



openHPI-Kurs: Wie funktioniert das Internet?

Tutorial: Kodierung von Medien

Matthias Bauer
Hasso-Plattner-Institut
Universität Potsdam

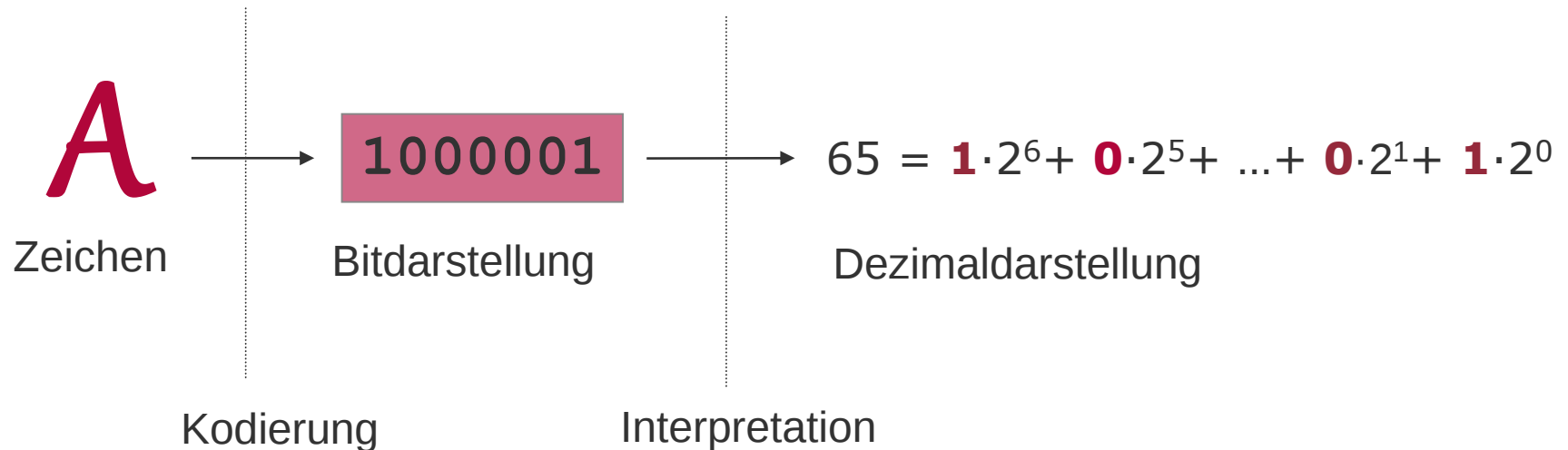
Warum Kodierung?

Im Internet werden immens viele Daten übertragen

- Um Informationen über das Internet versenden zu können, müssen diese binär kodiert sein
- Kodierung muss standardisiert sein, damit alle Beteiligten das Kodierte auch „verstehen“ können
- Viel mehr Daten in Form von Bildern, Audio und Video
→ noch mehr Speicherplatz- und Bandbreitebedarf
- Benutzer erwarten schnelle und zuverlässige Kommunikation

Kodierung von Zeichen (1/2)

Zeichen müssen in ein Format umgewandelt werden, mit dem Computer umgehen können



Einfaches Beispiel:

ASCII – American Standard Code for Information Interchange

Kodierung von Zeichen (2/2)

ASCII – 7-Bit-Kodierung mit 128 Zeichen:

ASCII value	Character	Control character	ASCII value	Character	ASCII value	Character	ASCII value	Character
000	(null)	NUL	032	(space)	064	@	096	a
001	☺	SOH	033	!	065	A	097	b
002	☹	STX	034	"	066	B	098	c
003	♥	ETX	035	#	067	C	099	d
004	♦	EOT	036	\$	068	D	100	e
005	♣	ENQ	037	%	069	E	101	f
006	♠	ACK	038	&	070	F	102	g
007	(beep)	BEL	039	'	071	G	103	h
008	■	BS	040	(	072	H	104	i
009	(tab)	HT	041	)	073	I	105	j
010	(line feed)	LF	042	*	074	J	106	k
011	(home)	VT	043	+	075	K	107	l
012	(form feed)	FF	044	,	076	L	108	m
013	(carriage return)	CR	045	-	077	M	109	n
014	♪	SO	046	.	078	N	110	o
015	☼	SI	047	/	079	O	111	p
016	▲	DLE	048	0	080	P	112	q
017	▼	DC1	049	1	081	Q	113	r
018	↕	DC2	050	2	082	R	114	s
019	!!	DC3	051	3	083	S	115	t
020	π	DC4	052	4	084	T	116	u
021	\$	NAK	053	5	085	U	117	v
022	▬	SYN	054	6	086	V	118	w
023	↕	ETB	055	7	087	W	119	x
024	↕	CAN	056	8	088	X	120	y
025	↕	EM	057	9	089	Y	121	z
026	↕	SUB	058	:	090	Z	122	{
027	←	ESC	059	;	091	[	123	
028	(cursor right)	FS	060	<	092	\	124	}
029	(cursor left)	GS	061	=	093	]	125	~
030	(cursor up)	RS	062	>	094	^	126	␣
031	(cursor down)	US	063	?	095	_	127	␣

Komprimierung von Dateien

Immer mehr Multimedia-Daten → Bedarf an Speicherplatz und Bandbreite

- Bilder, Audio und Video verbrauchen viel mehr Speicher als Text
 - müssen daher versuchen die Größe zu minimieren
- **Komprimierung** der Daten in ein standardisiertes Format ggf. unter Inkaufnahme von **Informationsverlust**



Grafikkodierung

Unkomprimierte Bitmaps

- Jedes Pixel wird einzeln gespeichert, hoher Speicherplatzbedarf
- BMP

Komprimierte Bitmaps

- Unnötige Information vermeiden (z.B. werden einzelne Pixel zu Linien oder Bereichen gleicher Farbe zusammengefasst)
- Einige Verfahren verlustbehaftet (JPEG); andere können verlustfrei benutzt werden (PNG, TIFF, z.T. GIF)

Vektorgrafik

- (Künstliche) Grafiken werden durch Vektoren (numerisch) beschrieben
- SVG

Audiokodierung

Unkomprimiertes Audio

- Sämtliche Audioinformation wird digital abgebildet (kodiert)
- WAV

Verlustbehaftet komprimiertes Audio

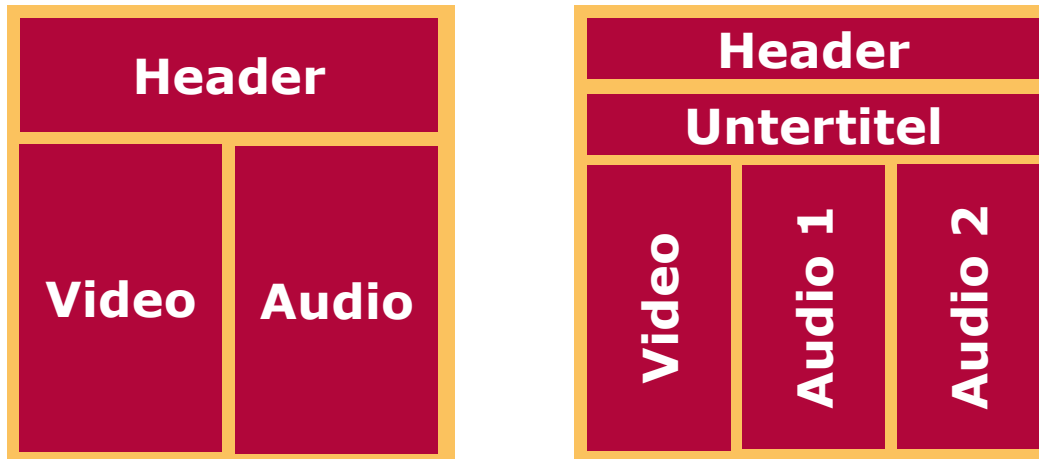
- Unnötige Information vermeiden (z.B. nicht hörbare Geräusche eliminieren, gleichartige Sequenzen zusammenfassen)
- MP3, AAC

Verlustfrei komprimiertes Audio

- (schwächere) Komprimierung von Geräuschen ohne Verlust von Information
- FLAC

Videokodierung

Typischerweise besteht ein Video (mindestens) aus einer **Video-** und einer **Audiospur**, die beide mit einem **Codec** komprimiert sind und in einem **Container** gespeichert werden



- **Container:** MP4, AVI, WebM, OGV, MKV, ...
- **Videocodecs:** H.265, H.264, Ogg Theora, WebM, ...
- **Audiocodecs:** MP3, AAC, Ogg Vorbis, ...