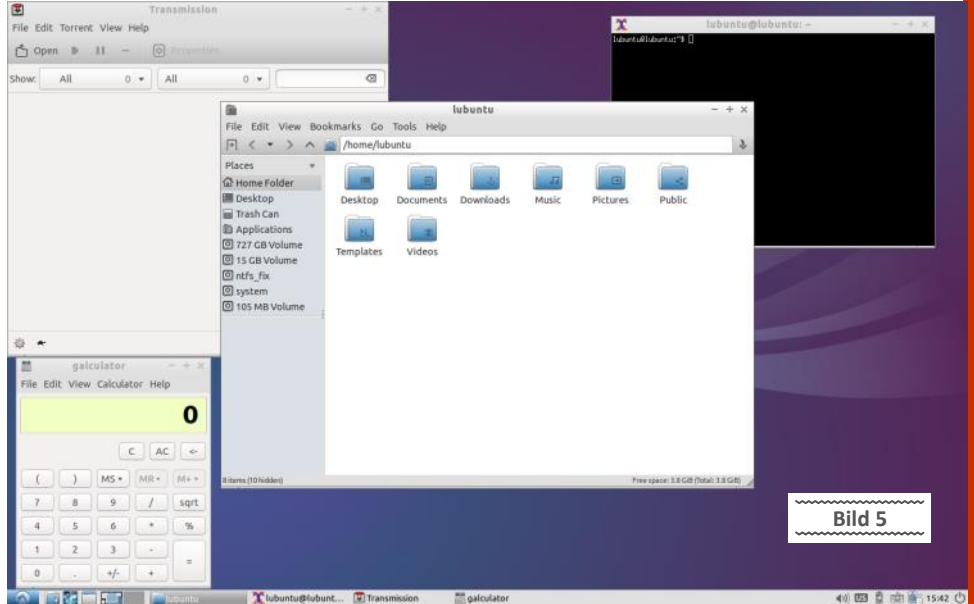


Bild 5

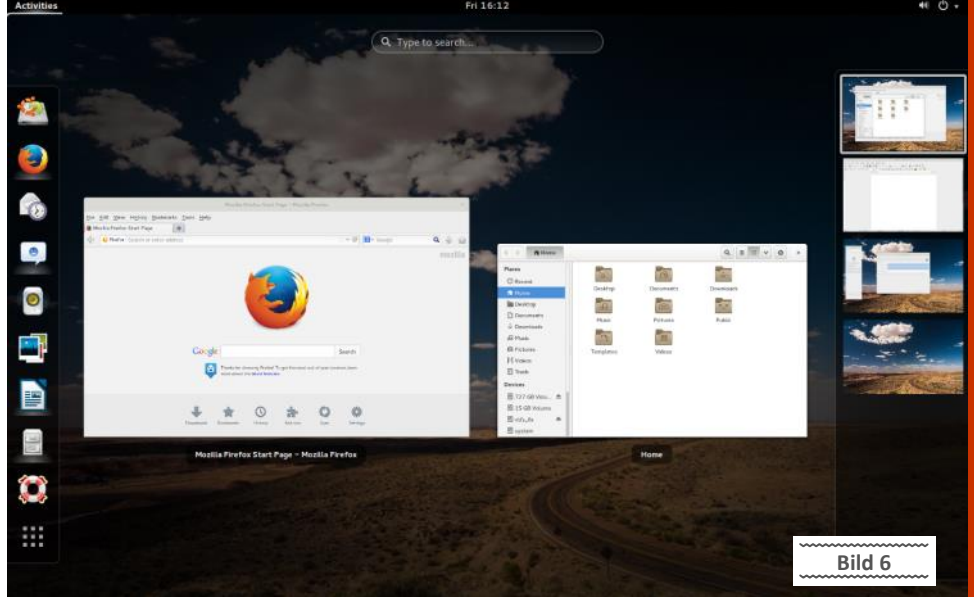


Bild 6

Mein Bruder arbeitet in einem internationalen Konzern. Jedes Mal kann ich mir sein Geflüche anhören, weil sie kürzlich von Lotus Notes auf Office 365 umgestiegen sind. Da klar kennt man sich anfangs nicht aus.

Der Unterschied zum Heimuserbereich ist nur ein einziger: In der Firma hast Du keine Wahlmöglichkeit. Irgendwann wirst Du Dich schon dran gewöhnen. Bleibt Dir e nix anderes über.

Wenn'st immer vorne mit dabei sein willst, installierst alle sechs Monate ein neues Ubuntu. Wenn'st Ruhe haben willst, installierst eine LTS-Version und Du hast fünf Jahre keinen Stress; Deine Wahl.

Nicht vergessen, nicht alle haben stundenlang Zeit, sich erstmal mit dem System auseinanderzusetzen. Viele wollen wirklich auch arbeiten am PC. Für einen Pensionisten kann ein Be-



triebssystem eine nette Spielerei sein. Warum nicht. Jeder, wie er will.

Wie schon eingangs erwähnt, verwende ich die KDE-Oberfläche. Und das schon seit Jahren. Das ausschlagende Kriterium für mich war und ist immer noch der Dateimanager. „Dolphin“ heißt der unter KDE. Einen Webbrowser bedient man auf einem Smartphone oder Desktopsystem ziemlich gleich. Aber beim Verwalten von Dateien oder Ressourcen gibt es schon erhebliche Unterschiede.

Es mag vielleicht altmodisch klingen. Aber eine Zwei-Fensteransicht ist für mich heutzutage ein Musskriterium. Das hat's doch schon 1995 mal gegeben unter Windows, wenn ich mich nicht irre. Dagegen ist jedes „snap-in“ eine Krücke.

Schau Dir mal meinen Dateimanager an. Bild 7. Ja, ich weiß. Schön bunt.

Links die Lesezeichen mit Schnelzugriff auf meine Ressourcen. Rechts außen der Dateibaum zum Navigieren (den verwende ich aber selten bis gar nicht).

Links unten werden angeschlossene Geräte angezeigt >dispositivos>16gb. Das ist mein angesteckter USB-Stick. Verschiedene Ansichtsmodi mit einem Mausklick. Die meisten Ordner haben ein eigenes Symbol. Ich bin eben ein visueller Typ.

Darum verstehe ich auch die Windows-Nutzer nicht, die kilometerlange Ordnerlisten im Explorer haben und dann herumnavigieren. Alles gelb in einer Wurst. Einfach ein auffälliges Symbol zuordnen, und es wird gleich ein wenig übersichtlicher.

Aufpassen muss man unter Windows nur, dass man den Ordner Direkt aufruft und nicht über die Bibliotheken. Unter diesen gibt es den Reiter „Anpassen“ nämlich nicht.

Also Direkt über den Pfad den Ordner in Windows ansteuern und dann ein Symbol zuordnen. Funktioniert unter Win7 und 8 so. Siehe Bild 8.

Im rechten Fenster siehst Du den entsprechenden Reiter. Links bin ich den Weg über die Bibliotheken gegangen. Nicht gut.

## Migration zu Linux

Weil mich ein paar Leute angesprochen haben bezüglich meines letzten Artikels mit der Linux-Migration in München.

Waren ca. 15000 Maschinen. Bitte nicht vergessen, das Projekt dauerte gut 10 Jahre. Also da kommen nicht ein paar Leute übers Wochenende und machen sich da im Serverraum wichtig und am Montag rennt das neue System.

Jede Migration kostet Zeit, Nerven und Geld. Know-How schadet auch nicht. Egal, welche Plattform man gerade „fährt“. Jede Migration (Windows > Linux, Linux > Windows oder Windows > Mac) ist ein größeres Projekt, das sich in der Regel über mehrere Monate (Jahre) zieht.

Darum ist es auch so immens wichtig, sich für die „richtige“ Plattform zu entscheiden. Erspart einem zu einem späteren Zeitpunkt eine Menge Kopfweh.

Sicher gibt es auch genug Beispiele, wo Migrationen in „die Hose“ gegangen sind. Meist überall, wo im Vorfeld nicht gut gearbeitet wurde. Kenne einige Fälle, wo Windows > Linux Migrationen gescheitert sind (Ditech war so ein Kandidat... jetzt keine falschen Schlüsse ziehen :-). Natürlich gibt es auch positive Beispiele.

Ja, auch die NASA ist schon vor einiger Zeit zu Linux gewechselt. Zumindest im Kontrollzentrum. Red Hat verwenden die dort. Bild 8a.

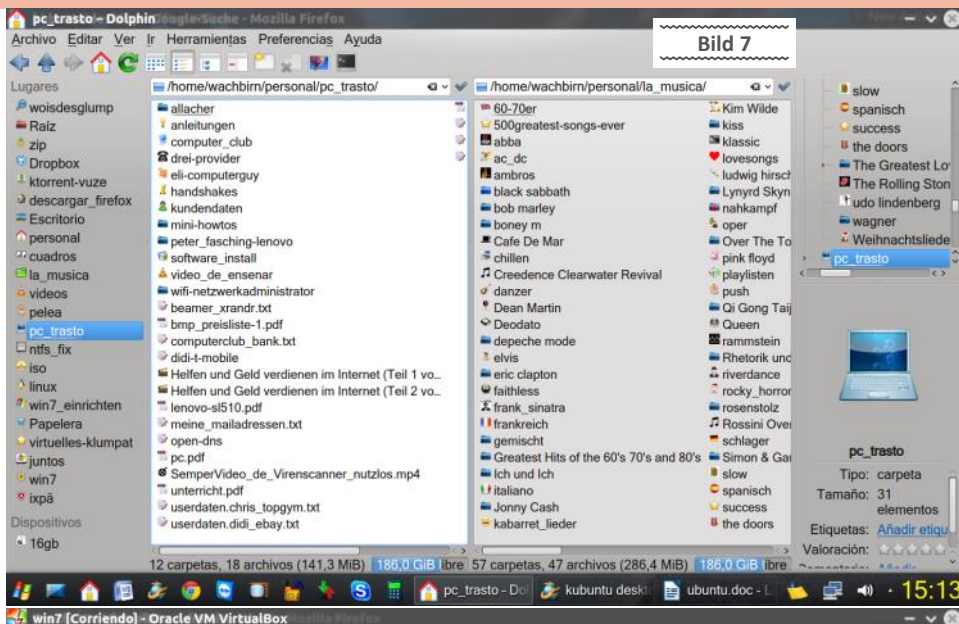


Bild 7

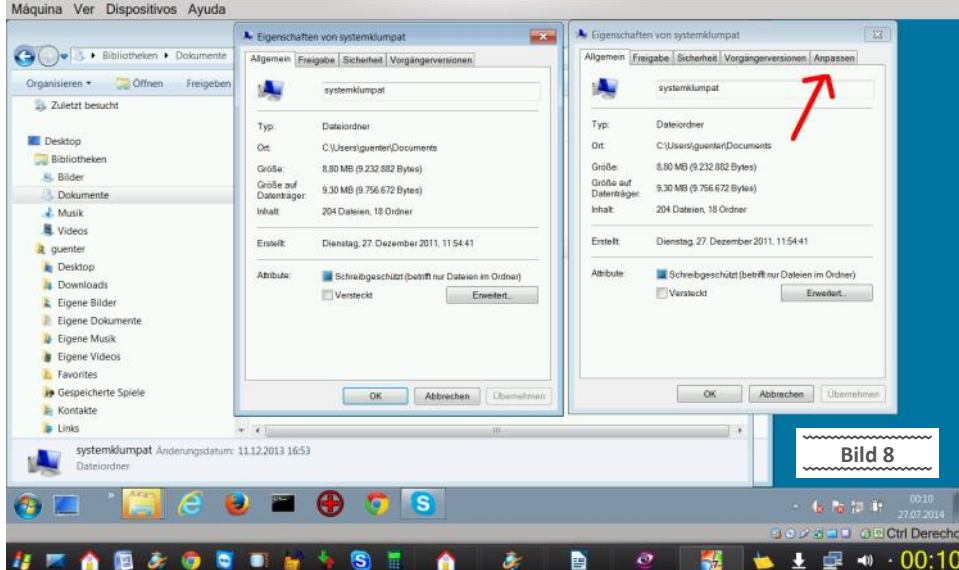


Bild 8



Bild 8a

Da lacht das Herz. Eine rustikale Tastatur mit einer kabelgebundenen Maus. Zwei erdige Monitore und ein Wandtelefon mit Direktverbindung zur Werkskantine. Nicht so ein Schnick-Schnack wie heute in Designerbüros.

Die wichtigere Frage lautet aber? Warum „woll(t)en“ die zu Linux „wechseln“? Ganz einfach, die meisten wollen gar nicht zu Linux sondern

einfach weg von Microsoft. Linux ist nur die bekannteste Alternative.

Microsoft ist eine wunderbare Plattform. Keine Frage. Das einzige damit verbundene Problem liegt in deren Unverträglichkeit mit anderen Systemen. Je mehr MS-Zeug im Netzwerk, desto inkompatibler zu anderen Systemen wird's.



Und wenn Du glaubst, dass es in die andere Richtung „schneller“ geht (Linux > Windows), leg das Heft beiseite und widme Dich wieder deinem Solitäre-Spiel.

Ein heterogenes Netzwerk ist wie ein Sack voll Flöhe zu behandeln. An allen Ecken und Enden gibt's ständig was zu tun :-).

Fazit: Es gibt immer Gründe, warum ein Unternehmen (oder Firma) ihrer Plattform treu bleibt oder eben nicht. Wie's einmal so schön in einer Mac-Werbung verlautbart wurde: *"Fear of switching is the foundation of customer loyalty to PCs [as in Windows]"*. Auf Deutsch, sinngemäß: *Die Angst vor dem Wechsel ist das Fundament der Treue zu (Windows)PCs*.

Das gilt natürlich auch für den Heimuserbereich.

*A bisserl hatschert, aber es kommt hin.* Wer will schon der „erste“ sein, wo er schon so viel Manpower und Geld in den Aufbau der Infrastruktur reingebuttert hat. Mit einer Migration das Ganze noch mal? Puh....

Schulungskosten noch dazu. Gut, die hast' bei Windows sowieso auch. Speziell wenn'st noch vom Server 2003 kommst und dann den 2012 implementieren willst und nach einer Internet-recherche draufkommst, dass der gar keine Bandlaufwerke (speziell zu Sicherungszwecken) mehr unterstützt. Das war jetzt a bisserl unfair. Der Server 2012 kann eine Menge. Kein Spaß.

Und nochmal. So aus dem Stand kannst keinen MS-Server mehr (richtig) konfigurieren. Unmöglich. Auch hier ist Know-How der Schlüssel zum Erfolg.

## Linux im Heimuserbereich

In letzter Zeit hatte ich das Vergnügen, einen 5 Jahre alten Laptop für eine Bekannte „herzurichten“.

Ein „*Asus aspire one*“ mit 1 GB Ram. „*i was nüt, er is so langsam und des gubl gät so zach auf...*“. Ehrlich gesagt, ich habe schon Pornos gesehen, die bessere Dialoge hatten.

Windows7 Starter war da drauf und rannte wie die Sau. Natürlich nicht. Abgesehen von dem Schrott, der da mit der Zeit auch draufkam, war es eine recht zähe Geschichte.

Ich richtete Dualboot ein. Einmal Windows7-Pro und einmal Linux-Slackware. Zu meinem Erstaunen rannte Windows7 recht flüssig auf dem Gerät. Die Frage lautet aber immer: wie lange?

Das ist das Problem.

Slackware Linux auf der anderen Seite kostete mich schon ein paar Stunden zum Einrichten. Als Oberfläche nahm ich „xfce“. Sehr leichtgewichtig und funktionell. Das Linux-System läuft auch recht geschmeidig drauf.

Der Unterschied von Slackware zu den bekannteren Ubuntu-Varianten liegt in deutlich reduzierten Overhead. Sämtliche Einstellungen kannst (musst) Du in Textfiles vornehmen. Es gibt da keine mausgesteuerten Anzeigedialoge.

Übrigens: diese mausgesteuerten Anzeigedialoge machen auch nix anderes, als dass sie Werte in die entsprechenden Textfiles reinschreiben. Bei Windows landet der ganze Krempel letzters irgendwann in der Registry. Bei Linux immer in Textfiles, die man natürlich editieren kann.

Darum lernst Du in Linux als erstes (oder zumindest immer anfangs) wie man einen Texteditor bedient. Damit kann man sich immer helfen. Lesen sollte man halt auch können. Schadet nicht.

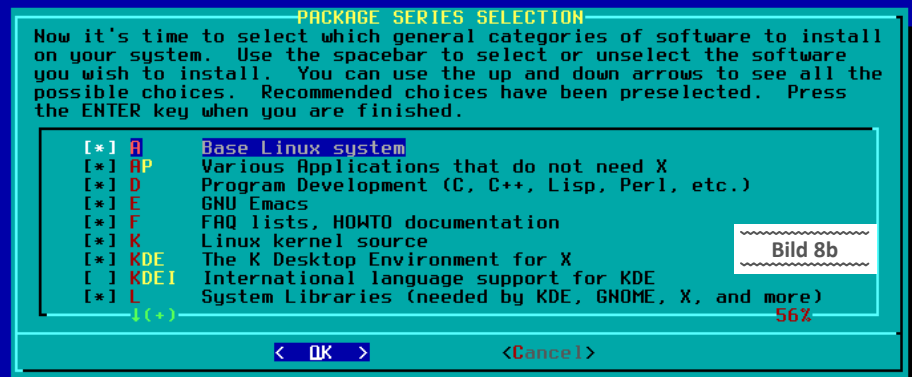
Alle anderen Einstellungen kann man auf der Kommandozeile in einem „ncurses“ Menü auf-

rufen, das aber nur über die Tastatur angesteuert werden kann. Das sieht zumeist sehr archaisch aus, erfüllt aber seinen Zweck. Siehe Bild 8b.

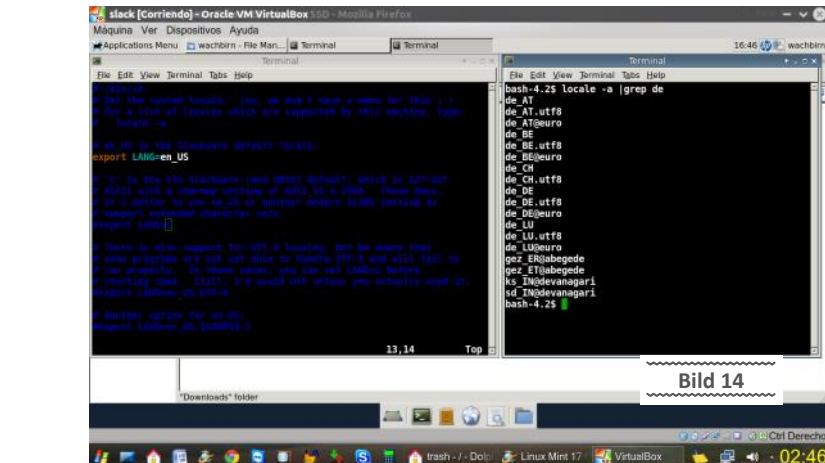
Deshalb ist das Installationsmenü auch seit gut 20 Jahren unverändert. Alte Hasen finden sich sofort zurecht. Auch die Anordnung der Konfigurationsfiles änderte sich über die Jahre nicht.

Nur mal im Gegensatz die Spracheinstellungen bei Ubuntu: siehe Screenshots 9-13.

Und jetzt bei Slackware: editiere das Textfile „lang.sh“ > en_US wird zu de_AT.utf8. Neu-Selecting software to install.



The A (base) series contains the kernel and main system utilities.



start. Fertig. Slackware ist auf Deutsch umgestellt.

Falls Du für alle User auf der Maschine dieselbe Sprache haben willst, kannst Du die Lokalisierung dementsprechend in das File „profile“ reinschreiben. Fertig.

Natürlich braucht man für sämtliche Filemanipulationen erhöhte Rechte. Logo. Sonst könnt ja jeder daherkommen... Siehe Bild 14

Links siehst Du das File, mit der aktuellen Spracheinstellung. In jedem File steht auch

Links siehst Du das File, mit der aktuellen Spracheinstellung. In jedem File steht auch

immer eine Beschreibung. ...no, we dont have a menu for this :-).

Rechts im Bild alle möglichen Lokalisierungen für den deutschsprachigen Raum.

Das heißt, bei Slackware erspar' ich mir schon mal das Geklickse durch's Menü und editier' einfach das File. Sieht nicht so schnuckelig aus, funktioniert aber immer.

Für einen absoluten Anfänger ist das natürlich ein Graus. Bis der sich zurechtfindet, hat der schon sein drittes Smartphone gekauft.

Natürlich musst' das lernen oder Dich ein bisschen „reinfuchsen“. Wenn'st das nicht willst, nimmst Ubuntu, Mac oder Windows.

Du hast die Wahl. Und Slackware läuft auch dementsprechend auf dem Laptop. Eigentlich ist's ja ein Netbook. Die üblichen Verdächtigen noch draufgebügelt: Skype, Team-Viewer und los geht's. *Keep it simple.* Bild 15.

Was mir immer wieder auffällt bei der Windows-Fraktion, ist das penetrante Unterjubeln von versteckter Werbung oder gar gleich einer Software. Speziell „Nur-User“ ohne tiefere Kenntnisse der EDV-Materie sind hier ein leichtes „Opfer“. Das ist meiner Meinung nach auch der Hauptgrund, warum Windows nach einer gewissen Zeitspanne nicht mehr so rund laufen könnte.

Egal, ob man den Flashplayer nur mal schnell installieren will, der McAfee ist sicherheitshalber schon mal „angekreuzt“ und man läßt sich das Zeug unwissentlich gleich mit rauf auf's System. Sobald Du was runterlädst, musst anpassen wie ein „Haftlmacher“.

Hier offenbart sich auch schon das Grundproblem der Windows-Plattform. Deine Software und Programme musst Du Dir im Internet zusammensuchen. Anfänger haben dann meist drei toolbars und zwei Virens Scanner auf dem System installiert.

Natürlich kommt dann immer ein entrüstetes „des wor i net...“, vielleicht noch unterstützt von einem unschuldig dreinschauenden Lexikon-Gesicht > aufschlagen, zuschlagen und immer wieder nachschlagen.

Du merkst selber, was für eine verrohte Bestie die EDV aus mir gemacht hat. Aber ich arbeite daran...

Woher sollen sie es auch wissen. Aber durch diesen Umstand werden Milliarden in der Werbebranche verdient.

Alleine was Google, das größte Marketingunternehmen der Welt, mit dieser Masche verdient, ist absurd.

In Linux beziehst Du Deine Software immer von einem Paketmanager. Der kümmert sich um Deine Programme und hält sie auch aktuell. Ist so ähnlich wie Windows-8 mit seinem Store.

Sieh mal. Bild 16. Das ist mein Paketmanager.

Grün ist installiert. Calibre, der E-Book-Konverter ist gerade angezeigt. Du hast hier „keine Chance“, Dir irgendwie unabsichtlich oder nicht, unnötige Werbung oder andere Software „einzutreten“.

Das war auch der Hauptgrund für die schlechte Performance des Netbooks der Bekannten > zu viel Schrott drauf.

Alleine bei Skype ist's ja schon ein mittleres Drama. Schau Dir mal Bild 17 an.

Links meine Linux-Version und rechts die für Windows-7. Na klar wird da ein Anfänger mal draufdrücken. Ob absichtlich oder nicht. Es lenkt ab vom Wesentlichen.

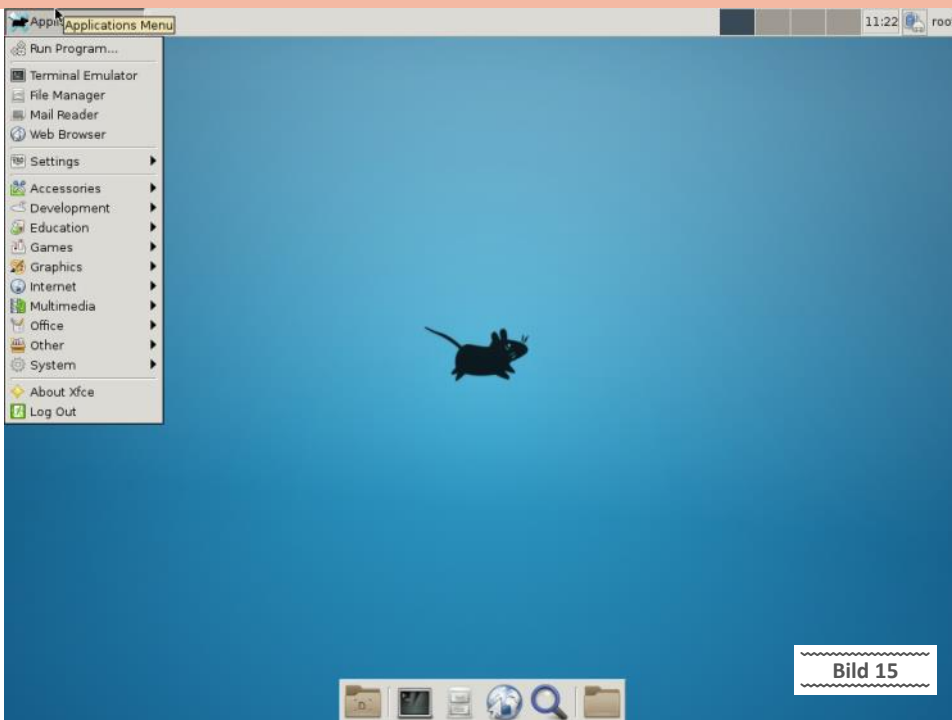


Bild 15

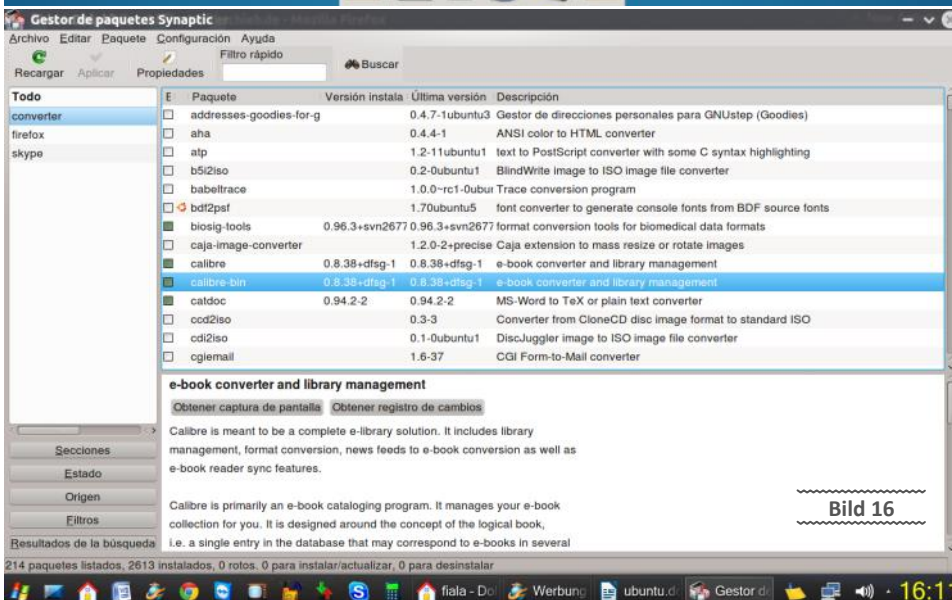


Bild 16

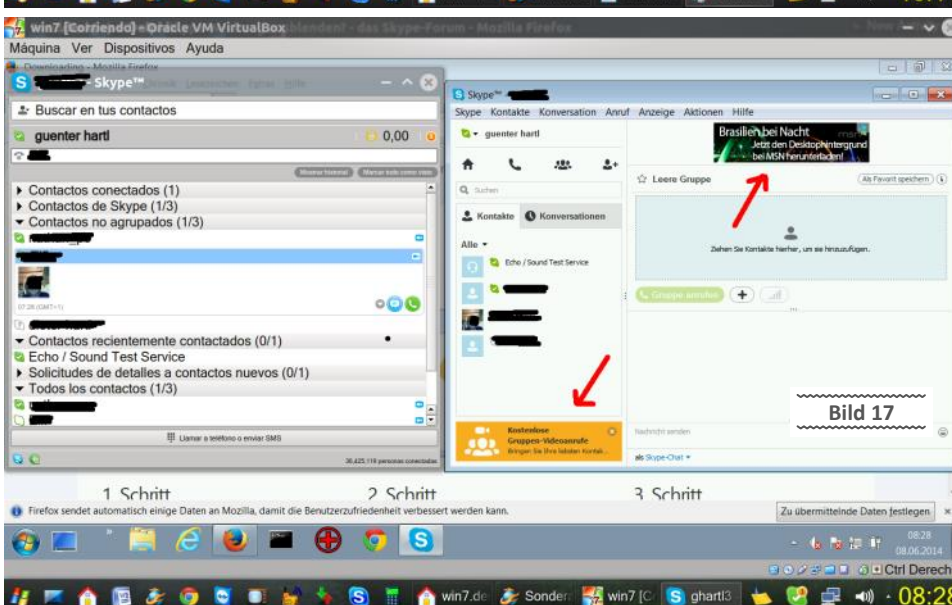


Bild 17

Abbrechen das Glump? „Geh in die hosts Datei, schreib das rein und dann leere den DNS Cache; wenns später wieder kommt, leere noch mal den Cache...“. Was soll ein Enduser ohne tiefere Systemkenntnisse mit so einer Aussage anfangen? Vergiss das.

Solange Programme im Internet zu besorgen sind, wird's immer schwer bleiben. Da red' ich noch nicht mal von den umgeleiteten Links auf irgendwelche dubiosen Seiten.



Nur zur Info, wir reden da nicht über exotische Programme. Eigentlich sollten die Programme schon Teil des Systems sein.

„Sound-Converter“ unter KDE wandelt YouTube-Videos in Mp3 um. Einfach und intuitiv, ohne dass ich da die URL in eine Webseite eintragen muss und was weiß ich noch. Siehe Bild 18.

Na klar kannst Du unter Windows 17 verschiedene Konvertierungs-Tools installieren. Aber nicht von einer zentralen Quelle aus. Darum halte ich das Konzept eines Paketmanagers (oder zumindest eines guten Store-Konzeptes) für unentbehrlich.

Nur zur Info: in Linux-Paketmanagern ist nie Shareware oder proprietäre Software verfügbar.

### Der ganz normale Wahnsinn

Ich könnte stundenlang darüber schwafeln, was mir so im Alltag unterkommt. Ein ständiges Ärgernis sind die zunehmend inflationär verwendeten Euphemismen, mit denen jeder Mensch mal mehr, mal weniger konfrontiert wird. Meiner Meinung nach sind diese Euphemismen nur dazu gedacht, Kompetenz, Intelligenz oder was weiß ich vorzutauschen.

Hier mal ein paar Auszüge, die mir öfters über den Weg laufen:

- **Überkapazitäten im Personalbereich abbauen** > Kündigungswelle
- **Freisetzungsprozesse** > Massenkündigungen
- **Lenkungen der Ausgabenströme** > Kürzungen der Geldmittel
- **Sparsaßnahmen** > Kürzungen (hat nichts mit Sparen zu tun)
- **Preisanpassungen** > Preiserhöhungen
- **Infopoint** > Beschwerdestelle
- **nonterritoriales Arbeitsplatzkonzept** > kein „eigener“ Schreibtisch in Firma
- **Entwicklungsfelder** > Schwächen (falls Du bei einem Vorstellungsgespräch danach gefragt wirst..schau wenigstens nicht zu belämmert drein)
- **„bail in“-Konzept** > bei einer Bankenpleite haften auch die Sparer. Die EU ist gerade dabei, das Durchzunicken.
- **Migrationshintergrund** > Leute mit arabisch-türkischen Wurzeln
- **Rotationseuropäer** > Roma
- **Anchorman** > Nachrichtensprecher/in ?

Diese Schwurbel-Begriffe wurden nur erfunden, damit man das Kind nicht beim Namen nennen muss. Obige Beispiele zeigen mir nur unmiss-

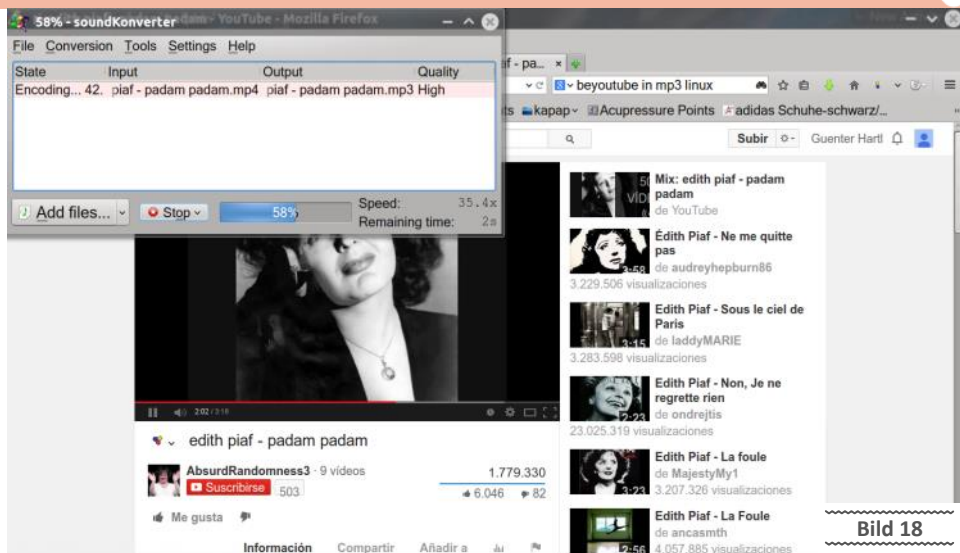


Bild 18

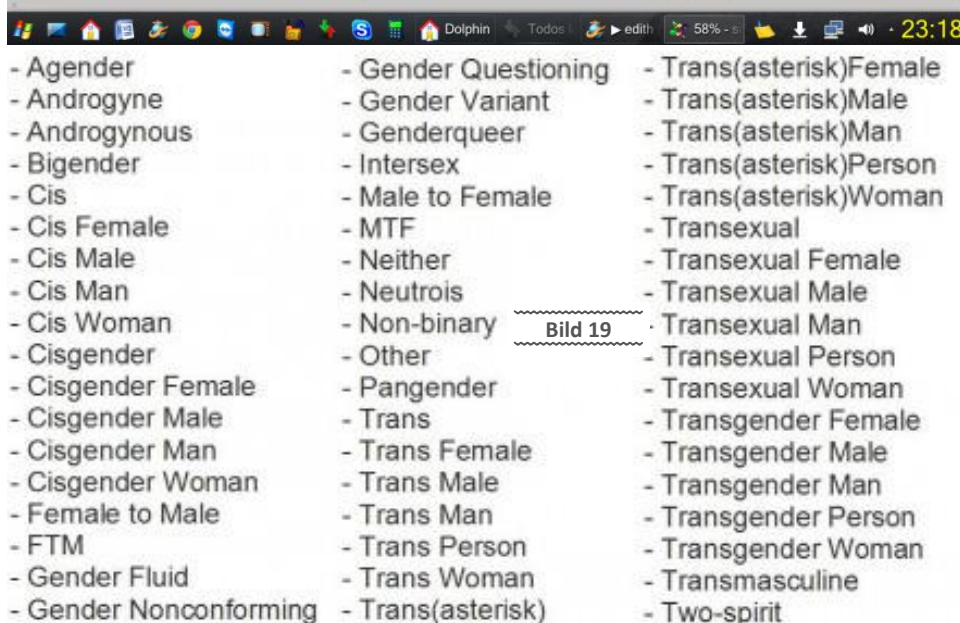


Bild 19

verständlich, was passieren kann, wenn Kondome versagen.

Das Ganze noch durchmischt mit einer Prise Genderwahn. Beispielsweise kannst Du bei Facebook (derzeit nur im englischsprachigen Raum) aus 54 Geschlechtern Dein passendes auswählen. Kein Witz. Bild 19.

Alles wird gut. Natürlich kannst Du das alles mit einer lässigen Handbewegung wegwischen und mit einem trotzigem „mir doch egal...“ reagieren.

Abgesehen davon, dass „egal“ die kleine Schwester von „leck mich am Arsch ist“... Wenn

Du Pensionist bist oder in einer geschützten Werkstätte arbeitest, geb' ich Dir recht.

Im Berufsleben wirst Du aber einen schweren Stand mit dieser Haltung haben. Speziell im Trainer-und Lehrtätigkeitsbereich.

Sodah, um jetzt noch mal die Kurve zu meiner Artikelüberschrift zu bekommen > reißt Euch z'samm!!!

Man liest sich

Gruß Günter

# Sony Xperia Z2 – das bessere Z1?

Paul Belcl

Ich habe nun schon seit März 2013 das Sony Xperia Z im Dauertest. Zwischendurch auch kurz ein Xperia Z1. Nun konnte ich fast 3 Wochen das neue Xperia Z2 in der Praxis testen.

Ich werde hier nicht detailliert auf alles eingehen, sondern die Dinge beschreiben, die mir besonders aufgefallen sind.

## Hardware

Das Sony Xperia Z2 macht einen sehr edlen Eindruck. Metallrahmen, Glas Rückseite und das alles mit silberfarbigem Rand. Die quadratische Form des Gerätes mit den scharfen Kanten, ist ja seit dem ersten Modell fast schon das Markenzeichen. Alle Anschlüsse bis auf den Kopfhörer-Anschluss, sind hinter wasserdichten Klappen versteckt. Stereo-Lautsprecher, Kamera-Taste und Öffnung für Halteschlaufe, alles da.

Im Vergleich zum Vorgänger ist das Z2 etwas dünner geworden. Dafür ist das Display etwas größer. Das quadratische Design ist geblieben. Der Stereo-Lautsprecher an der Vorderseite, lässt sich nun nicht mehr so leicht mit der Hand abdecken, wie bei meinem Xperia Z.

## Bildschirm

Das Display hat 5,2 Zoll und Full-HD-Auflösung. Obwohl dadurch die Pixeldichte im Vergleich zum Vorgänger etwas geringer ausfällt, ist der Bildschirm hervorragend abzulesen. Satte lebendige Farben und ein sehr guter Blickwinkel von der Seite.

Bei der Hintergrundbeleuchtung kann eine bestimmte Grundhelligkeit vordefiniert werden, von der aus dann die automatische Helligkeitsregelung ausgeht. Ich hatte damit während meines Tests immer eine hervorragende Ablesbarkeit. Egal, ob in der Sonne oder in der U-Bahn!

Mit der X-Reality-Bildverbesserung kann man die Farben zwar etwas kräftiger angezeigt bekommen, sie wirken aber manchmal schon fast überzeichnet.

Das Display des Z2 ist jedenfalls eines der besten seit ich Smartphones teste; es lässt wirklich keine Wünsche offen!

## Ton

Der Klang des Gerätes ist etwas besser als beim Vorgänger. Die Stereolautsprecher am unteren und oberen Bildschirmrand sind nicht nur gut versteckt, sondern geben Musik und Videos mit ordentlichem Klang wieder.

Bei Alarmen oder wenn das Telefon läutet, habe ich oft den Wunsch, die Lautstärke noch etwas zu erhöhen obwohl sie schon voll aufgedreht ist. Hier sollte Sony noch nachbessern, denn so wie es aussieht, wird bei der Ausgabe von Systemtönen nur einer der beiden Lautsprecher verwendet. Das könnte ruhig lauter sein...

## Kamera

Das absolute Highlight des Z2 ist meiner Meinung nach die Kamera. Die Fotos werden mit dem Auslöse-Knopf am Gehäuse geknipst und entstehen in guter Geschwindigkeit. In meinem Test machte die Kamera meist sehr farbtreue und gut belichtete Bilder, auch bei schlechterer Beleuchtung.

Nur bei sehr dunkler Umgebung gab es manchmal eine Zeitverzögerung beim Scharfstellen. Das Bildrauschen ist auch bei Aufnahmen im Innenraum ohne Blitz erträglich.

Videoaufnahmen in HD-Qualität kommen farbgetreu und ruckelfrei

Auch 4K-Videoaufnahmen sind mit dem Z2 möglich, allerdings habe ich mich damit mangels passendem Abspielgerät nicht beschäftigt. Ich persönlich halte den Hype für diese Technik generell für übertrieben, da es derzeit weder viel brauchbares Material zum Abspielen, noch preiswerte Abspielgeräte gibt. Und die Größe des gefilmten Materials steht nur sehr bedingt im Zusammenhang mit dem Nutzen. Manchen Leuten wird es aber möglicherweise helfen, das eigene Ego zu stärken ;-)

## Software und Erweiterungen

Das Betriebssystem Android 4.4.2. funktioniert stabil und die Änderungen an der Oberfläche halten sich in Grenzen. Allerdings hat Sony verhältnismäßig viele Programme vorinstalliert. Walkman-App, eine nette Album- und Video-Anwendung, Sony Select, Social Life, Video Unlimited (Sonys Streaming-Dienst), TrackID und TrackID TV (ein Shazam-Pendant), E-Book-Reader samt Store-Anbindung, PlayStation Mobile und PlayStation Network, Garmin-Navigation (Trial Version!), Fernbedienungs-App für den Sony TV. Wer sie mangels Sony Entertainment Hardware nicht braucht, muss auf den Speicherplatz, den sie belegen, leider verzichten.

Einige dieser Programme sind allerdings wirklich brauchbar! Ich konnte daher bei meinem Test viele Zusatzsoftware weglassen, die ich normalerweise installiere.

Beispielsweise die Walkman-App oder die Kamera App mit vielen Spielereien und Bildeffekten, als auch eine Software zum Abspielen von Videos.

Auf dem Home-Screen sind einige Widgets über aktuelle Filme, Musik oder Apps aus Sonys Stores verfügbar.

In den Einstellungen sind mir ebenfalls viele sinnvolle Erweiterungen aufgefallen, die dem Benutzer den Alltag erleichtern. Ortsbezogenes Wifi gibt es ja schon länger. Die Gestensteuerung bei Anrufen ist allerdings erst seit kurzer Zeit dabei. Man kann damit einen Anruf ablehnen, indem man das Telefon schüttelt oder ihn stummschalten, indem man das Telefon mit dem Display nach unten ablegt.

Annehmen ist durch einfaches Aufheben zum Ohr möglich. Die Funktion reagiert zwar etwas zeitverzögert, ist aber durchaus brauchbar!

Wenn man das Einstellungsmenü genau untersucht, findet man viele weitere nützliche Helferlein zur Steuerung. Tippen zum Aufwecken, intelligente Gegenlichtsteuerung, Software zur Profilverwaltung und viele andere nützliche Dinge!

Zur Verbindung mit anderen Geräten (Computer, Fernseher, DNLA Geräten) gibt es ausreichende Möglichkeiten über WLAN, Bluetooth oder andere Techniken.

Nur drahtlose Ladetechnik gibt es nicht, aber dazu später mehr...

## Custom Roms geduldet?

Interessante Information für Leute, die ihr Gerät rooten möchten, oder gerne Custom Roms auf ihr Smartphone spielen. Bei Sony ist, zum Unterschied von anderen Herstellern möglich, den Boot-Loader des Gerätes offiziell zu entsperren.

Auf der Seite:

<http://developer.sonymobile.com/unlockbootloader> kann man diesen Vorgang in nur drei Schritten durchführen. Sony macht allerdings darauf aufmerksam, dass man im Garantiefall die Kosten für eine Installation der Originalsoftware extra berechnet, falls diese verändert wurde!

Und dass natürlich nach Entsperren des Boot-Loaders kein Softwaresupport mehr gewährleistet ist.





# Aus für Vorratsdatenspeicherung

Werner Illsinger

Heute hat der Österreichische Verfassungsgerichtshof, die anlasslose Vorratsdatenspeicherung, die am 1. April 2012 in Österreich eingeführt wurde und bei der das Kommunikationsverhalten aller Österreicher "auf Vorrat" gespeichert wurde, für Verfassungswidrig befunden und das Gesetz vollständig aufgehoben.

Zuvor hatte schon der Europäische Gerichtshof die zugrunde liegende Richtlinie, die die Mitgliedsländer dazu verpflichtet hatte die Vorratsdatenspeicherung einzuführen, als nicht Grundrechtskonform aufgehoben.

Herzlichen Dank an dieser Stelle der AK Vorrat (im Besonderen an Andreas Krisch und Thomas Lohninger) sowie dem Nationalratsabgeordneten Albert Steinhauser. Aber auch ein großes Dankeschön an unsere Mitglieder die zuerst die Bürgerinitiative und später die Klage gegen das Gesetz unterstützt haben. Immerhin sind 11.139 Menschen in Österreich darunter auch viele unserer Mitglieder und ich selbst als Mitkläger aufgetreten. Es ist ein großer Tag für die Menschenrechte und auch die Demokratie in Österreich und es zeigt sich einmal mehr, dass es sich

auszahlt sich für etwas einzusetzen und nicht alles als gegeben hinzunehmen.

Ich werde immer wieder gefragt, warum die Vorratsdatenspeicherung (oder auch die generelle Überwachung) schlecht sei. Sie habe doch auch positive Effekte wie z.B. die Verbrechensbekämpfung. Das ist unbestritten und auch der Verfassungsgerichtshof gesteht ein, dass auf neue technologische Entwicklungen auch neue gesetzliche Antworten gefunden werden müssen. Die Vorratsdatenspeicherung ist aber nicht die richtige Antwort. Der Verfassungsgerichtshof in seiner Presseaussendung zum Schutz der Kommunikation:

Das Grundrecht auf Datenschutz, so der Verfassungsgerichtshof, ist in einer demokratischen Gesellschaft auf die Ermöglichung und Sicherung vertraulicher Kommunikation zwischen den Menschen gerichtet. Der Einzelne und seine freie Persönlichkeitsentfaltung sind nicht nur auf die öffentliche Kommunikation in der Gemeinschaft angewiesen; die Freiheit als Anspruch des Individuums und als Zustand einer

Gesellschaft wird bestimmt von der Qualität der Informationsbeziehungen.

Eine demokratische Gesellschaft kann sich nur entwickeln, wenn Menschen keine Angst haben müssen wenn sie kommunizieren. Die Beobachtung eines Menschen verändert bereits sein Verhalten. Deswegen ist die Wahrung der Privatsphäre gerade in Kommunikationsbeziehungen für eine freie demokratische Entwicklung besonders wichtig.

Eine besonders schlechte Figur haben in dieser Sache unsere Bundesregierung(en) abgegeben. Zum einen die ÖVP geführte Regierung unter Wolfgang Schüssel und Karin Gastinger (Justiz) – da sie damals nichts auf EU Ebene gegen die EU Richtlinie unternommen haben. Und die derzeitige Regierung – die unter Kanzler Faymann und Justizminister Brandstetter, die obwohl es dem Urteil des EuGH nicht an Klarheit fehlte das Gesetz zur Vorratsdatenspeicherung bis zur bitteren Neige verteidigt haben.

Links siehe <http://www.illsinger.at/wordpress/?p=756>



Dank der intelligenten Anrufverwaltung können Sie eingehende Anrufe ohne den Bildschirm berühren zu müssen annehmen oder ablehnen.

- Annehmen: Führen Sie das Gerät an Ihr Ohr heran.
- Ablehnen: Schütteln Sie das Gerät.
- Klingeln stumm schalten: Legen Sie

Ich gehe allerdings davon aus, dass dieses Zugeständnis keine Auswirkung auf den Hardware Support hat, zumindest klingt es so :-)

## Warum ist das interessant?

Weil ich bereits einen Fall kenne, wo ein anderer Smartphone Hersteller mit „S“ die Garantieleistung für einen defekten Lautstärkknopf (Hardwaredefekt) verweigert hat, nur weil der Benutzer ein Custom Rom drauf hatte!

Und das, obwohl der Benutzer zuvor das Original Rom wieder aufgespielt hatte.

Ich hoffe, dass Sony in solchen Fällen kulanter oder zumindest realistisch reagiert!

Denn in den meisten Fällen will man das Gerät nur rooten, um eine komplette Datensicherung

machen zu können und verändert die Originalsoftware dadurch nicht!

## Kabellos aufladen – nur mit Sony Dock

Durch die wasserdichte Ausführung ist es auch beim Sony Xperia Z2 nötig, bei jedem Ladevorgang den Deckel für den Ladeanschluss zu öffnen. Diese Klappe macht bei allen Sony Modellen, die ich bis jetzt getestet habe, einen stabilen Eindruck! Trotzdem ist es nach einiger Zeit etwas lästig, die Klappe dauernd aufzufummeln.

Leider hat Sony in seine Geräte bis jetzt noch keine induktive Lademöglichkeit eingebaut (z.B. nach dem Qi-Standard).

Im Internet kursieren allerdings Gerüchte, dass Sony ein Qi-Lade Etui (unter „Wireless Charging Cover WCR12“) und ein Qi-Ladegerät („Wireless Charging Plate WCH10“) für das Xperia Z2 herausbringen wird. Die Geräte sind im Internet schon zu finden, aber kaufen konnte man sie zur Drucklegung des Artikels noch nicht!

Kaufen kann man derzeit allerdings das Sony Ladedock DK36, welches ich allerdings nicht getestet habe. In dieses kann man das Z2 mit magnetischer Hilfe eindocken und laden, ohne irgendwelche Klappen zu öffnen

## Fazit

Das Sony Xperia Z2 ist das bessere Z1!

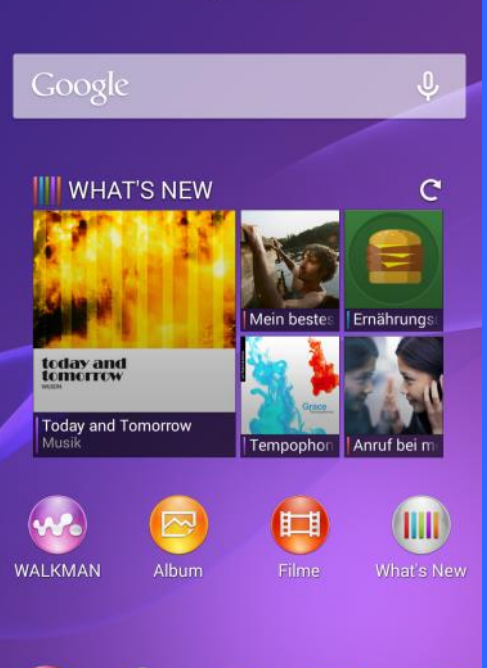
Wenn man allerdings bereits ein Z1 besitzt, wird das Z2 nicht zu einem sofortigen Umstieg verführen, dazu sind die Änderungen zu gering. Wenn allerdings ein Neukauf ansteht, sollte man sich das Z2 genauer ansehen.

Die Kamera ist eine der besten, die ich je in einem Smartphone verwendet habe!

Und der hervorragende Bildschirm lässt wirklich keine Wünsche offen!

Damit kann das Sony Xperia Z2 ohne Probleme mit der Smartphone Elite von HTC und Samsung mithalten.

Mir persönlich ist das Z2 schon fast etwas zu groß! Es passt in keine, derzeit verfügbare Gürteltasche und in der Hosentasche macht es



auch keine schlanke Figur obwohl es sehr dünn ist!

Das Softwareangebot von Sony ist üppig, speziell für Leute die weitere Sony Komponenten haben ist alles da! Die Software ist gut und in vielen Fällen kann sie durchaus mit Programmen aus dem Playstore mithalten.

Die vielen, meist etwas versteckten Helferlein, die Sony in der Software eingebaut hat, machen das Leben mit diesem Smartphone in vielen Bereichen zur Freude. Man muss sich allerdings auch die Mühe machen, diese zu finden und einzurichten.

Die hochwertige Verarbeitung des wasserdichten Gerätes ist meiner Meinung nach besser als bei vielen Mitbewerbern. Das rechtfertigt auch den Preis von derzeit knapp unter € 500,-.

# Sony Smartband SWR10

Paul Belcl



Smartbänder, Smartwatches, Wearables u.s.w sind derzeit in aller Munde. Im Zuge des Sony Xperia Z2 Tests, wurde mir von Sony angeboten, auch das Smartband SWR10 auszuprobieren.

## Erster Eindruck - Hardware

Das SWR10 besteht aus zwei Teilen. Die Basiseinheit des Armbandes ist ein kleines weißes Plastikrechteck. Darin befinden sich die gesamte Elektronik, der Micro USB Ladeanschluss, sowie eine Taste und drei LEDs.

Teil zwei ist ein Silikonarmband wo die Basiseinheit hineingeschoben wird. In der Schachtel liegen zwei schwarze Armbänder mit unterschiedlichen Längen. So ist sichergestellt, dass jeder Handgelenkumfang bedient werden kann. Die Armbänder gibt es als Zubehör auch in vielen unterschiedlichen Farben zu kaufen!

Mit einem Edelstahl Verschluss, der einem Manschettenknopf ähnelt, kann man das Armband

am Handgelenk befestigen. Das SWR10 ist wasserdicht (Zertifiziert nach IP58).

Die Oberseite des Armbandes, wo das Basiselement drinsteckt, ist ca. 4 mm hoch.

Die Höhe stört nicht, aber das Material rutscht schlecht, was manchmal beim Anziehen von Kleidungsstücken mit langen Ärmeln stört.

Der schwarze Kautschuk oder Gummi des Armbands ist außerdem ein Staubmagnet. Alles was so an Fuzeln oder statischen Verunreinigungen in seine Nähe kommt, haftet auf dem Armband. Allerdings lässt sich das Band unter fließendem Wasser abwaschen, somit kann es genau so leicht wieder gereinigt werden, wie es schmutzig wird.

In zwei Fällen während meines fast dreiwöchigen Tests bin ich mit dem Armband so fest an einem Gegenstand hängen geblieben, dass es aufgegangen ist und herunterfiel. Kein Problem, denn das Armband ist dadurch nicht beschädigt worden. Man sollte allerdings beim Tragen berücksichtigen, dass das Band etwas vom Handgelenk absteht, und keine gleitende Oberfläche hat.

Die Akkulebensdauer von ca. 5-7 Tagen ist ein durchaus brauchbarer Wert.

Der kleine Akku ist in etwas mehr als einer Stunde auch wieder aufgeladen. Die drei LEDs der Basiseinheit sind auch durch das Armband gut zu sehen.

## Software

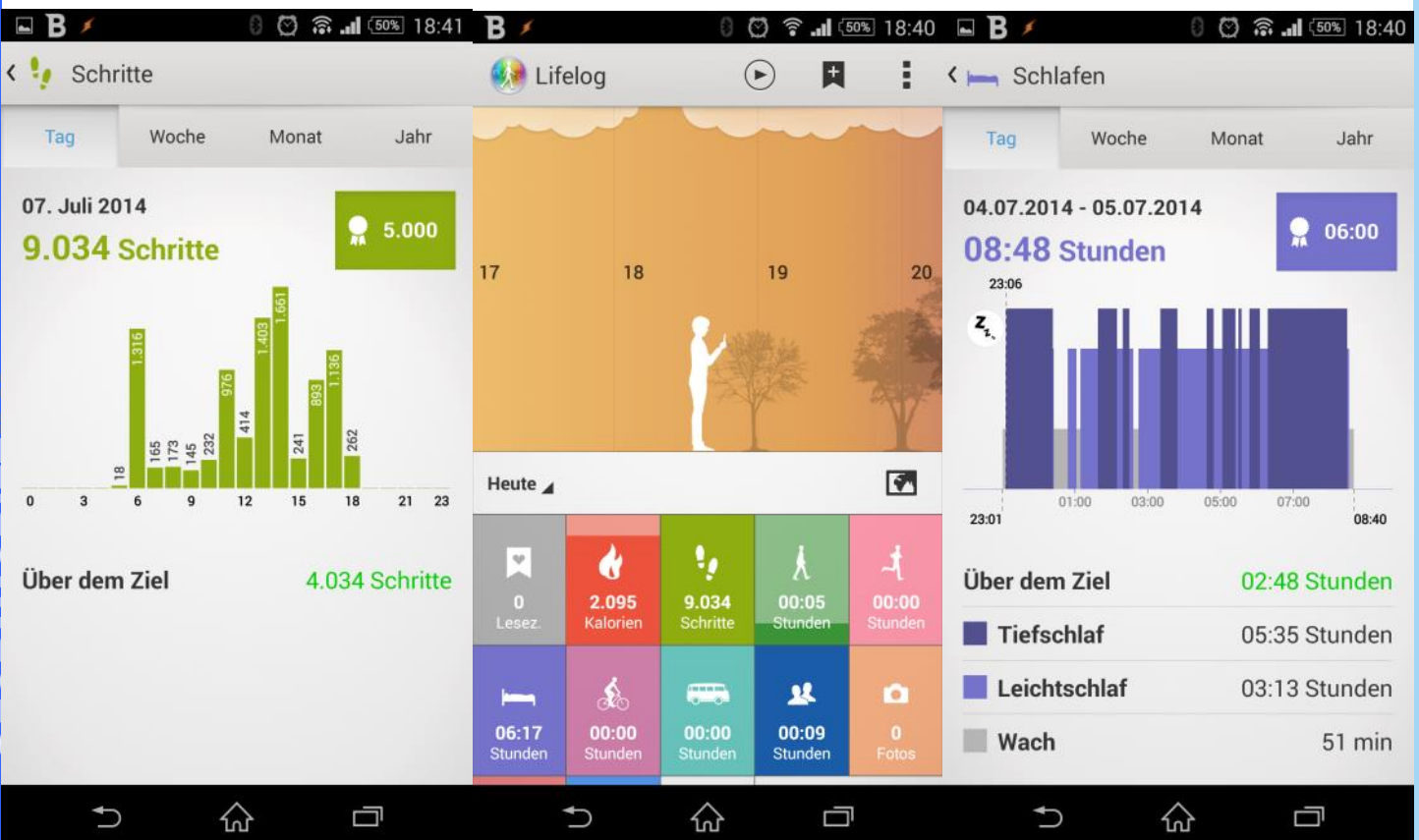
Um Daten vom Armband auszulesen sind einige Apps auf dem Smartphone nötig.

- Smart Connect App  
Baut die Verbindung zum Smartband auf

- Lifelog App  
zeigt die aufgezeichneten Daten an und bereitet sie auf

Wenn die Programme noch nicht vorinstalliert sind, wie auf meinem Z2, kann man sie im Playstore kostenlos runterladen. Grundsätzlich funktioniert das SWR10 mit jedem Smartphone ab Android Version 4.4 welches *Bluetooth Low Energy* unterstützt.

Bevor es jedoch loslegen kann, muss man ein Benutzerkonto bei Sony anlegen. **Das** ist für mich der erste Minuspunkt, denn es ist meiner Meinung nach nicht zwingend nötig, diese Daten in die Sony Cloud zu schicken! Sony ist der Meinung, dass die zentrale Speicherung ein besseres Benutzererlebnis ermöglicht. Das stimmt grundsätzlich, speziell wenn man oft Geräte wechselt. Allerdings wäre eine Option, die Daten auch in einen Cloud-Service wie Dropbox zu sichern. Da habe ich sie wenigstens unter meiner Kontrolle! Eine solche Möglichkeit dazu habe ich allerdings im Programm nicht gefunden. Also werden meine Testdaten in der Cloud bei Sony gespeichert.





Erfasst werden folgende Ereignisse

- Schlafzyklus: Leichtschlaf, Tiefschlaf und Wachphasen, Schlafdauer
- Schrittzähler
- Kalorienverbrauch
- Zeit gehend
- Zeit laufend
- Zeit radfahrend
- Zeit in öffentlichen Verkehrsmitteln
- Zeit telefonierend (nur auf Sony Geräten)
- Anzahl der geschossenen Fotos (nur auf Sony Geräten)
- Zeit Musik hörend (nur auf Sony Geräten)
- Zeit im Internet

Auf Sony Telefonen kann die Software auch mit den Multimedia Apps Kontakt aufnehmen, von dort Informationen auslesen und diese auch steuern; aber dazu später mehr.

### Praxistest

In der Lifelog App werden alle diese Informationen übersichtlich angezeigt und man kann daraus einige interessante Rückschlüsse auf gewisse Abläufe ziehen.

Ich bin sehr interessiert mehr über mein Schlafverhalten zu erfahren, daher spricht mich diese Funktion besonders an.

Mit meiner Pebble, geht das zwar auch, aber nur unter dauernder Funkverbindung mit dem Handy, was beim Schlafen nicht empfehlenswert ist.

Das SWR10 kann diese Informationen ohne Funkverbindung aufzeichnen während ich schlafe.

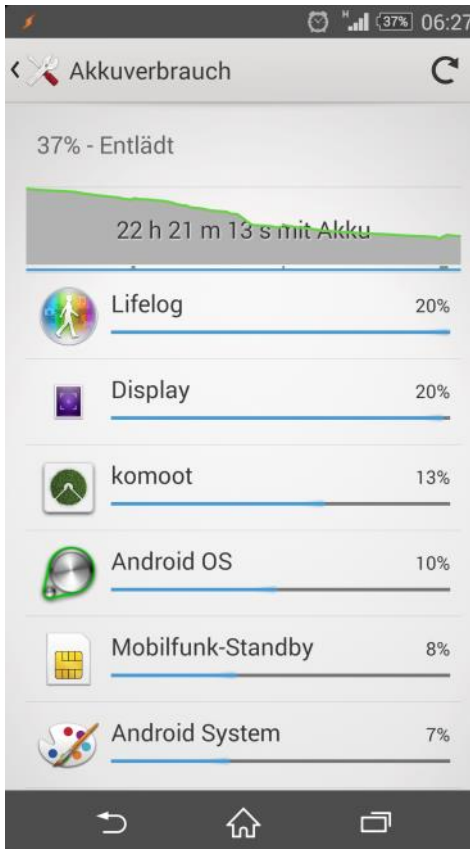
Man kann in der LifeLog App ein bestimmtes Schlafziel in Stunden angeben, welches bei der Auswertung dann berücksichtigt wird.

Am nächsten Tag in der Früh, wenn ich mein Smartphone einschalte, werden die gesammelten Daten automatisch übertragen. Die Lifelog App zeigt alle vom Smartband aufgezeichneten Daten in einer übersichtlichen Zusammenfassung an. (siehe Abbildung). Es gibt Ansichten für Tag, Woche und eine Monatsübersicht. Das gefällt mir sehr!

Die aufgezeichneten Daten entsprechen, sofern ich sie nachvollziehen konnte, der Wahrheit.

Der Schrittzähler funktioniert recht brauchbar, man kann sich damit anzeigen lassen wie viele Schritte man, verteilt über einen bestimmten Zeitraum (Tag, Woche, Monat) zurücklegt. Auch die verbrauchten Kalorien pro Tag kann man sich anzeigen lassen. Wie das Programm die Kalorienwerte errechnet, kann ich nicht ohne großen Aufwand prüfen, somit muss ich glauben, was ich lese ;-)

Leider hatte das SWR10 in meinem Test Probleme, die Zeiten beim Radfahren und in öffentlichen Verkehrsmitteln richtig zu erfassen. Ich



kann mir sowieso nicht vorstellen, wie solche Bewegungsmuster zuverlässig zugeordnet werden können. Aber wenn die Funktion angeboten wird, sollte sie auch richtige Ergebnisse liefern!

Da ich diese beiden Verkehrsmittel nahezu täglich nutze, wäre hier ein großes Testpotential vorhanden gewesen.

Ich bekam aber an **keinem** Tag auch nur ein halbwegs richtiges Ergebnis dieser beiden Bewegungsmuster angezeigt. Bei der Zeit in öffentlichen Verkehrsmitteln, wurden nachweislich auch Zeiten erfasst, die ich in meinem privaten PKW saß!

Die Radfahrzeit stimmte an zwei Tagen nur ungefähr, die meiste Zeit gab es fürs Radfahren **keine** Daten, obwohl ich oft an solchen Tagen oft mehr als 15 km/h fuhr.

Hier gibt es also noch Verbesserungspotential!

In der Software am Smartphone kann man das Armband so konfigurieren, dass sich damit bestimmte Apps fernsteuern lassen. Das funktioniert allerdings nur mit ausgewählten Programmen, meist sind es Sony Apps!

Ist alles richtig eingestellt, drückt man einen Knopf auf dem Armband um die Funktion zu aktivieren. Danach klopft man ein- bis dreimal auf die Oberseite des Bandes. Je nach Konfiguration kann damit je eine Funktion aktiviert oder deaktiviert werden.

Zum Beispiel kann man mit einmal klopfen auf das Band, die Kamera im Handy ein Foto machen lassen, oder mit zweimal klopfen das Tele-

fon klingeln lassen, wenn man es verlegt hat. Die Sony Walkman Software kann damit zum nächsten Titel schalten usw.

Für mich als Pebble-Besitzer keine Besonderheit, aber wenn man nur simple Dinge über so eine Fernsteuerung machen möchte, reicht die Funktion des Armbands durchaus!

Gewisse Zustände am Smartphone lassen sich auf dem Armband durch Vibrieren bemerkbar machen. Beispielsweise wenn das Telefon läutet oder ein SMS kommt. Auch eine Weckfunktion, die das Armband zu einer gewissen Zeit vibrieren lässt, kann man einstellen. Alles ganz nützlich!

Abschließend muss noch gesagt werden, dass die Lifelog App in meinem Test, manchmal einen relativ hohen Akkuverbrauch erzeugt hat, wie man in der Abbildung sehen kann. Ich vermute, das passiert dann, wenn man die App nicht richtig beendet. Genauer untersucht habe ich diese Sache allerdings nicht!

### Fazit

Das Sony Smartband SWR10 hat einige durchaus interessante Funktionen im Angebot: die Schlafanalyse, den Schrittzähler, den Kalorienzähler; auch die einfache Fernsteuerung ist gut gelungen. Auch der Vibrationswecker hat durchaus Potential.

Besonders gefällt mir die Aufzeichnung des Schlafverhaltens **ohne** Funkverbindung zum Handy!

Auch die Übersicht der Informationen in der Lifelog Software ist wirklich gut gemacht!

Will ich ein SWR10 Smartband haben? **Nein!**

Die Informationen, die das SWR10 sammelt, sind mir persönlich nicht so wichtig, dass ich dafür ein Armband tragen möchte. Ich würde es eventuell kaufen, um meinen Schlaf zu protokollieren. Aber wie lange interessiert mich diese Information? Und möchte ich, dass sie bei Sony gespeichert ist? Eher nein!

Mir persönlich fehlt auf dem Band eine Anzeige!

Auf meiner Pebble kann ich Zustände des Smartphones anzeigen lassen (Anrufe, SMS, usw.), Zeit ablesen und vieles mehr!

Das Sony Smartband ist etwas für Leute, die Informationen sammeln wollen ohne ein auffälliges „Gadget“ am Handgelenk zu haben. Dafür ist das Sony Smartband hervorragend geeignet!

Ob das SWR10 den Preis von etwas über € 80,- wert ist, muss jeder für sich selbst entscheiden.

Ich persönlich finde, dass all diese „Gadgets“ derzeit noch etwas zu teuer verkauft werden, da sie gerade boomen. Aber vielleicht interessiert es euch ja gerade deshalb!

# Mikrocontroller-News aus Hollabrunn

Manfred Resel

Wie schon in der PCNEWS-125 und 135 vorgestellt, wird in unserer Elektronikabteilung als Mikrocontroller der neue Industriestandard ARM-Cortex-M0 bis M4 eingesetzt. Für die Prototypenfertigung werden die LQFP-Gehäuse auf einer Leiterplatte bestückt und mit Stiftleisten für einen Steckbrettaufbau tauglich gemacht. Dieser Artikel beschreibt drei Diplomarbeiten sowohl mit Cortex-M0-M3, Infineon Cortex-M4 Eval-Boards als auch mit mbed-kompatiblen ARM-DIL-Modulen.

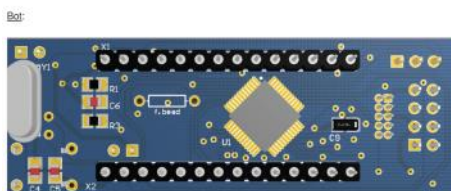
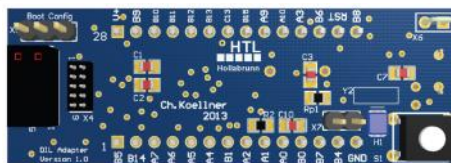
## Bienenschwarm-Frühwarnsystem

Im Rahmen einer Diplomarbeit wurde von einem Schülerteam der Abteilung Elektronik ein Bienenschwarm-Frühwarnsystem entwickelt.

Die Honigbiene hat für die Landwirtschaft und das gesamte Ökosystem eine zentrale Bedeutung. Die Bienenhaltung ist aber heute in einer großen Krise, es gibt keine natürlichen Nistmöglichkeiten mehr und die Bienen können nicht alleine mit den von außen eingeschleppten Parasiten (Varroamilbe) zurecht kommen. Bienenschwärme hätten aber die besten Voraussetzungen für eine gesunde Entwicklung. Der Schwarmvorgang wird häufig als ein Heilungsprozess bezeichnet. Beim Auszug aus dem Muttervolk lassen die Schwarmbienen nicht nur fast alle Krankheitserreger, sondern auch den allergrößten Teil der Varroamilben zurück. So können sie sich in einer neuen Bienenwohnung eine hervorragende Grundlage für die nächsten Jahre schaffen.

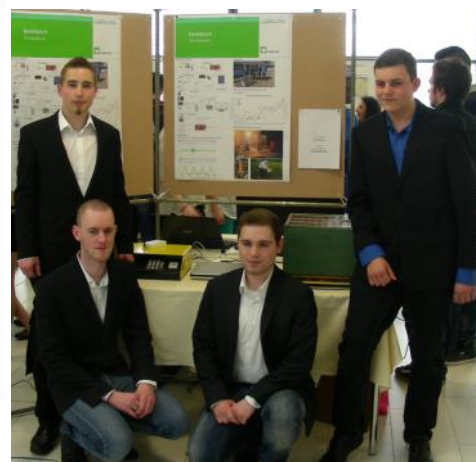
Das Ziel dieser Arbeit bestand daher darin den Bienenschwarm rechtzeitig zu erkennen und den Imker oder die Imkerin über E-Mail oder SMS zu verständigen, damit die Schwarmtraube nicht verlorengeht sondern wieder neu einquartiert werden kann.

Das von den Schülern entwickelte Messgerät erfasst über ein Mikrofon signifikante Geräusche im Bienenstock, ermittelt über eine Waage das Gewicht und rechnet über die wochenlang

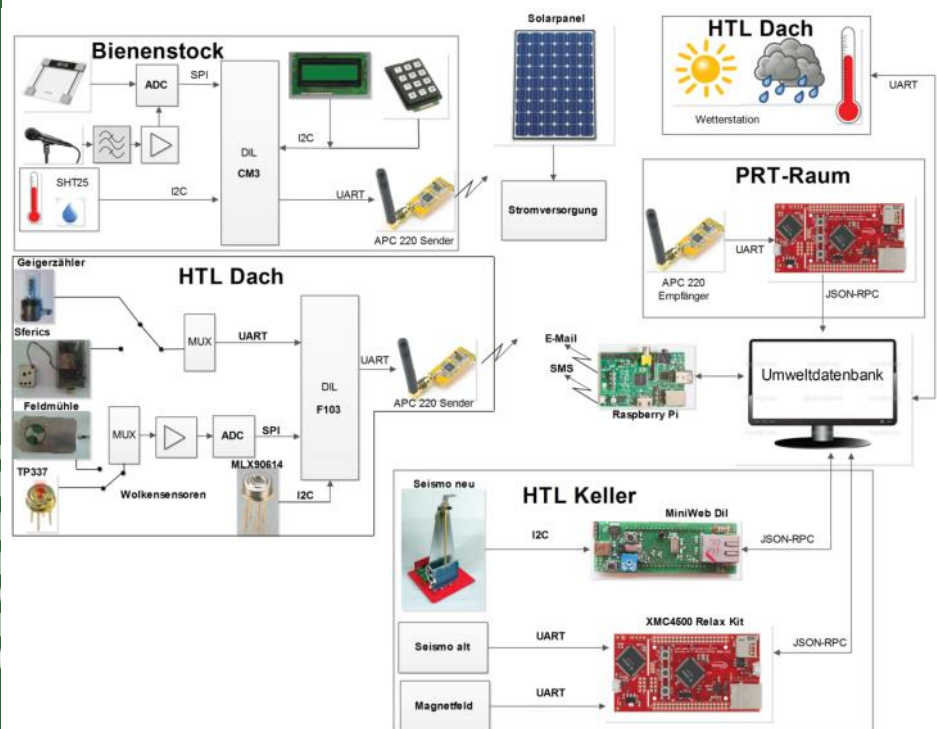


ermittelte Tageshöchsttemperatur das Verhältnis von Bienen zu verdeckelter Bienenbrut aus, was nach Arbeiten von Dr. Josef Bretschko eine hochsignifikante Schwarmerkennung ermöglicht. Zusätzlich soll erforscht werden ob es Zusammenhänge zwischen Schwarmneigung Magnetfeldschwankungen, Erdbeben oder dem Auftreten von Gewittern gibt. Das Frühwarngerät (gelbes Gehäuse am Tisch) war bereits 14 Tage im praktischen Einsatz und hat aufschlussreiche Ergebnisse geliefert.

Die Schüler wurden im Rahmen des Halbfinales (für Wien NÖ und Burgenland und der österrei-



Team : v.l.n.r Oliver Rieder, Andreas Riedinger (Bienenstockelektronik, Waage); David Fichtinger (Bienenbeute) Thomas Weis





### XMC4500 – Relax Kit

Infineon Evaluation Board

2 Pin Headers 2 x 20 mit:

- 4 x SPI-Master
- 3 x I2C
- 3 x I2S
- 4 x UART
- 2 x CAN
- 17 x ADC (12 bit),
- 2 x DAC
- 31 x PWM

Entwicklungsumgebung: DAVE3

### XMC4500 – Relax Kit

- Basiert auf 32-Bit ARM Cortex M4
- Systemtakt max. 120 MHz
- 512kB Flashmemory
- 128kB RAM
- MAC On-Chip



chischen Auslandsschule in Albanien) des Schülerwettbewerbs Jugend Innovativ in die HTL Mödling eingeladen, um ihre Arbeit der Öffentlichkeit zu präsentieren, und bekamen auch eine entsprechende Projektförderung zugesprochen. Zusätzlich erreichten Sie beim Bosch „Technik fürs Leben-Preis 2014“ in der Kategorie Energie- und Gebäudetechnik sowie Gebrauchsgüter den ersten Platz.

Wie auf dem Blockschaltbild zu erkennen ist, gibt es drei Messstationen. Direkt am Bienenstock wird das Gewicht des Stockes, Bienen Geräusche sowie die Innen- und Außentemperatur und Innen- und Außenluftfeuchtigkeit gemessen. Am Dach der HTL befindet sich eine Wetterstation und hier werden Radioaktivität, Gewitter, Bewölkungsgrad und Elektromagnetische Felder gemessen. Im Keller befinden sich zwei Seismografen (Nord/Süd, Ost/West) sowie ein dreiachsiges XYZ Messgerät für das Erdmagnetfeld. All diese Daten werden zur Umweltdatenbank gesendet. Ein Raspberry Pi wertet anschließend diese gesammelten Daten aus und benachrichtigt den Imker/ die Imkerin, falls ein Schwarmvorgang bevorsteht.

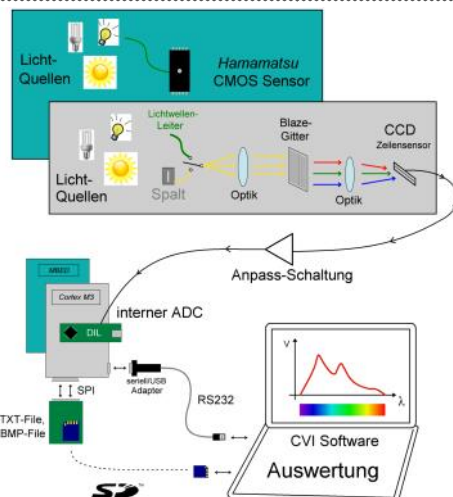
Als Mikrocontroller im solarversorgten Gerät kommt eine von uns neu entwickelte stromsparende Cortex M0 Platine, die in einen 28 poligen Standard DIL Sockel passt, zum Einsatz. Als Webserver, der die Daten über ein Funkmodul empfängt und über das JSON RPC Internetprotokoll zum Umweltdatenbankserver weiter schickt, kamen mehrere Infineon XMC4500 Relax Kits zum Einsatz.

## Maturaprojekt LowCost Spektrometer

Ein Spektrometer ist ein üblicherweise teures Messgerät, welches das Farbspektrum von Lichtquellen misst, mithilfe dessen auf die chemische Zusammensetzung der Lichtquelle zurückgeschlossen werden kann. Das Ziel der Diplomarbeit war es, eine LowCost-Variante eines solchen Messgerätes mit integriertem SD-Card-Interface sowie USB/RS232-Schnittstelle zu entwickeln. Mithilfe dieser Schnittstellen sollten die gemessenen Spektren verschiedener Lichtquellen als Bilddateien gespeichert werden. Um das Licht in seine Farbanteile aufzuspalten, wird ein optisches Gitter verwendet. Das zu untersuchende Licht wird durch einen dünnen Spalt geleitet, wodurch eine möglichst hochauflösende Darstellung des Spektrums erreicht wird. Nachdem das Licht in seine Farbanteile aufgespalten wurde, wird das Spektrum mithilfe eines CCD (*Charged Coupled Device*) Zeilen-Sensor gemessen. Um die kostengünstigste Variante zu realisieren, wurden zwei verschiedene Aufbauten untersucht: einerseits ein optischer Aufbau „Marke Eigenbau“ und andererseits wurde ein zugekaufter CMOS-Sensor mit integriertem Optikaufbau (Hamamatsu C10988MA) verwendet. Um den CCD-Zeilensensor bzw. den CMOS-Sensor auszulesen und die analogen Werte zu digitalisieren, wurden Analog-Digital-Wandler (ADC) verschiedener Mikrocontroller untersucht und letztendlich jener des MBED (NXP LPC1768) verwendet. Es können Spektren sowohl mit dem Eigenaufbau als auch mit dem Hamamatsu-CMOS-Sensor mithilfe verschiedener Modi aufgenommen werden (Einzelmessung, kontinuierliche Messung, automatische Kalibrierung). Rohdaten dieser Spektren können entweder auf einer SD-Karte abgespeichert, oder mithilfe der USB/RS232-Schnittstelle übertragen werden. Weiters ist es möglich diese Rohdaten entweder direkt durch den Mikrocontroller mithilfe einer selbstprogrammierten Bitmap-Library in eine Bitmap-Grafik (Diagramm) zu verwandeln, oder mithilfe einer in C++ programmierten Auswertesoftware auf einem PC darzustellen und zu vermessen.



Roman Bauer und Paul Guber



## Ergebnis

Beide gemessenen Spektren weisen eine Intensitätsverteilung auf, die mit dem Spektrum aus dem Datenblatt übereinstimmt. Lediglich eine kleine Abweichung im Bereich von 600nm kann festgestellt werden.

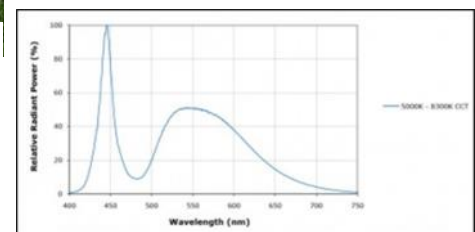
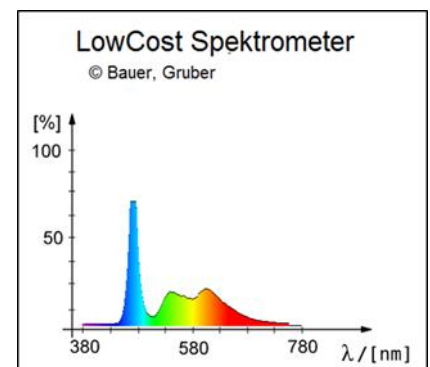
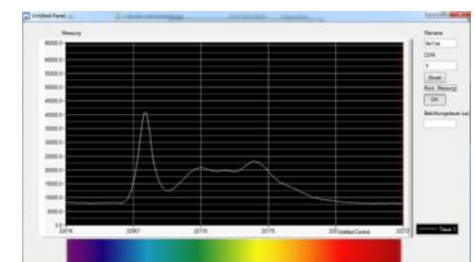
Die ausgezeichnete Qualität dieser Arbeit wurde durch die Einladung zum „agilent xtest award“ Finale (die zehn besten Teams) im Museumssaal des WUK in Wien Währung bestätigt.

Es wurde ein Spektrum einer weißen Leuchtdiode aufgenommen und mit jenem Spektrum aus dem Datenblatt verglichen.

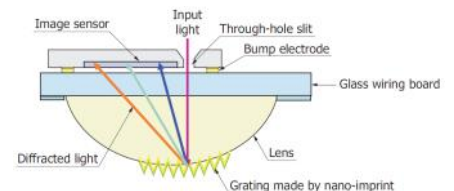
Messung LED weiß - PC-Auswertesoftware

Messung LED - BMP auf µC-SD-Karte

Spektrum LED weiß – Datenblatt



## Hamamatsu



## Projekt: Getreidedosieranlage / Home Automation

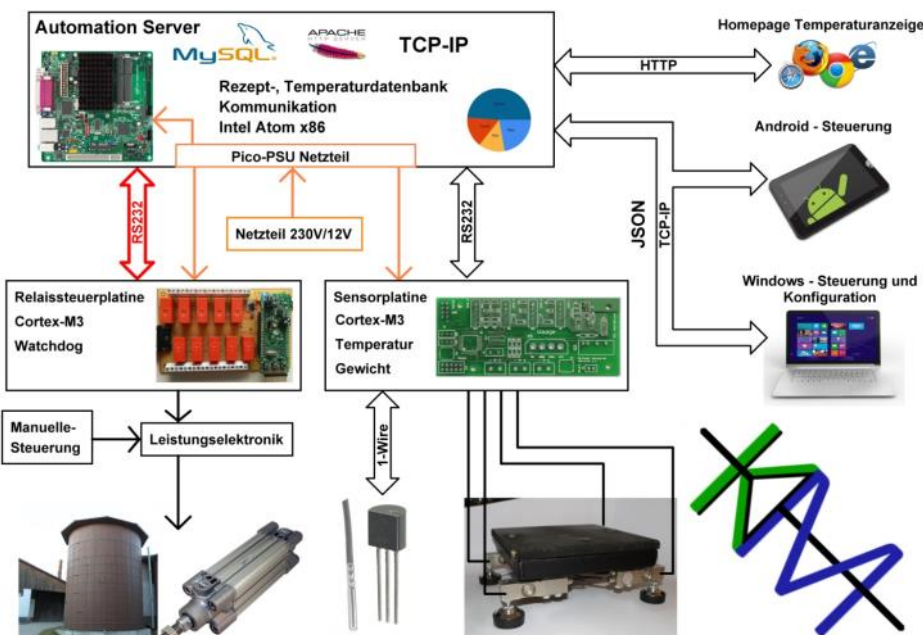
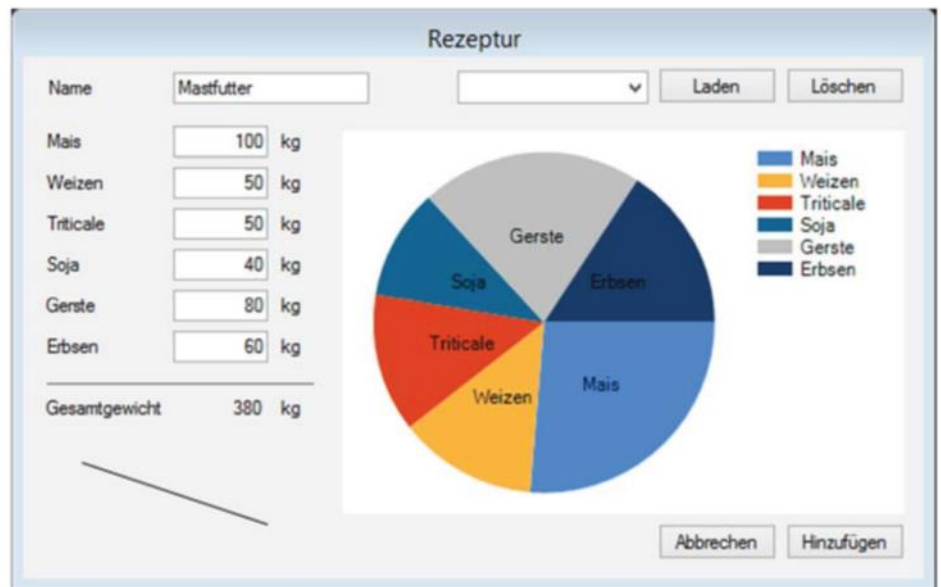
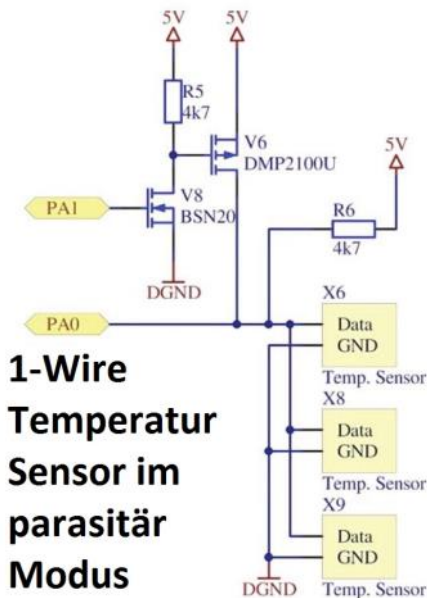
Als Test-Einsatzgebiet für die Dosieranlage dient die zu automatisierende Futterzubereitung eines Schweinemastzuchtbetriebes im Waldviertel. Diese Dosieranlage besteht neben dem HMI zur Steuerung noch aus einer neu zu entwickelnden elektronischen Waage, einer Relaissteuerplatine und einem bauseitig vorhandenen Schaltschrank mit der Leistungselektronik zur Ansteuerung von Förderschnecke und Druckluftschieber und einer Handsteuerung mit Not-Aus Abschaltvorrichtung. Das *Human Machine Interface* besteht aus einer Cortex A5 CPU. Auf dem HMI kommt Android als Betriebssystem zum Einsatz. Es verfügt über USB 2.0, RS232, CAN, GBit-LAN und einen 4,3"-Touchscreen. Für dieses System wurde eine Leiterplatte angefertigt und auch eine User-Interface-App für die Dosieranlage geschrieben, welche auch auf einem Tablet oder Smartphone verwendet werden kann. Die Waage für die Dosieranlage besteht aus 4 Wiegezellen an jeder Ecke (maximal Last = 2000kg). Die Auswerterschaltung besteht aus einem sehr rauscharmen Instrumentenverstärker, einem hochauflösenden 24-Bit-ADC und



Michael Weisgrab und Kevin Hofbauer

einem Cortex-M3 Mikro-Controller und ermöglicht mit einer Auflösung von 160g Getreide zu dosieren. Die Verbindung zum Server ist über RS232 realisiert. Weiters ermöglicht die Platine eine Messung der Getreidetemperatur über mehrere 1-Wire-Sensoren (bis zu 100 Meter Entfernung) in den Silos. Der Server basiert auf einem stromsparenden Intel Atom System. Er übernimmt die Messung des Gewichtes, der

Temperaturen und die Steuerung der Getreidezubereitung. Zur Speicherung der Rezepte und Temperaturen wird eine MySQL-Datenbank eingesetzt. Steuerungssoftware und Verwaltungsprogramm wurden in C# realisiert. Als Schnittstelle für das User Interface wird eine TCP Verbindung eingesetzt. Die Ausführung der Getreidezubereitung erfolgt über eine Relaissteuerplatine, welche ebenfalls über RS232 mit dem Server kommuniziert um die Sicherheit zu gewährleisten wurden Watchdogs und Relais mit Rückmeldekontakt eingesetzt.





# DAVE3

## Zu Bee-Watch-Projekt

Seit 2012 ist Infineon auch mit Cortex M0 oder Cortex M4 auf dem Markt präsent und hat sein umfassendes Know-how in der Entwicklung von Mikrocontrollern für echtzeitkritische Anwendungen mit den Vorteilen eines Industriestandard-Kerns kombiniert. Das Ergebnis, die XMC-Mikrocontrollerfamilie (XMC = *Cross Market Mikrocontroller*), auf Basis der Cortex™-M-Cores von ARM®. Das XMC-Mikrocontroller-Portfolio reicht von Low-End-Produkten mit niedriger Pinzahl bis zu hochentwickelten Lösungen für industrielle Anwendungen. Modellübergreifend einsetzbare Peripheriesysteme und Entwicklungstools gewährleisten ein hohes Maß an Skalierbarkeit und Kompatibilität zwischen allen Mitgliedern der XMC-Familie. Sie ist speziell für Anwendungen in den Bereichen erneuerbare Energien, Fertigungs- und Gebäudeautomation, Transport, Logistik, Medizintechnik sowie Beleuchtung und Haushaltsgeräte konzipiert, und profitiert dabei von seiner schnellen *Embedded Flash*-Technologie, seinen hohen Qualitätsstandards, der langen Lebensdauer seiner Produkte, den hohen Temperaturbereichen bis zu 125 °C und nicht zuletzt von seiner Position als führender Anbieter von Automotive-Produkten

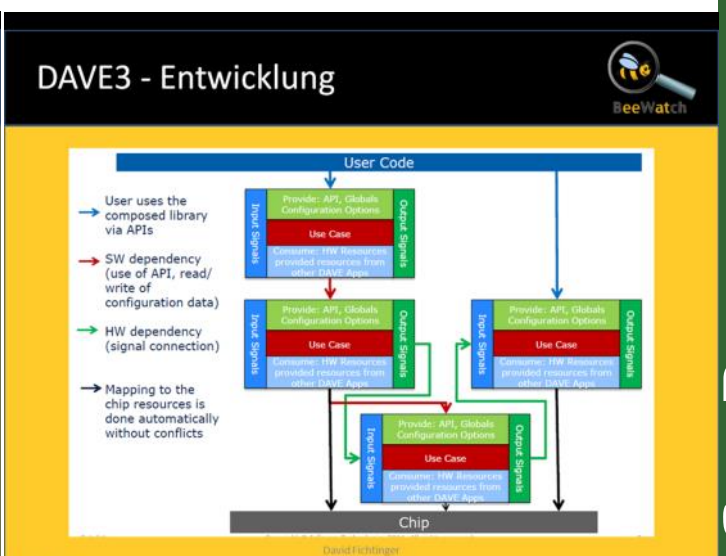
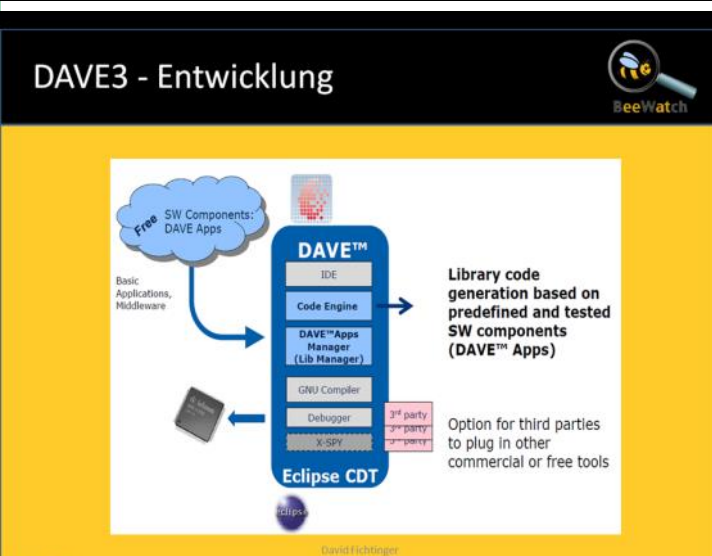
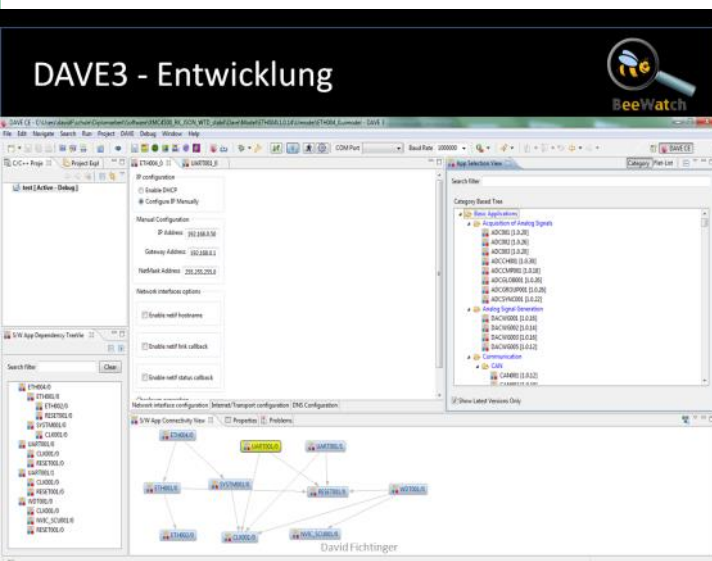
DAVE™, die zukunftsweisende Infineon-Plattform für alle XMC-Produkte, beschleunigt und vereinfacht die Entwicklungsarbeit und stellt Entwicklern eine leistungsfähige Mikrocontroller-Plattform für die unterschiedlichsten Anwendungen bereit

DAVE3 ist die von Infineon bereitgestellte IDE unter anderem für die XMC4500 Controller und basiert auf Eclipse. Neben den Standardfunktionen bietet DAVE3 auch die Verwendung von sogenannten Apps an. Diese Apps sind grafisch konfigurierbare Codegeneratoren, mit denen schnell und effizient die Hardware des XMC4500 konfiguriert werden kann. Die

sogenannten Apps sind ein wesentliches Merkmal der DAVE3 IDE. Die Apps ermöglichen es dem Bediener, Teile der Hardware oder komplette Funktionen grafisch und nach Parametern zu konfigurieren. Der dazu gehörige Code wird dann automatisch generiert und der Benutzer kann über Handler und mit zur Verfügung gestellten Funktionen, mit der zuvor mittels der Apps konfigurierten Hardware, kommunizieren. Der wesentliche Vorteil dieser neuen Art der Programmierung ist es, dass der Benutzer sehr schnell und effizient programmieren kann. Des Weiteren sind die Apps sehr gut dokumentiert und das Konfigurationsmenü ist übersichtlich gestaltet. Nachdem man eine App aus der App Selection ausgewählt hat, kann man die Hardware über ein grafisches Fenster konfigurieren. Wie an dem folgenden Beispiel zu sehen ist, können verschiedene Parameter eingestellt werden und an die Forderungen des Benutzers angepasst werden. Gegenüber der Code-Ansicht bietet DAVE3 noch die Möglichkeit der S/W App Connectivity Ansicht und die der H/W Connectivity Ansicht. Diese Ansichten zeigen die verwendeten Apps und die dazugehörigen Signale. Nachdem die Apps generiert wurden, kann man über Handler mit diesen Apps kommunizieren

<http://www.infineon.com/cms/de/product/microcontroller/>

**Besonderer Dank gebührt Ing Wilhelm Brezovits von Infineon der uns die XMC4500 Boards gesponsert hat, und der sich die Zeit genommen hat, uns den DAVE3 zu erklären.**



# K&K—oder Karma und Kismet

Nachlese eines Clubabends vom 20.3.2014

Thomas Sulak

## Die Ausgangssituation

Sie sitzen in ihrem Lieblings Café, verbinden sich mit einem kostenlosen Hotspot und jemand mit etwas technischem Wissen und einer guten Portion krimineller Energie liest all ihre persönlichen Daten aus. Zugegeben – man könnte jetzt sagen: *selbst schuld, wer sich mit einem kostenlosen WLAN verbindet!* Was aber, wenn ihr Smartphone das automatisch, ohne ihr Zutun, in der Hostentasche macht? Gibt's nicht?

## Das Szenario

Findet er ein kostenloses, offenes WLAN und stellt eine Verbindung her, denkt der gemeine User oft ein Schnäppchen gefunden zu haben. Tatsächlich surfen sie möglicherweise über einen Hotspot, den der Typ neben ihnen gerade eingerichtet hat, und der schaut ihnen vielleicht sogar noch über die Schulter bei Ihren Banküberweisungen.

Kostenlose Hotspots stellen in der Öffentlichkeit Internet zur Verfügung für Jedermann, jeder Orts. Diese stellen sich aber immer wieder als Mogelpackung heraus. Als Kostenfalle. Es ist unvorstellbar, wie viele Hotspots man auf öffentlichen Plätzen findet - offen! – unverschlüsselt - mit einem einzigen Ziel: Sie und Ihr Smartphone „abzuzocken“. Nun ja – vielleicht ist es nicht ganz so schlimm, sie sollten jedoch davon ausgehen.

Ihnen passiert das nicht? Erlauben sie mir eine Frage: verbindet sich ihr Smartphone automatisch mit dem WLAN im Büro oder zu Hause? Ja? Praktisch und bequem! – Keine Frage. Aber damit haben Sie bereits den Opferstatus!

Es ist sehr einfach, einen eigenen Hotspot einzurichten und diesen z.B. „Café Free Web“ zu benennen. Das kann heute jeder Smartphone User selbst. Will man sich etwas professioneller aufstellen, nimmt man einen dafür eigens konzipierten Router – einen sogenannten „Ja-Sager“.

## Ja-Sager und KARMA

„Karma attacks Radio machines automatically.“ Radio machines sind einfach gesagt alle Geräte, die mit WiFi ausgerüstet sind. Das kann ein Desktop, Notebook, Smartphone, Tablet oder sogar eine Xbox oder Playstation sein.

Wie funktioniert KARMA? Jedes Mal, wenn sich Ihr Smartphone mit einem Hotspot verbindet, speichert es den Namen (SSID) und das Passwort. Somit verbindet es sich in Zukunft automatisch. Praktisch und bequem!

Was jedoch die wenigsten Benutzer wissen ist – egal wo sie sind, das Smartphone fragt ständig nach, ob eines der vertrauten Netzwerke in der Nähe ist; eine der Hauptursachen für einen erhöhten Stromverbrauch.

Dazu sendet das Gerät einen sogenannte Probe Request (Abb.1). Ein Probe Request ist die permanente Nachfrage, ob ein vertrautes Netzwerk in der Nähe ist, und falls ja, verbindet es sich.

KARMA analysiert diese Anfragen und extrahiert die Netzwerk Identifikationsdaten wie Netzwerkname und MAC Adresse und benutzt diese Informationen um einen Klon dieses Netzwerkes zu kreieren. Es verhält sich an dieser Stelle wie ihr vertrautes Netzwerk und antwortet auf diese Probe Request mit „Ja, ich bin Dein vertrautes WLAN „zu Hause“ – verbinde dich“.

Nachdem Ihr Smartphone nicht zwischen dem echten und dem geklonten Netzwerk unterscheiden kann, verbindet es sich (Abb.2).

Sobald die Verbindung hergestellt ist, beginnt das Smartphone auch schon zu kommunizieren und ruft E-Mails ab, verbindet sich mit Facebook, usw. – das alles, über den scheinbar kostenlosen Hotspot, in ihrer Tasche ohne ihr Wissen und ohne ihr Zutun.

So kann es schon mal vorkommen, dass sie am Stephansplatz mit dem WLAN zu Hause verbunden sind. Was technisch nicht möglich ist, sofern Ihre Adresse nicht 1010 Wien, Stephansplatz 1 lautet. Diese Verbindung kommt in jedem Fall zu Stande, auch wenn das eigentliche Netzwerk verschlüsselt ist. Das Smartphone fragt an dieser Stelle nicht nach.

Mit diversen Tools wie Wireshark kann nun der komplette Netzwerkverkehr mitgelesen werden. All ihre Passwörter, Login Daten, ihr komplettes Surfverhalten. Es werden manipulierte Login Seiten zur Verfügung gestellt – von diesen Fakes werden dann Benutzernamen und Passwörter abgegriffen.

## Tipp

Live Demos dieser Hacking Attacken finden immer wieder im Zuge diverser Veranstaltungen von Thomas Sulak statt. Infos und Anmeldung finden Sie unter [www.thomas-sulak.at](http://www.thomas-sulak.at)

## Conclusio

Noch immer gehen viele Smartphone User viel zu sorglos mit Ihren Datenverbindungen um! Im Inland wie im Ausland. Das Umdenken vom „Handy“ zum „Smartphone“ hat in vielen Köpfen noch nicht stattgefunden und wird viele Menschen noch sehr viel Geld kosten.

Ein Test in der Öffentlichkeit zeigt, dass sich zahlreiche Geräte bereits nach einigen Minuten erfolgreich mit dem Router verbunden haben (Abb. 3).

## In der Praxis

In der Praxis würde man vermutlich folgender Maßen vorgehen:

- Ja-Sager aktivieren und das vorhandene WLAN Klonen (Starbucks, McDonalds, usw.)

- Mittels einer DeAuth-Attacke, die man zum Beispiel über ein Smartphone ausführen kann, werden sämtliche Smartphones vom aktuellen WLAN getrennt und verbinden sich neu. Jetzt aber mit dem geklonten WLAN.

- Über einen manipulierten DNS Server, der am Router installiert ist, werden die ahnungslosen Benutzer nun auf falsche Webseiten umgeleitet, wie zum Beispiel: Facebook, wo sie ihre freundlicherweise protokolliert.

Nachdem nun der gesamte Netzwerkverkehr über den Ja-Sager läuft, kann dieser sehr leicht mittels Wireshark ausgelesen und protokolliert werden.

## Fazit

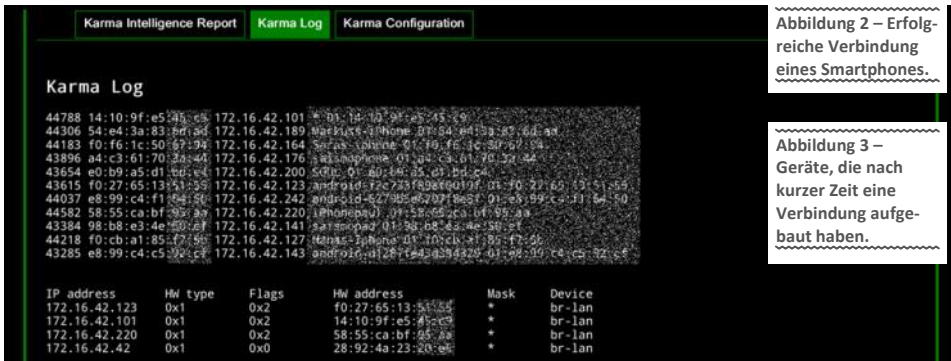
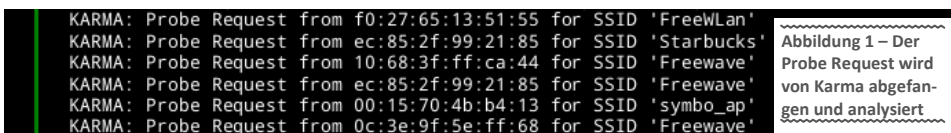
Der Datenaustausch über WLAN und Bluetooth ist sehr mangelhaft gesichert und lädt Spione ein. Darum schalten's diese Funktionen nur dann ein, wenn Sie unmittelbar benötigt werden. Ein angenehmer Nebeneffekt - stark reduzierter Akku-Verbrauch!

Ich stelle mir ohnehin die ernstgemeinte Frage, wozu ich WLAN im Inland, in der Öffentlichkeit auf meinem Smartphone jemals aktivieren sollte? Der Schalter um diese Funktion zu deaktivieren ist mittlerweile sowohl bei iOS als auch bei Android sehr einfach über Wischen zu erreichen. Selbiges gilt natürlich auch für Bluetooth – abdrehen!

Moderne Mobilfunk Verträge beinhalten mindestens 1-GB-Download-Volumen. Das ist bei weitem ausreichend für www, E-Mails, Facebook und Co. Wer, so wie ich, viel Musik online streamt, kann für ein paar Euro pro Monat das Downloadvolumen auf 4GB oder mehr erhöhen.

## Wie schützt man sich?

1. WLAN und Bluetooth abschalten, wenn diese nicht benötigt werden.
2. Kein automatisches Verbinden zulassen.
3. Login ausschließlich über „https“ – Seiten.
4. kostenloses WLAN nur mit Registrierung. Stichwort: McDonalds oder Starbucks.
5. Steganos Online Shield® für ein paar Euro pro Jahr.





# Zum Zusammenhang von Motivation und E-Learning

Matthias Hütthaler

Dass die Einstellung zu einem erlernenden Thema und die daraus resultierende Motivation beim Lernenden die Grundlage jeden Lernprozesses bildet und somit das Lernen beeinflusst, ist in der Kognitionswissenschaft hinreichend bewiesen. Fehlende Motivation führt sehr häufig zu deutlich verringerter kognitiver Leistungsfähigkeit. Deshalb scheint eine positive Wahrnehmung des Lernstoffes notwendig um die Basis eines wirksamen Lernprozesses zu bilden. Es werfen sich nun die Fragen auf, ob durch E-Learning-Maßnahmen die Motivation beim Lernenden positiv beeinflusst werden kann und ob es einen Zusammenhang zwischen Motivation und E-Learning gibt.

Zieht man einschlägige Fachliteratur (z.B. **Hilzensauer** 2008, **Holzinger** 2001, **Ehlers** 2005) zur Beantwortung zu Rate, erkennt man relativ rasch, dass es einige Argumente gibt, die dafür sprechen, dass E-Learning die Motivation steigert. Beispielsweise kann das Lernen durch den Einsatz verschiedener digitaler Medien wie kurze Videosequenzen abwechslungsreicher werden. Oftmals ist schon alleine durch den Wechsel der Lehr- oder Lernmethode ein positiver Motivationseffekt zu erwarten. Möchte man dieses Argument kritisch betrachten, kann man sagen, dass der Unterricht durch ein jedes Medium abwechslungsreicher gemacht werden kann und man dazu keine konkreten Umsetzungen von E-Learning-Maßnahmen bzw. digitale Medien braucht.

Für manche Autoren (**Hilzensauer** 2008, **Wiedenmann** 2002) wirkt sich vor allem das Ansprechen von technischer Affinität positiv auf die Motivation aus. Wenn man die aktuellen Entwicklungen im Bereich Internet oder die von sozialen Netzwerken betrachtet, lässt sich durchaus vermuten, dass besonders bei jungen Lernenden ein gewisser Motivationseffekt durch das Einbinden von neuen Medien einhergeht.

Als einer der wichtigsten Vorzüge des Lernens mit neuen Medien im Vergleich zum Lernen mit traditionellen Medien wird häufig die große Zeit- und Ortsunabhängigkeit genannt (**Welsh** 2003, **Hudetz** 2003). E-Learning ermöglicht somit eine

flexible Lernorganisation. Lehrende und Lernende müssen sich nicht mehr zur gleichen Zeit am gleichen Ort treffen. Man kann alleine zu Hause sitzen und muss eigenverantwortlich dafür sorgen, dass die Inhalte verstanden und gelernt bzw. die Aufgaben durchgearbeitet werden. Für jene, die die Reife der Eigenverantwortlichkeit haben, ermöglicht E-Learning somit eine flexible Lernorganisation. Heute weiß man (**Winkelbauer o. J.**), dass sich die meisten Menschen beim Lernen zu Hause wohler fühlen, was einen Anstieg der Motivation zur Folge hat.

Die Annahme, dass E-Learning die Motivation steigert, ist allerdings nicht unumstritten und wird auch kritisch gesehen. Dieses Argument wird von manchen Experten sogar als Mythos bezeichnet (**Holzinger** 2001, **Rietsch** 2003). Die Vermutung, alleine die Verwendung neuer Technologien sei motivierend, wird mit dem Konzept der Neugiermotivation erklärt, kann jedoch kaum als dauerhaft angesehen werden. Durch die Einführung eines digitalen Mediums ist zwar ein kurzzeitiger Motivationsanstieg beim Lernenden erkennbar. Dieser lässt allerdings nach einem bestimmten Zeitraum bis zur Gänze nach.

Wie man im Vorangegangenen erkennen kann, gibt es eine Reihe von Argumenten die für einen positiven Motivationsanstieg sprechen. Um diese Motivation auf Dauer aufrecht zu erhalten, scheint vor allem die Gestaltung der konkreten Form der Umsetzung von E-Learning (z.B. bei einem Kurs in einer Lernplattform) ausschlaggebend und einer der wichtigsten Erfolgsfaktoren beim E-Learning zu sein. In diesem Sinne können digitale Medien nur mithilfe eines didaktischen Konzepts zielführend eingesetzt werden.

**Ehlers, U.** (2005). Was wissen wir über den E-Lerner? – Zum Stand der Qualitätsforschung aus Nutzersicht. In Balli, C. (Hrsg.), E-Learning – wer bestimmt die Qualität (S. 10-12). Bielefeld: Bertelsmann.

**Hilzensauer, W., Attwell, G., Chrzaszcz, A., Buchberger, G., Hornung-Prähauser, V. & Pallister, J.** (2008). Neue Kompetenzen für E-Portfolio-Begleiter/innen? – Der Kurs MOSEP – More Self-Esteem with my E-Portfolio. In Zauchner, S., Baumgartner, P., Blaschitz, E. & Weissenböck, A. (Hrsg.), Offener Bildungsraum Hochschule – Freiheiten und Notwendigkeiten (S. 103–112). Münster/New York/München/Berlin: Waxmann.

**Holzinger, A.** (2001). Basiswissen Multimedia. Würzburg: Vogel.

**Hudetz, K.** (2003). E-Learning als methodisch-didaktischer Ansatz für die Gestaltung und Nutzung neuer Unternehmenskonzepte in KMU des Handels am Beispiel von E-Commerce. In J. Kutscha (Hrsg.), E-Learning – Die Anwender bestimmen die Qualität – Analysen und Konzepte für die Integration von E-Learning in Geschäftsprozesse kleiner und mittelständischer Handelsbetriebe (S. 51–77). Bielefeld: Bertelsmann.

**Rietsch, P.** (2003). Erfolgsfaktor Multimedia-Didaktik – Drei Beispiele. In Dittler, U. (Hrsg.), E-Learning – Einsatzkonzepte und Erfolgsfaktoren des Lernens mit interaktiven Medien (S. 77–91). München/Wien: Oldenbourg Verlag.

**Winkelbauer, W.** (o.J.). Lernen von zu Hause via Web. <http://www.wissenswertes.at/index.php?id=e-learning> (11. Jänner 2014)

Welsh, E.T., Wanberg, C.R., Brown, K.G. & Simmering, M.J. (2003). E-learning: emerging uses, empirical results and future directions. International Journal of Training and Development, 7. <http://onlinelibrary.wiley.com/doi/10.1046/j.1360-3736.2003.00184.x/abstract> (20. März 2013)

## Das Recht auf Vergessen...

Werner Illsinger

Im Mai hat der Europäische Gerichtshof nach einer Klage eines Spaniers – dessen Haus 1998 zwangsversteigert wurde – entschieden, dass jeder EU Bürger das Recht darauf hat, dass solche Informationen über Google nach einiger Zeit nicht mehr gefunden werden können. Die Suchmaschine Google hatte die Information über die Zwangsversteigerung im Online Archiv einer Zeitung indiziert.

Ich halte das Urteil, dass jeder Mensch ein Recht darauf hat, dass "Dummheiten" aus der Vergangenheit ihn nicht ewig verfolgen für sinnvoll. In den meisten Fällen ist das derzeit der Fall. So verjähren z.B. Vorstrafen nach einiger Zeit ebenfalls. So sollte ein Mensch auch das Recht haben

dass das Netz kompromittierende Dinge über ihn vergisst.

Den Zugang das über die Entfernung aus dem Index einer Suchmaschine zu tun, halte ich aber für falsch. Das zeigt auch die Umsetzung. Google entfernt die Suchergebnisse zwar aus den Europäischen Suchmaschinen (z.B. wenn ich den Inhalt über google.at suche) – nicht aber über die außerhalb der EU befindlichen. Wenn ich also den Namen des armen Spaniers auf der google.com Seite suche, dann finde ich den Zeitungsbericht über die Zwangsversteigerung noch immer.

Wenn es also ein Recht auf vergessen geben sollte, dann müssten solche Informationen an der Quelle gelöscht werden. Die Zeitung muss-

te den entsprechenden Artikel im Online Archiv entweder entfernen oder besser anonymisieren. Dann würde jede Suchmaschine der Welt – es gibt ja auch nicht nur Google – die Inhalte nicht mehr indizieren können – und der arme Kerl würde auch nicht mehr in den Suchtreffern aufscheinen.

Dann wäre er wirklich vergessen, der arme Spanier. So ist er ein Zombie – der noch immer durch die Suchmaschinenwelt geistert. Die Idee ist gut – die Umsetzung ist schlecht. Das war übrigens auch die Meinung des Generalanwalts des EuGH – dessen Empfehlung ist der Gerichtshof aber in diesem Fall unüblicher Weise nicht gefolgt.

# ClubComputer.at

ClubComputer.at ist einer der größten Computerclubs Österreich.

 Angebote	„wir begleiten unsere Mitglieder in die digitale Zukunft“	PCNEWS-Seite
	  Simmeringer Bier- und Kulturschmankerl 1110 Wien, Simmeringer Hauptstraße 152 <a href="http://www.kulturschmankerl.at">www.kulturschmankerl.at</a>	136-30 138-2 und 5
	Clubabend am ersten Dienstag und dritten Donnerstag, Stammtisch am zweiten Montag im Monat ab 18:00, Vortrag/Diskussion ab 19:00	
	<b>pcnews.at</b> 32 Seiten, 5x jährlich: Feb, Apr, Jun, Sep, Nov  	<b>2014:</b> 137, 138, 139, 140, 141
	Hier wird Dir geholfen <a href="http://www.clubcomputer.at/forum/forum.php">www.clubcomputer.at/forum/forum.php</a>	
	Preisnachlass bei Alternate, Computerkabel Kaminek, Conrad, Gerko, Metro  	135-5
	<b>Jahresveranstaltung Juni 2015</b>	138-31
	<a href="http://www.name.clubcomputer.at">www.name.clubcomputer.at</a> 500MB, 5 Mailboxen, 5 Aliases	126-15 138-6,8,18
	<b>name.at</b> 17,90 €, <b>name.com</b> 9,90 €, <b>name.computer.at</b> kostenlos	
	<b>name@clubcomputer.at</b> 500 MB/POP3/IMAP/Web-Interface	126-9
	30 GB Speicherplatz, betrieben in Österreich verwendbar von PC und Smartphone,	137-8
	Faxe ohne Faxgerät als PDF per E-Mail zugestellt bekommen.	
	<a href="http://www.facebook.com/clubcomputer">www.facebook.com/clubcomputer</a>	
	24/7: +43 1 6009933-11 FAX: +43 1 6009933-12	
	<a href="http://www.adim.at">www.adim.at</a>	138-4
	Siccardsburggasse 4/1/22 1100 Wien <a href="mailto:buero@clubcomputer.at">buero@clubcomputer.at</a> <a href="mailto:buchhaltung@clubcomputer.at">buchhaltung@clubcomputer.at</a> <a href="mailto:support@clubcomputer.at">support@clubcomputer.at</a>	
	<a href="http://billing.clubcomputer.at">billing.clubcomputer.at</a> -&gt; „Mitglied werden“	126-5

Foto	Mobile	Akademie	NEWS	Präsident	Marketing	DigitalHome	Linux	WebDesign
Andi	Pauli	Georgie	Franz	Werner	Ferdinand	Christian	Günter	Herbert
								

3,25 €  
pro Monat