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

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_	Ç	ü	é	â	ä	à	å	ç	ê	ë	è	ï	î	ì	Ä	Å
9_	É	æ	Æ	ô	ö	ò	û	ù	ÿ	Ö	Ü	¢	£	¥	₧	ƒ
a_	á	í	ó	ú	ñ	Ñ	ª	º	¿	⌐	¬	½	¼	¡	«	»
b_	░	▒	▓	│	┤	╡	╢	╖	╕	╣	║	╗	╝	╜	╛	┐
c_	└	┴	┬	├	─	┼	╞	╟	╚	╔	╩	╦	╠	═	╬	╧
d_	╨	╤	╥	╙	╘	╒	╓	╫	╪	┘	┌	█	▄	▌	▐	▀
e_	α	ß	Γ	π	Σ	σ	µ	τ	Φ	Θ	Ω	δ	∞	φ	ε	∩
f_	≡	±	≥	≤	⌠	⌡	÷	≈	°	∙	·	√	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_	€	•	‚	ƒ	„	…	†	‡	ˆ	‰	Š	‹	Œ	•	Ž	
9_		‘	’	“	”	•	–	—	˜	™	š	›	œ	•	ž	Ÿ
a_		¡	¢	£	¤	¥	¦	§	¨	©	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

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.