

I Unicode

Nicolas Seriot

October 24th, 2014



{{ softshake }}

<http://soft-shake.ch>

2014
@GENEVE



Marco Scheurer

@phink0

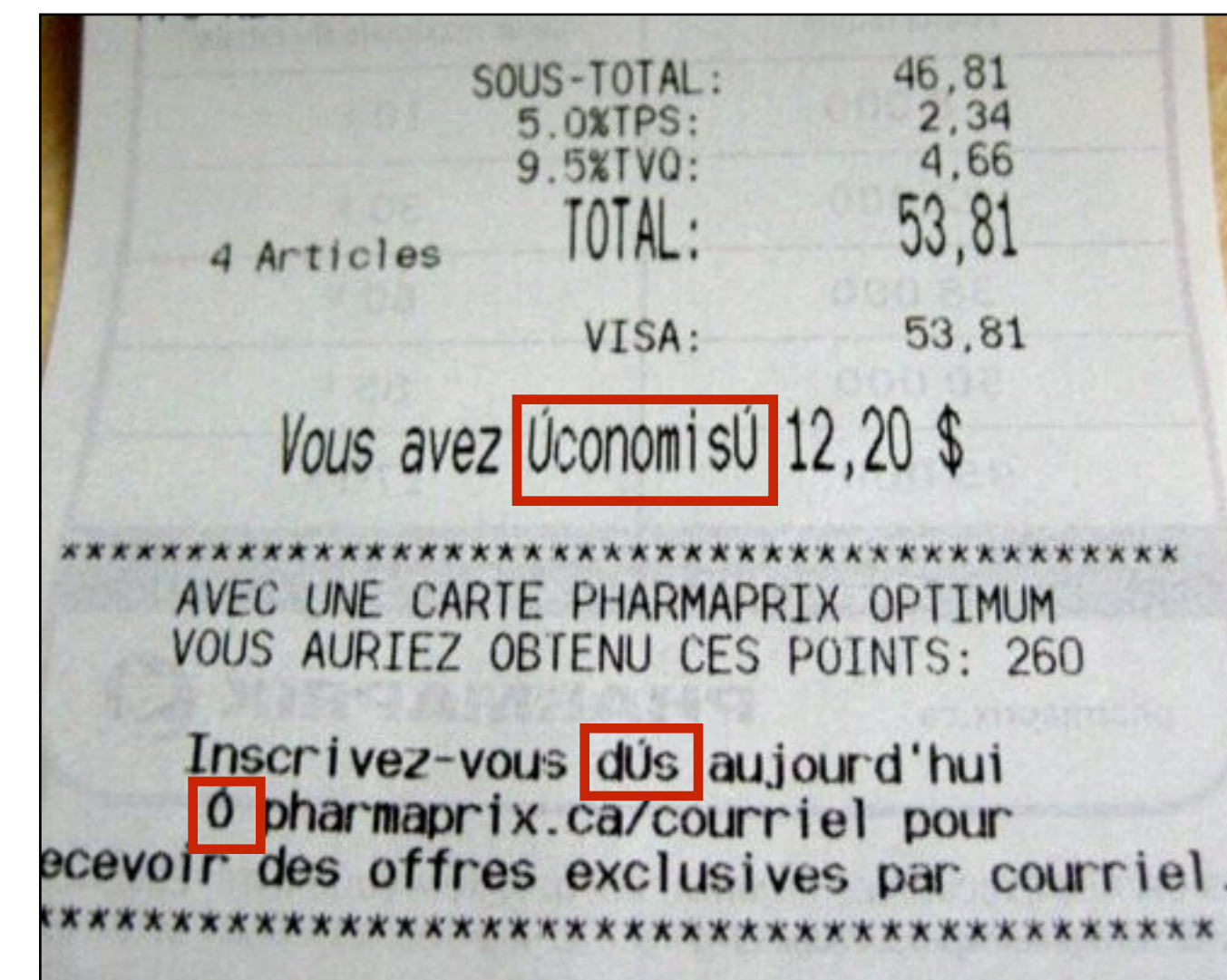
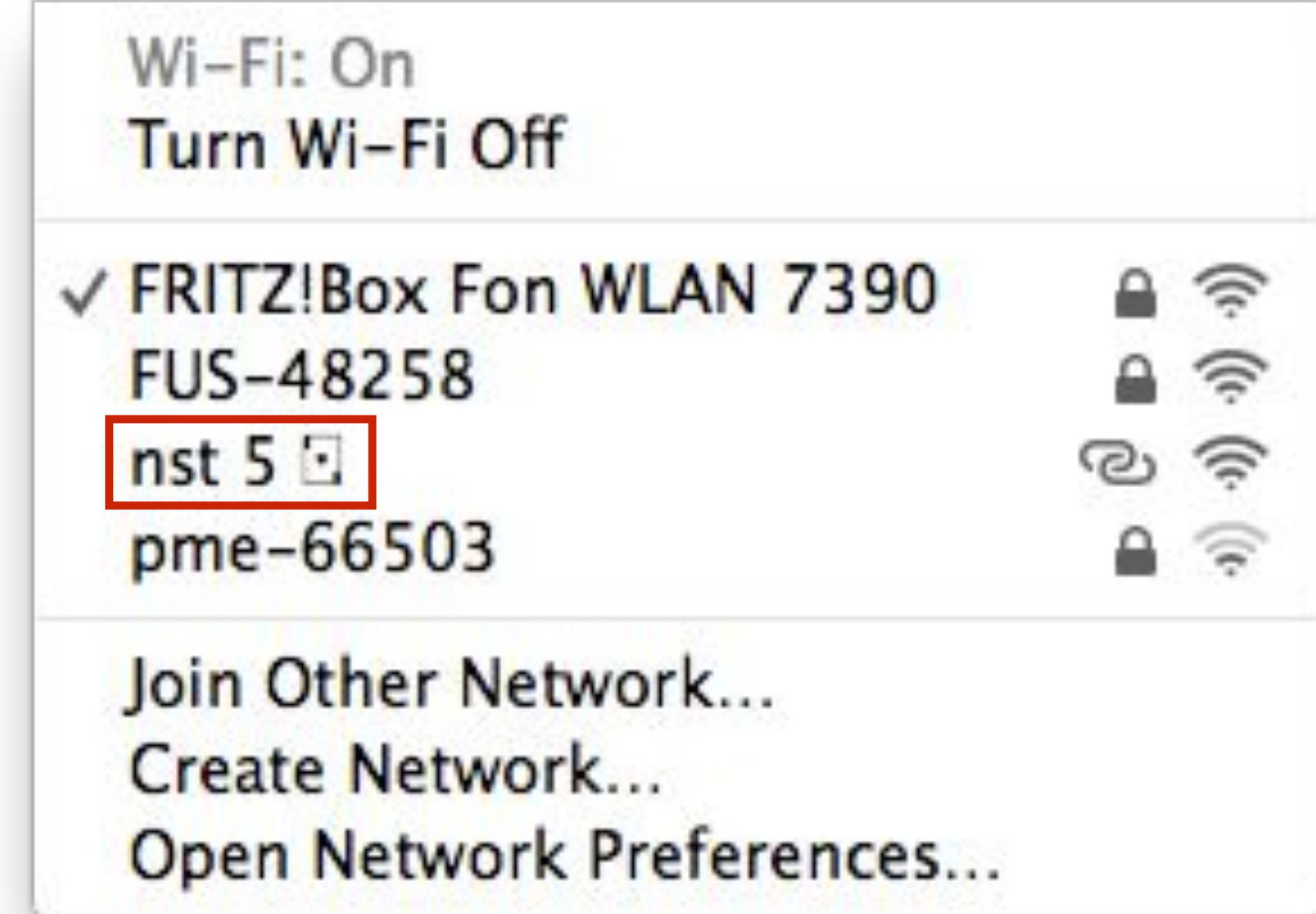
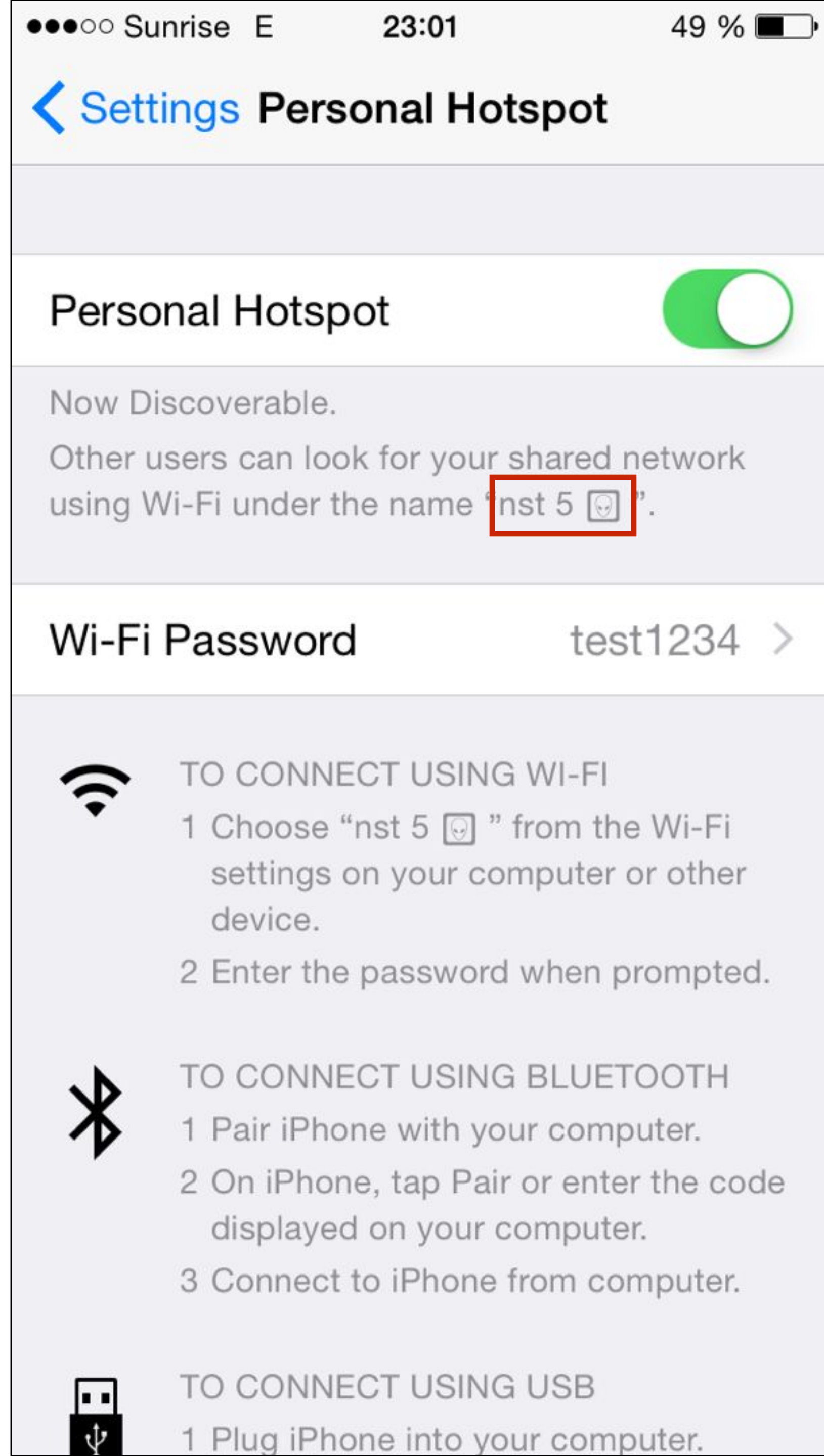


Following

2014. From Geneva Airport trains are running to **Genève** **Neuchâtel** and **Zürich**

← Reply ↻ Retweet ★ Favorited ... More

 Trains au départ de Genève-Aéroport					
Catégories	Heure	Destinations suisses			Voie
IR	12:53	Genève Nyon Lausanne	Brig		3
ICN	13:09	Genève Neuchâtel Biel/Bienne	Basel SBB		4
IR	13:23	Genève Nyon Lausanne	Brig		3
IR	13:53	Genève Nyon Lausanne	Brig		3
ICN	14:09	Genève Olten Zürich HB	St. Gallen		4
IR	14:23	Genève Nyon Lausanne	Brig		3
IR	14:53	Genève Nyon Lausanne	Brig		3
ICN	15:09	Genève Neuchâtel Biel/Bienne	Basel SBB		4
IR	15:23	Genève Nyon Lausanne	Brig		3
IR	15:53	Genève Nyon Lausanne	Brig		3
ICN	16:09	Genève Olten Zürich HB	St. Gallen		4
IR	16:23	Genève Nyon Lausanne	Sion		3
IR	16:53	Genève Nyon Lausanne	Brig		3



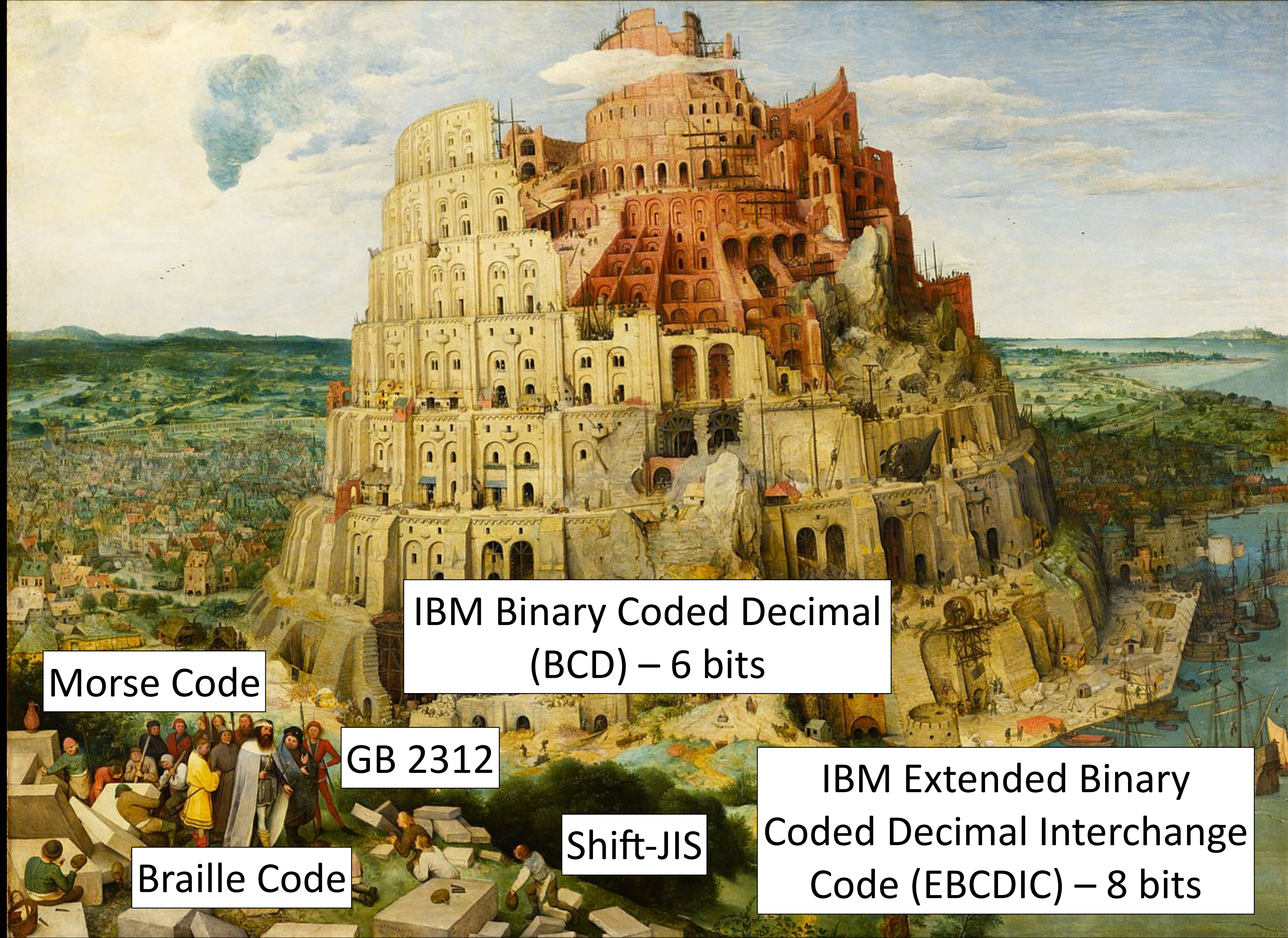
<http://unicode-wall-of-shame.com>

1. The Unicode Consortium

2. Selected Unicode Specifications

3. Unicode in Practice

4. Unicode Hacks



Morse Code

IBM Binary Coded Decimal
(BCD) – 6 bits

GB 2312

Braille Code

Shift-JIS

IBM Extended Binary
Coded Decimal Interchange
Code (EBCDIC) – 8 bits

1963: ASCII – 7 bits

(American Standard Code for Information Interchange)

	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	SPC	!	"	#	\$	%	&	'	()	*	+	,	-	.	/
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

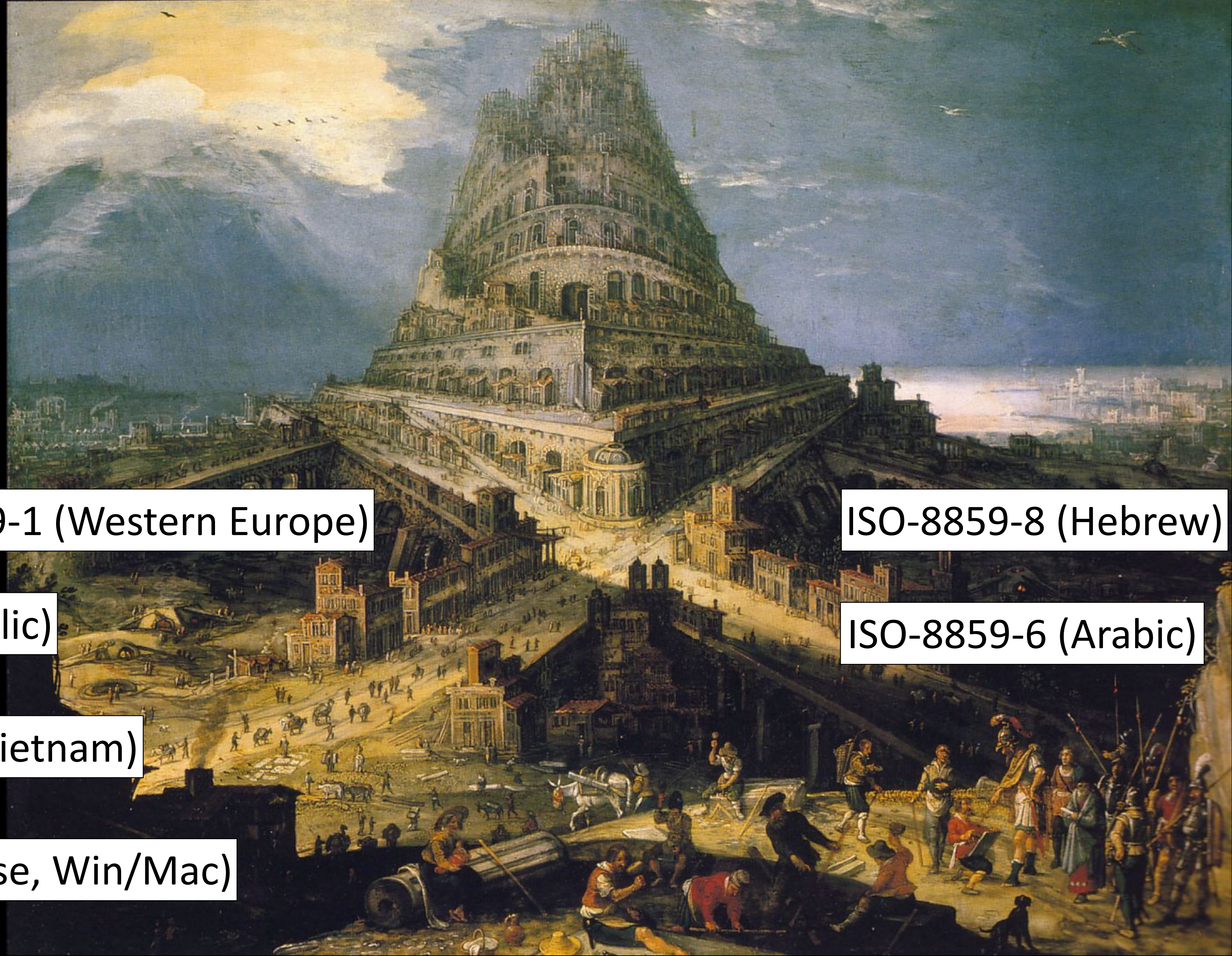
8 bits Encodings

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00	<u>NUL</u> 0000	<u>STX</u> 0001	<u>SOT</u> 0002	<u>ETX</u> 0003	<u>EOT</u> 0004	<u>ENQ</u> 0005	<u>ACK</u> 0006	<u>BEL</u> 0007	<u>BS</u> 0008	<u>HT</u> 0009	<u>LF</u> 000A	<u>VT</u> 000B	<u>FF</u> 000C	<u>CR</u> 000D	<u>SO</u> 000E	<u>SI</u> 000F
10	<u>DLE</u> 0010	<u>DC1</u> 0011	<u>DC2</u> 0012	<u>DC3</u> 0013	<u>DC4</u> 0014	<u>NAK</u> 0015	<u>SYN</u> 0016	<u>ETB</u> 0017	<u>CAN</u> 0018	<u>EM</u> 0019	<u>SUB</u> 001A	<u>ESC</u> 001B	<u>FS</u> 001C	<u>GS</u> 001D	<u>RS</u> 001E	<u>US</u> 001F
20	<u>SP</u> 0020	!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0 0030	1 0031	2 0032	3 0033	4 0034	5 0035	6 0036	7 0037	8 0038	9 0039	:	; 003B	< 003C	= 003D	> 003E	? 003F
40	@ 0040	A 0041	B 0042	C 0043	D 0044	E 0045	F 0046	G 0047	H 0048	I 0049	J 004A	K 004B	L 004C	M 004D	N 004E	O 004F
50	P 0050	Q 0051	R 0052	S 0053	T 0054	U 0055	V 0056	W 0057	X 0058	Y 0059	Z 005A	[005B	\ 005C] 005D	^ 005E	_ 005F
60	` 0060	a 0061	b 0062	c 0063	d 0064	e 0065	f 0066	g 0067	h 0068	i 0069	j 006A	k 006B	l 006C	m 006D	n 006E	o 006F
70	p 0070	q 0071	r 0072	s 0073	t 0074	u 0075	v 0076	w 0077	x 0078	y 0079	z 007A	{ 007B	 007C	} 007D	~ 007E	<u>DEL</u> 007F
80																
90																
A0	<u>MBSP</u> 00A0	ı 00A1	ç 00A2	£ 00A3	* 00A4	¥ 00A5	ı 00A6	§ 00A7	ˆ 00A8	@ 00A9	ª 00AA	« 00AB	¬ 00AC	— 00AD	® 00AE	— 00AF
B0	° 00B0	± 00B1	² 00B2	³ 00B3	´ 00B4	µ 00B5	¶ 00B6	· 00B7	¸ 00B8	¹ 00B9	º 00BA	» 00BB	¼ 00BC	½ 00BD	¾ 00BE	¿ 00BF
C0	À 00C0	Á 00C1	Â 00C2	Ã 00C3	Ä 00C4	Å 00C5	Æ 00C6	Ç 00C7	È 00C8	É 00C9	Ê 00CA	Ë 00CB	Ì 00CC	Í 00CD	Î 00CE	Ï 00CF
D0	Ð 00D0	Ñ 00D1	Ò 00D2	Ó 00D3	Ô 00D4	Õ 00D5	Ö 00D6	× 00D7	Ø 00D8	Ù 00D9	Ú 00DA	Û 00DB	Ü 00DC	Ý 00DD	Þ 00DE	ß 00DF
E0	à 00E0	á 00E1	â 00E2	ã 00E3	ä 00E4	å 00E5	æ 00E6	ç 00E7	è 00E8	é 00E9	ê 00EA	ë 00EB	ì 00EC	í 00ED	î 00EE	ï 00EF
F0	ø 00F0	ñ 00F1	ò 00F2	ó 00F3	ô 00F4	õ 00F5	ö 00F6	÷ 00F7	ø 00F8	ù 00F9	ú 00FA	û 00FB	ü 00FC	ý 00FD	þ 00FE	ÿ 00FF

ISO/IEC 8859-1 (Latin 1)

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00	<u>NUL</u> 0000	<u>STX</u> 0001	<u>SOT</u> 0002	<u>ETX</u> 0003	<u>EOT</u> 0004	<u>ENQ</u> 0005	<u>ACK</u> 0006	<u>BEL</u> 0007	<u>BS</u> 0008	<u>HT</u> 0009	<u>LF</u> 000A	<u>VT</u> 000B	<u>FF</u> 000C	<u>CR</u> 000D	<u>SO</u> 000E	<u>SI</u> 000F
10	<u>DLE</u> 0010	<u>DC1</u> 0011	<u>DC2</u> 0012	<u>DC3</u> 0013	<u>DC4</u> 0014	<u>NAK</u> 0015	<u>SYN</u> 0016	<u>ETB</u> 0017	<u>CAN</u> 0018	<u>EM</u> 0019	<u>SUB</u> 001A	<u>ESC</u> 001B	<u>FS</u> 001C	<u>GS</u> 001D	<u>RS</u> 001E	<u>US</u> 001F
20	<u>SP</u> 0020	!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0 0030	1 0031	2 0032	3 0033	4 0034	5 0035	6 0036	7 0037	8 0038	9 0039	:	; 003B	< 003C	= 003D	> 003E	? 003F
40	@ 0040	A 0041	B 0042	C 0043	D 0044	E 0045	F 0046	G 0047	H 0048	I 0049	J 004A	K 004B	L 004C	M 004D	N 004E	O 004F
50	P 0050	Q 0051	R 0052	S 0053	T 0054	U 0055	V 0056	W 0057	X 0058	Y 0059	Z 005A	[005B	\ 005C] 005D	^ 005E	_ 005F
60	` 0060	a 0061	b 0062	c 0063	d 0064	e 0065	f 0066	g 0067	h 0068	i 0069	j 006A	k 006B	l 006C	m 006D	n 006E	o 006F
70	p 0070	q 0071	r 0072	s 0073	t 0074	u 0075	v 0076	w 0077	x 0078	y 0079	z 007A	{ 007B	 007C	} 007D	~ 007E	<u>DEL</u> 007F
80																
90																
A0	<u>MBSP</u> 00A0	Ё 0401	Ђ 0402	Ѓ 0403	Є 0404	Ѕ 0405	І 0406	Ї 0407	Ј 0408	Љ 0409	Њ 040A	Ћ 040B	Ќ 040C	— 00AD	Ў 040E	Џ 040F
B0	А 0410	Б 0411	В 0412	Г 0413	Д 0414	Е 0415	Ж 0416	З 0417	И 0418	Й 0419	К 041A	Л 041B	М 041C	Н 041D	О 041E	П 041F
C0	Р 0420	С 0421	Т 0422	У 0423	Ф 0424	Х 0425	Ц 0426	Ч 0427	Ш 0428	Щ 0429	Ъ 042A	Ы 042B	Ь 042C	Э 042D	Ю 042E	Я 042F
D0	а 0430	б 0431	в 0432	г 0433	д 0434	е 0435	ж 0436	з 0437	и 0438	й 0439	к 043A	л 043B	м 043C	н 043D	о 043E	п 043F
E0	р 0440	с 0441	т 0442	у 0443	ф 0444	х 0445	ц 0446	ч 0447	ш 0448	щ 0449	ъ 044A	ы 044B	ь 044C	э 044D	ю 044E	я 044F
F0	№ 2116	ё 0451	ђ 0452	ѓ 0453	є 0454	ѕ 0455	і 0456	ї 0457	ј 0458	љ 0459	њ 045A	ќ 045B	ќ 045C	— 00A7	ў 045E	џ 045F

ISO/IEC 8859-5 (Cyrillic)



ISO-8859-1 (Western Europe)

ISO-8859-8 (Hebrew)

ISO-8859-5 (Cyrillic)

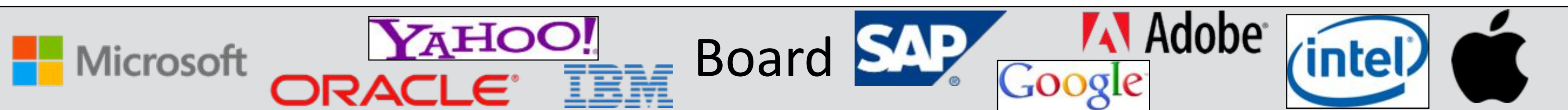
ISO-8859-6 (Arabic)

Windows-1258 (Vietnam)

SHIFT_JIS (Japanese, Win/Mac)



The Unicode Consortium



Board

Executive Officers

Technical Officers

Technical Committee Chairs

Staff

Technical Committee

- Unicode Standard
- Code Charts
- Unicode Character Database
- Standard Annexes

CLRD Technical Committee

- Unicode Locales Project
- Common Locale Data Repository

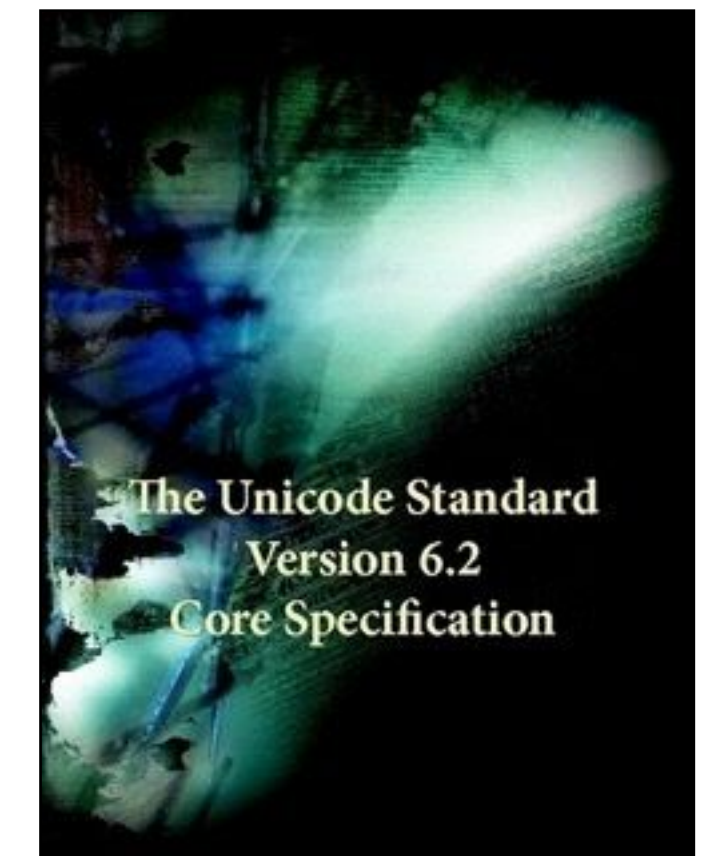
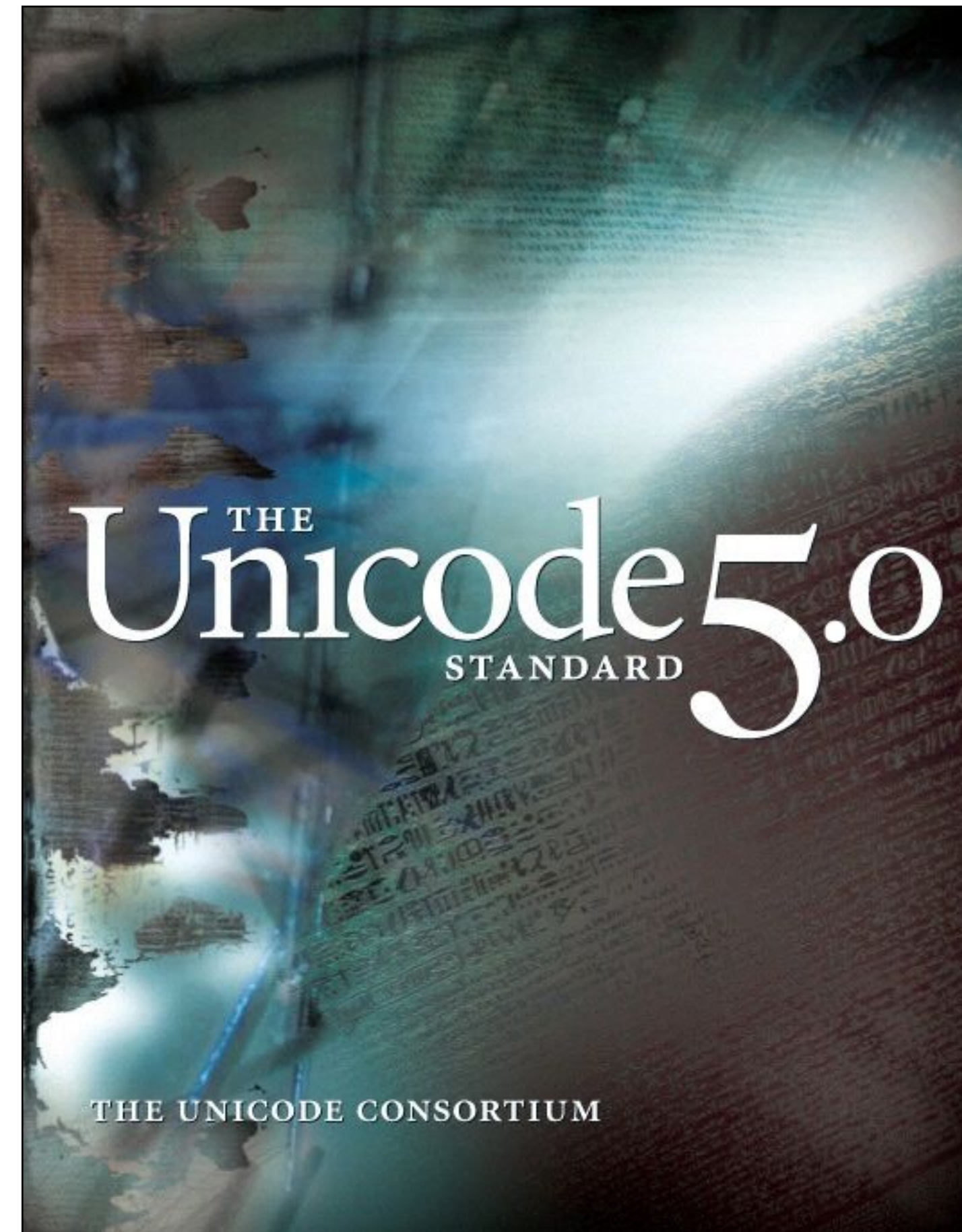
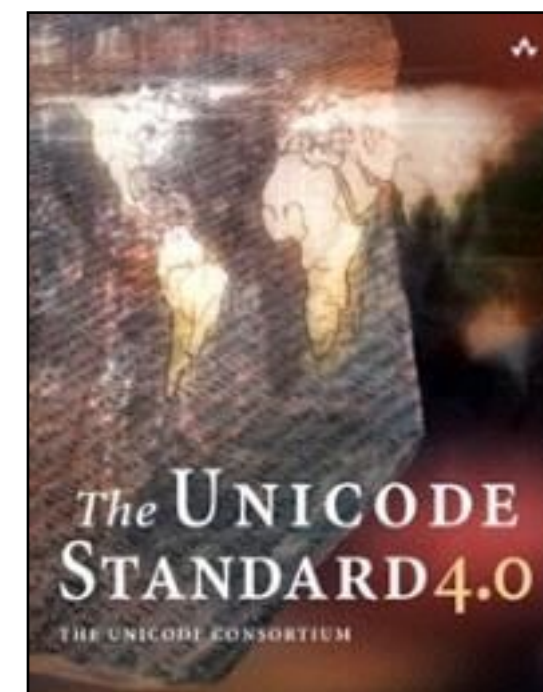
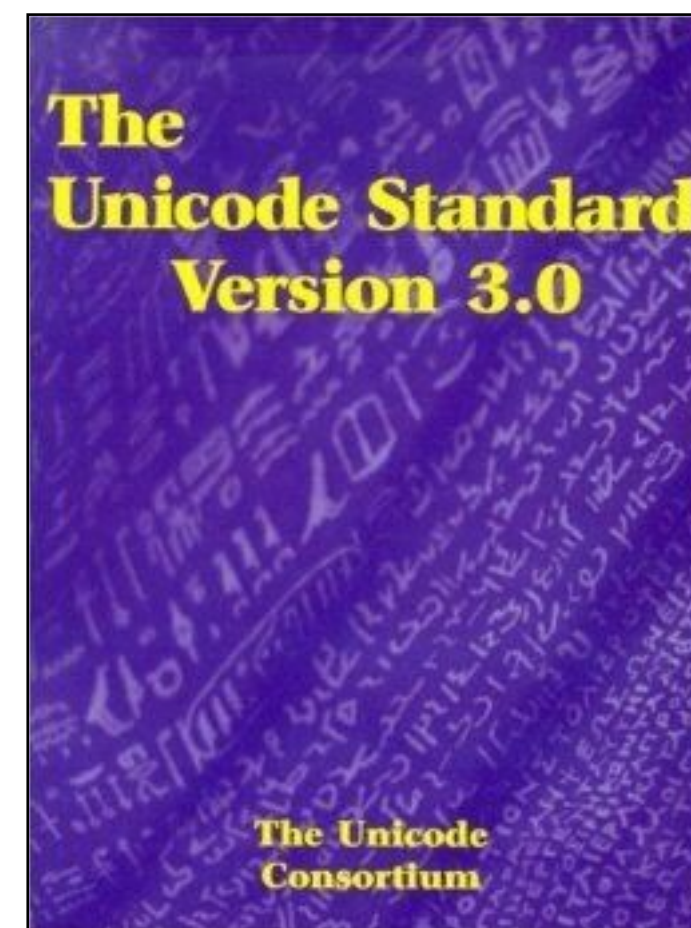
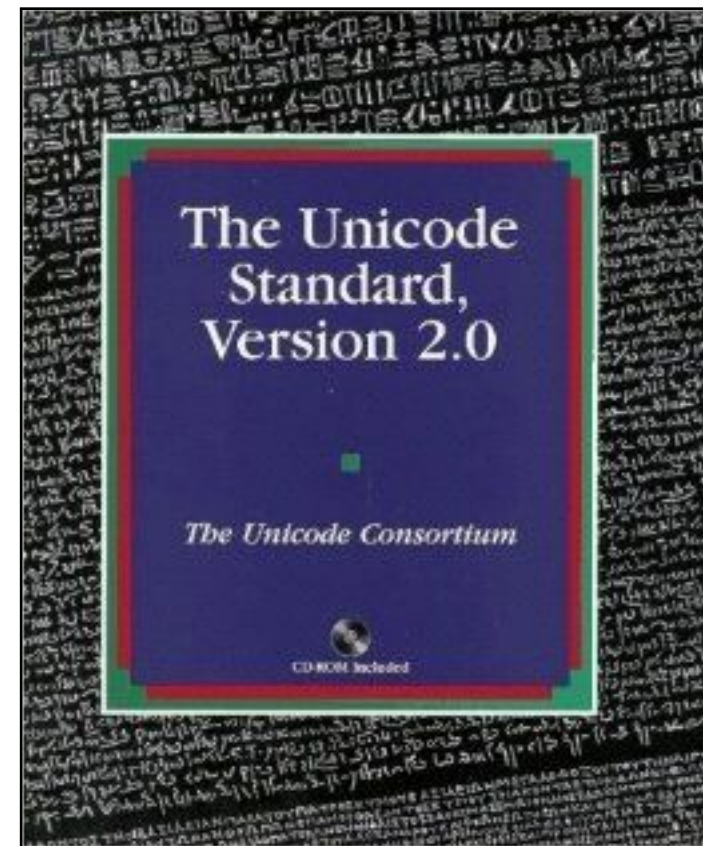
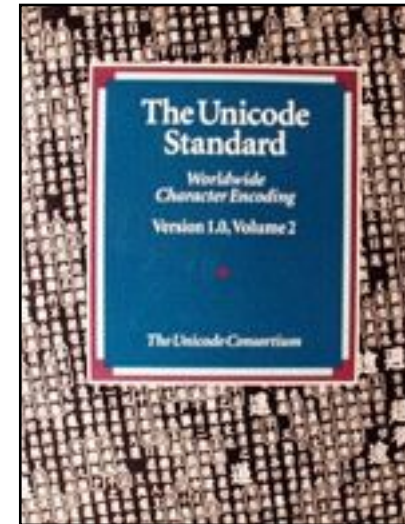
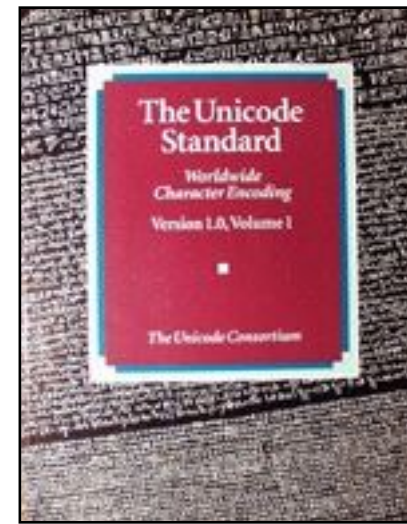
Localization Interoperability Technical Committee

- Data interchange formats for localization-related assets

Editorial Committee

- Edition of the Consortium's publications and web pages

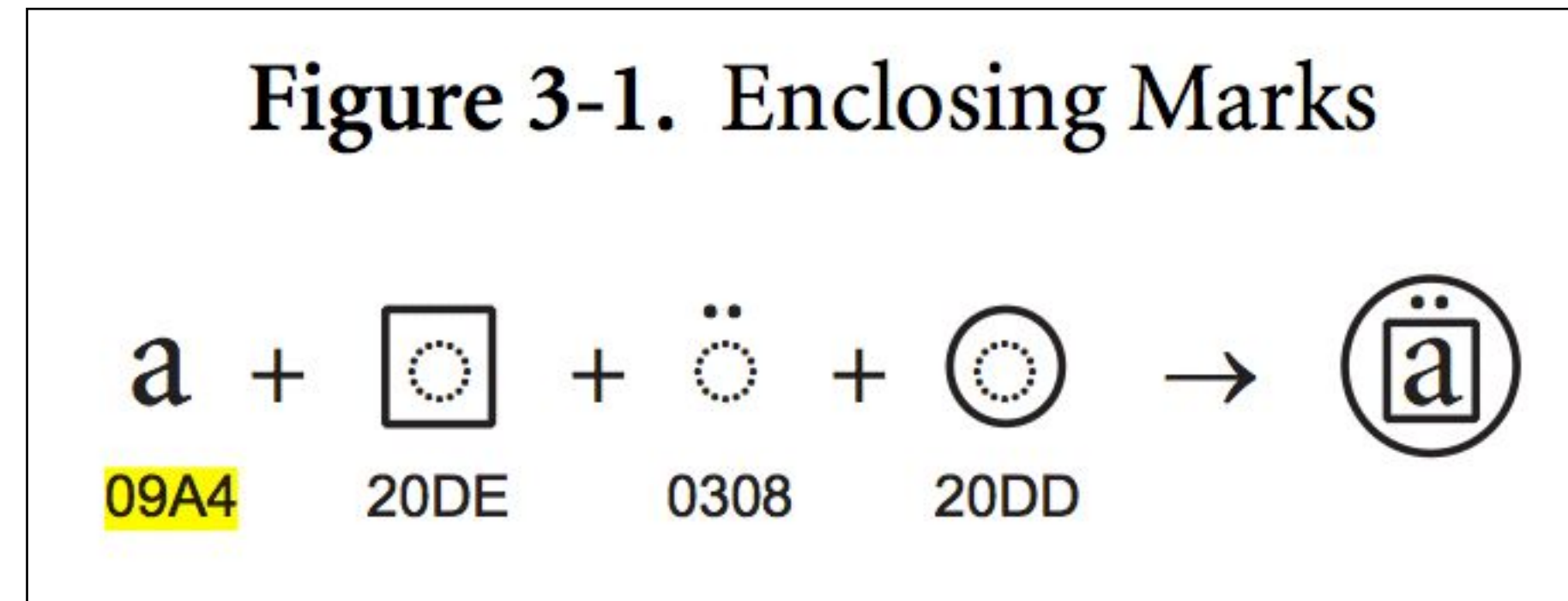
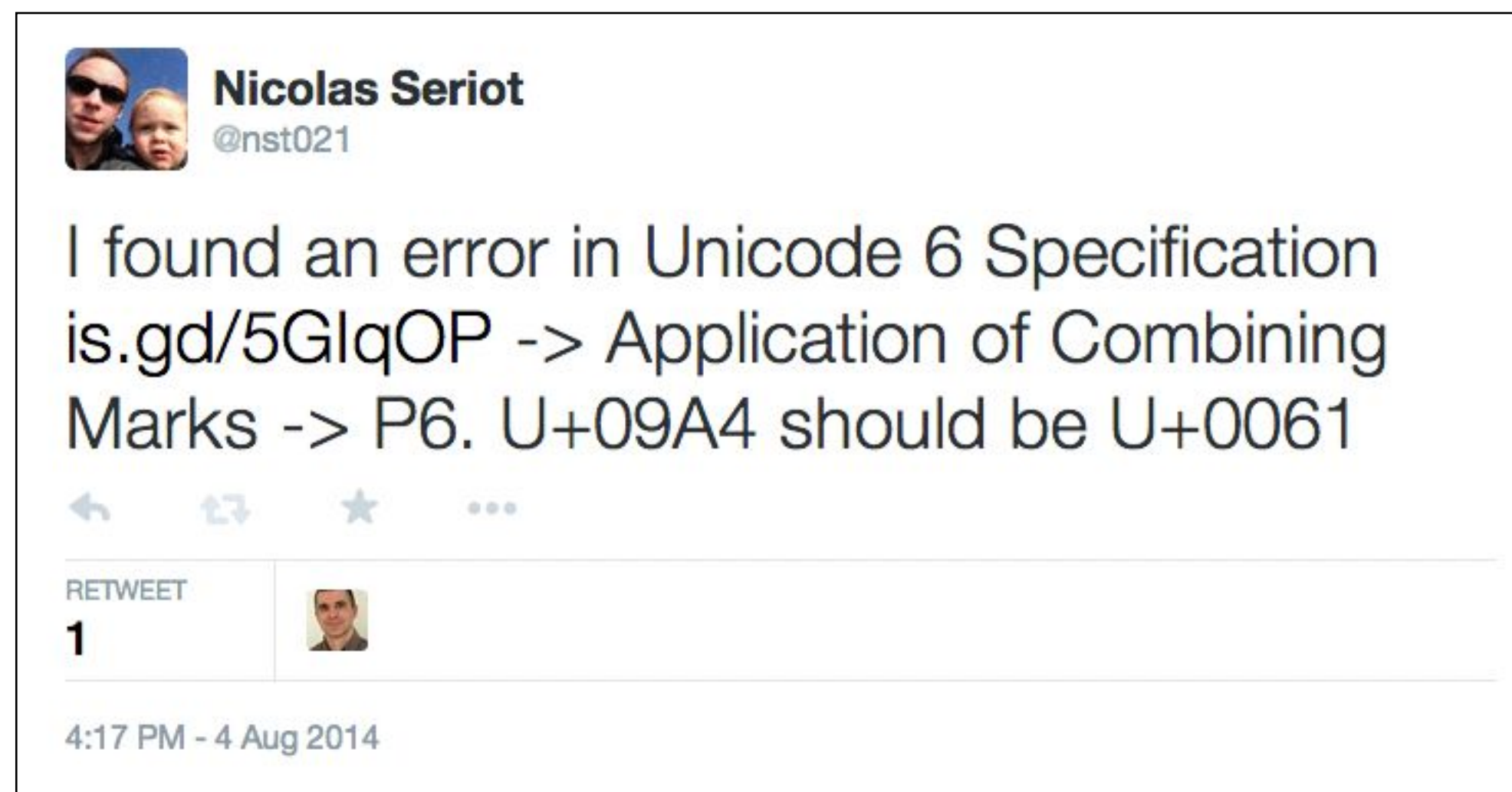
1991














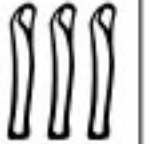

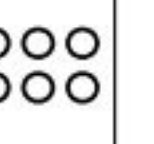













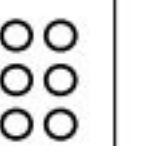














June 2014















<http://www.unicode.org/versions/Unicode7.0.0/UnicodeStandard-7.0.pdf>

You can still Find Errors, Though...



<http://www.unicode.org/versions/Unicode7.0.0/ch03.pdf>

13000		Egyptian Hieroglyphs												130DF	
		1300	1301	1302	1303	1304	1305	1306	1307	1308	1309	130A	130B	130C	130D
0															
		13000	13010	13020	13030	13040	13050	13060	13070	13080	13090	130A0	130B0	130C0	130D0
1															
		13001	13011	13021	13031	13041	13051	13061	13071	13081	13091	130A1	130B1	130C1	130D1
2															
		13002	13012	13022	13032	13042	13052	13062	13072	13082	13092	130A2	130B2	130C2	130D2

13000		Egyptian Hieroglyphs		13073	
<p><i>The characters in this block are taken primarily from Alan Gardiner's works on Middle Egyptian.</i></p>					
<p>A. Man and his occupations</p>					
13000		EGYPTIAN HIEROGLYPH A001	1303A		EGYPTIAN HIEROGLYPH A049
13001		EGYPTIAN HIEROGLYPH A002	1303B		EGYPTIAN HIEROGLYPH A050
13002		EGYPTIAN HIEROGLYPH A003	1303C		EGYPTIAN HIEROGLYPH A051
13003		EGYPTIAN HIEROGLYPH A004	1303D		EGYPTIAN HIEROGLYPH A052
13004		EGYPTIAN HIEROGLYPH A005	1303E		EGYPTIAN HIEROGLYPH A053
			1303F		EGYPTIAN HIEROGLYPH A054
			13040		EGYPTIAN HIEROGLYPH A055
			13041		EGYPTIAN HIEROGLYPH A056
			13042		EGYPTIAN HIEROGLYPH A057

Code Charts

<http://www.unicode.org/charts/>

聾	聾	聾	聽	聵	聵	職	瞻
8071	8072	8073	8074	8075	8076	8077	8078
健	腭	腳	腴	股	股	腩	腸
8171	8172	8173	8174	8175	8176	8177	8178
艱	色	艷	艷	艷	艷	艷	艸
8271	8272	8273	8274	8275	8276	8277	8278
毫	莖	荳	扶	葱	吟	荷	葶
8371	8372	8373	8374	8375	8376	8377	8378
葱	蓂	葳	葳	葵	葶	葷	蔥



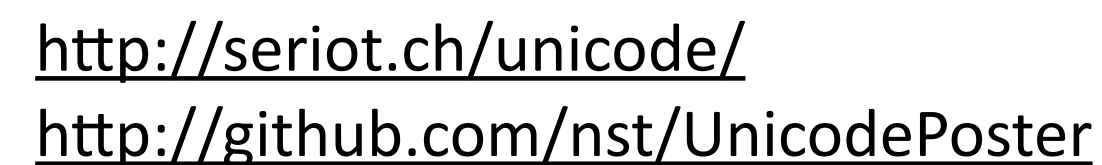
Ian Albert Unicode Chart

TIF, 100.8 MB

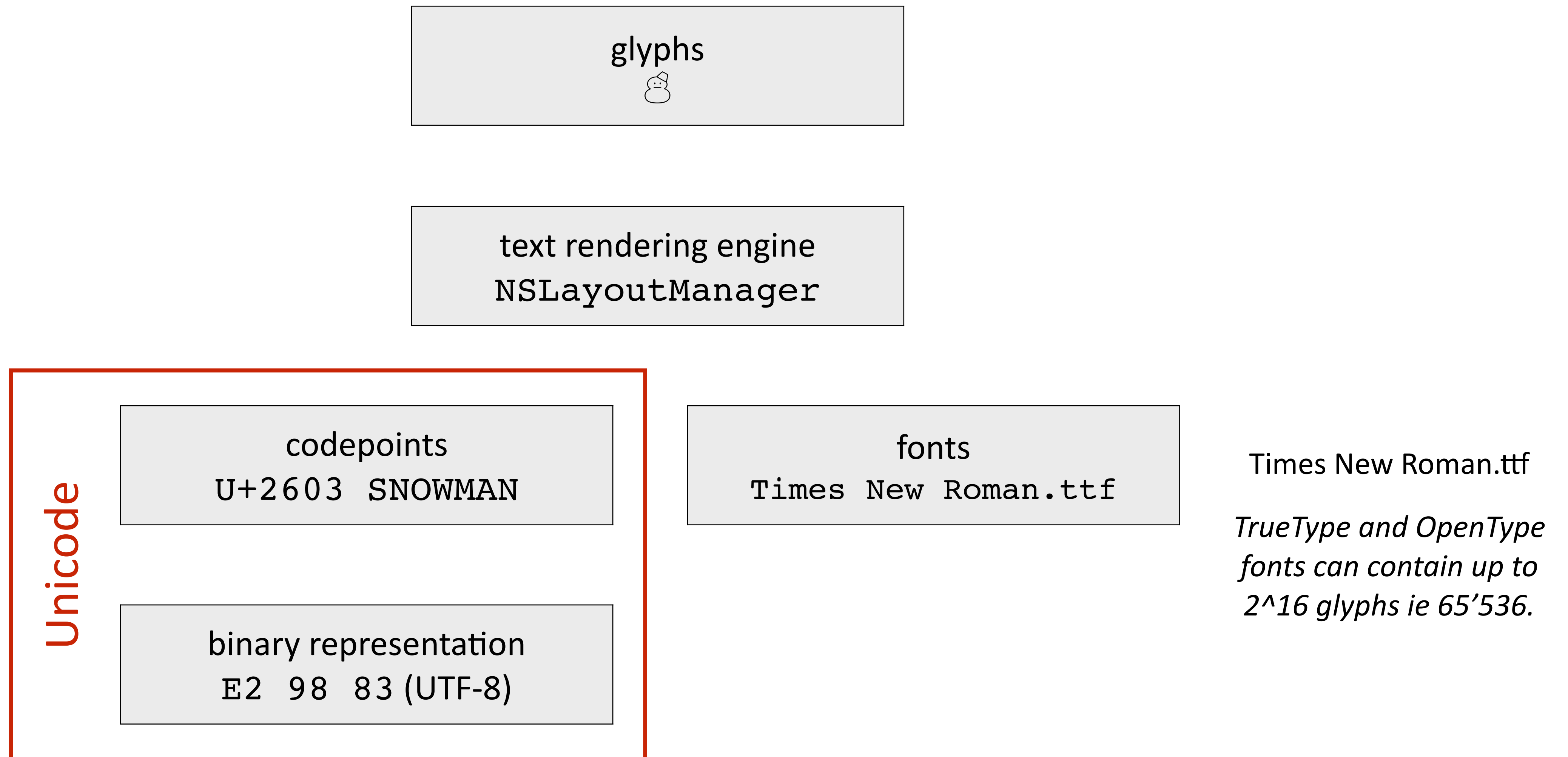
1'114'112 code points

22'017 x 42'807 pixels

http://ian-albert.com/unicode_chart/

[illegible]

Unicode does not address characters rendering



<div>KANBUN</div> <div>天</div> <div>KANBUN</div> <div>0x70</div>	<div>BOPOMOFO</div> <div>ㄅ</div> <div>EXT.</div> <div>0x71</div>	<div>CJK BASIC</div> <div>ノ</div> <div>STROKES</div> <div>0x72</div>	<div>KATAKANA</div> <div>ク</div> <div>EXT.</div> <div>0x73</div>	<div>ENCLOSED</div> <div>(木)</div> <div>CJK</div> <div>0x74</div>	<div>CJK</div> <div>ア パ ー ト</div> <div>COMPAT.</div> <div>0x75</div>	<div>CJK IDEOGR.</div> <div>止</div> <div>EXT. A</div> <div>0x76</div>	<div>YIJING</div> <div>䷀</div> <div>HEXAGRAMS</div> <div>0x77</div>	<div>CJK</div> <div>字</div> <div>IDEOGRAPHS</div> <div>0x78</div>	<div>VI SYLLABLES</div> <div>𐄎</div> <div>VI SYLLABLES</div> <div>0x79</div>	<div>VI RADICALS</div> <div>𐄎</div> <div>VI RADICALS</div> <div>0x7A</div>	<div>LISU</div> <div>ꐣ</div> <div>LISU</div> <div>0x7B</div>	<div>UAI</div> <div>ꐣ</div> <div>UAI</div> <div>0x7C</div>	<div>BAMUM</div> <div>ꐣ</div> <div>BAMUM</div> <div>0x7D</div>	<div>CYRILLIC</div> <div>Ы</div> <div>EXT. B</div> <div>0x7E</div>	<div>MODIFIER</div> <div>Г</div> <div>TONE LETTERS</div> <div>0x7F</div>
--	--	--	--	---	--	---	---	---	--	--	--	--	--	--	--

<div>LATIN EXT. D</div> <div>AA</div> <div>LATIN EXT. D</div> <div>0x80</div>	<div>SYLOTI NAGRI</div> <div>𑖦</div> <div>SYLOTI NAGRI</div> <div>0x81</div>	<div>CMN. INDIC</div> <div>𑖦</div> <div>NUM. FORMS</div> <div>0x82</div>	<div>PHAGS-PA</div> <div>𑖦</div> <div>PHAGS-PA</div> <div>0x83</div>
---	--	--	--

Apple Last Resort Font

<div>TAI VIET</div> <div>ꐣ</div> <div>TAI VIET</div> <div>0x8C</div>	<div>MEETEI</div> <div>ꐣ</div> <div>MAVEK EXT.</div> <div>0x8D</div>	<div>ETHIOPIC</div> <div>ሐ</div> <div>EXT. A</div> <div>0x8E</div>	<div>MEETEI</div> <div>ꐣ</div> <div>MAVEK</div> <div>0x8F</div>
--	--	--	---

<div>HANGUL</div> <div>가</div> <div>SYLLABLES</div> <div>0x90</div>	<div>HNGL. JAMO</div> <div>가</div> <div>EXT. B</div> <div>0x91</div>	<div>HIGH</div> <div>A</div> <div>SURROGATES</div> <div>0x92</div>	<div>PRIVATE</div> <div>A</div> <div>SURROGATES</div> <div>0x93</div>	<div>LOW</div> <div>A</div> <div>SURROGATES</div> <div>0x94</div>	<div>PRIVATE USE</div> <div>W</div> <div>PRIVATE USE</div> <div>0x95</div>	<div>CJK COMPAT.</div> <div>什</div> <div>IDEOGR.</div> <div>0x96</div>	<div>ALPHA. PRES.</div> <div>fi</div> <div>FORMS</div> <div>0x97</div>	<div>ARABIC</div> <div>بن</div> <div>PRES. A</div> <div>0x98</div>	<div>ILLEGAL</div> <div>⊘</div> <div>NOT UNICODE</div> <div>0x99</div>	<div>VI SYLLABLES</div> <div>VS</div> <div>SELECTORS</div> <div>0x9A</div>	<div>VERTICAL</div> <div>’</div> <div>FORMS</div> <div>0x9B</div>	<div>COMBINING</div> <div>◌̇</div> <div>HALF FORMS</div> <div>0x9C</div>	<div>CJK COMPAT.</div> <div>𐄎</div> <div>FORMS</div> <div>0x9D</div>	<div>SMALL FORM</div> <div>𐄎</div> <div>VARIANTS</div> <div>0x9E</div>	<div>ARABIC</div> <div>ب</div> <div>PRES. B</div> <div>0x9F</div>
---	--	--	---	---	--	--	--	--	--	--	---	--	--	--	---

<div>HALF & FULL</div> <div>力</div> <div>WIDTH FORMS</div> <div>0xA0</div>	<div>SPECIALS</div> <div>?</div> <div>SPECIALS</div> <div>0xA1</div>	<div>ILLEGAL</div> <div>⊘</div> <div>NOT UNICODE</div> <div>0xA2</div>	<div>UNDEFINED</div> <div>1</div> <div>PLANE 1</div> <div>0xA3</div>	<div>LINEAR B</div> <div>𐀀</div> <div>SYLLABARY</div> <div>0xA4</div>	<div>LINEAR B</div> <div>𐀀</div> <div>IDEOGRAPHS</div> <div>0xA5</div>	<div>EGEAN</div> <div>𐀀</div> <div>NUMBERS</div> <div>0xA6</div>	<div>ANC. GREEK</div> <div>Α</div> <div>NUMBERS</div> <div>0xA7</div>	<div>ANCIENT</div> <div>2</div> <div>SYMBOLS</div> <div>0xA8</div>	<div>PHAISTOS</div> <div>𐀀</div> <div>DISC</div> <div>0xA9</div>	<div>LYCIAN</div> <div>Α</div> <div>LYCIAN</div> <div>0xAA</div>	<div>CARIAN</div> <div>A</div> <div>CARIAN</div> <div>0xAB</div>	<div>OLD ITALIC</div> <div>A</div> <div>OLD ITALIC</div> <div>0xAC</div>	<div>GOthic</div> <div>A</div> <div>GOthic</div> <div>0xAD</div>	<div>UGARITIC</div> <div>𐎀</div> <div>UGARITIC</div> <div>0xAE</div>	<div>OLD PERSIAN</div> <div>𐎀</div> <div>OLD PERSIAN</div> <div>0xAF</div>
--	--	--	--	---	--	--	---	--	--	--	--	--	--	--	--

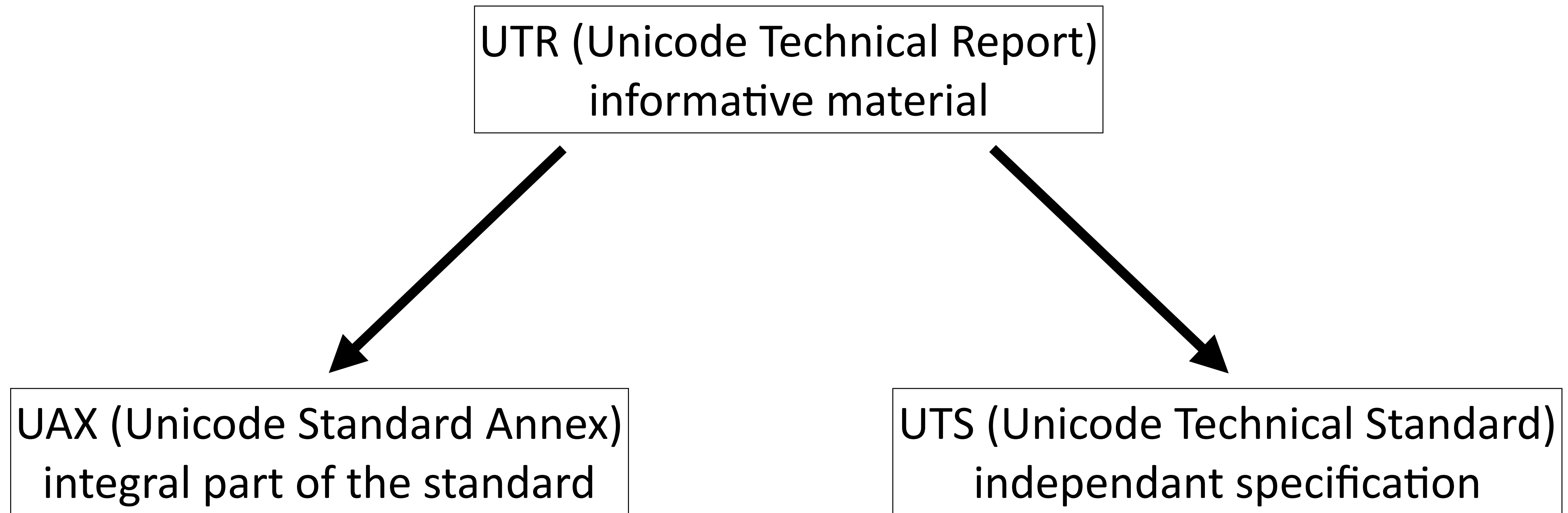
<div>DESERET</div> <div>ᑭ</div> <div>DESERET</div> <div>0xB0</div>	<div>SHAVIAN</div> <div>ᑭ</div> <div>SHAVIAN</div> <div>0xB1</div>	<div>OSMANYA</div> <div>ᑭ</div> <div>OSMANYA</div> <div>0xB2</div>	<div>CYPRIOt</div> <div>ᑭ</div> <div>SYLLABARY</div> <div>0xB3</div>	<div>IMPERIAL</div> <div>ᑭ</div> <div>ARABIC</div> <div>0xB4</div>	<div>PHOENICIAN</div> <div>ᑭ</div> <div>PHOENICIAN</div> <div>0xB5</div>	<div>LYDIAN</div> <div>Α</div> <div>LYDIAN</div> <div>0xB6</div>	<div>MEROITIC</div> <div>𐀀</div> <div>HIERO.</div> <div>0xB7</div>	<div>MEROITIC</div> <div>𐀀</div> <div>CURSIVE</div> <div>0xB8</div>	<div>KHAROSHThI</div> <div>𐀀</div> <div>KHAROSHThI</div> <div>0xB9</div>	<div>OLD SOUTH</div> <div>𐀀</div> <div>ARABIAN</div> <div>0xBA</div>	<div>AVESTAN</div> <div>𐀀</div> <div>AVESTAN</div> <div>0xBB</div>	<div>INSCR.</div> <div>𐀀</div> <div>PARTHIAN</div> <div>0xBC</div>	<div>INSCR.</div> <div>𐀀</div> <div>PAHLAVI</div> <div>0xBD</div>	<div>OLD TURKIC</div> <div>𐀀</div> <div>OLD TURKIC</div> <div>0xBE</div>	<div>RUMI</div> <div>ᑭ</div> <div>NUM. SYM.</div> <div>0xBF</div>
--	--	--	--	--	--	--	--	---	--	--	--	--	---	--	---

<div>BRAHMI</div> <div>𑀀</div> <div>BRAHMI</div> <div>0xC0</div>	<div>KAITHI</div> <div>𑀀</div> <div>KAITHI</div> <div>0xC1</div>	<div>SORA</div> <div>𑀀</div> <div>SOMPENG</div> <div>0xC2</div>	<div>CHAKMA</div> <div>𑀀</div> <div>CHAKMA</div> <div>0xC3</div>	<div>SHARADA</div> <div>𑀀</div> <div>SHARADA</div> <div>0xC4</div>	<div>TAKRI</div> <div>𑀀</div> <div>TAKRI</div> <div>0xC5</div>	<div>CUNEIFORM</div> <div>𑀀</div> <div>CUNEIFORM</div> <div>0xC6</div>	<div>CUNEIFORM</div> <div>𑀀</div> <div>NUMBERS</div> <div>0xC7</div>	<div>EGYPTIAN</div> <div>𑀀</div> <div>HIERO.</div> <div>0xC8</div>	<div>BAMUM</div> <div>𑀀</div> <div>SUPPL.</div> <div>0xC9</div>	<div>MIAO</div> <div>𑀀</div> <div>MIAO</div> <div>0xCA</div>	<div>KANA</div> <div>𑀀</div> <div>SUPPL.</div> <div>0xCB</div>	<div>BYZANTINE</div> <div>𑀀</div> <div>MUSICAL</div> <div>0xCC</div>	<div>MUSICAL</div> <div>𑀀</div> <div>SYMBOLS</div> <div>0xCD</div>	<div>ANC. GREEK</div> <div>𑀀</div> <div>MUSICAL</div> <div>0xCE</div>	<div>TAI HUAN JING</div> <div>𑀀</div> <div>SYMBOLS</div> <div>0xCF</div>
--	--	---	--	--	--	--	--	--	---	--	--	--	--	---	--

<div>COUNTING ROD</div> <div>𑀀</div> <div>NUMBERS</div> <div>0xD0</div>	<div>ALPHANUM</div> <div>A</div> <div>MATH SYMBOLS</div> <div>0xD1</div>	<div>ARAB. MATH.</div> <div>ط</div> <div>ALPHA. SYM.</div> <div>0xD2</div>	<div>MAH JONG</div> <div>東</div> <div>TILES</div> <div>0xD3</div>	<div>DOMINO</div> <div>÷</div> <div>TILES</div> <div>0xD4</div>	<div>PLAYING</div> <div>♠</div> <div>CARDS</div> <div>0xD5</div>	<div>ENCL. ALPHA</div> <div>(A)</div> <div>SUPPL.</div> <div>0xD6</div>	<div>ENCL. IDEO.</div> <div>ほか</div> <div>SUPPL.</div> <div>0xD7</div>	<div>MISC. SYM.</div> <div>🌐</div> <div>AND PICT.</div> <div>0xD8</div>	<div>EMOTICONS</div> <div>😺</div> <div>EMOTICONS</div> <div>0xD9</div>	<div>TRANSPORT</div> <div>🚀</div> <div>MAP SYM.</div> <div>0xDA</div>	<div>ALCHEMICAL</div> <div>W</div> <div>SYMBOLS</div> <div>0xDB</div>	<div>ILLEGAL</div> <div>⊘</div> <div>NOT UNICODE</div> <div>0xDC</div>	<div>UNDEFINED</div> <div>2</div> <div>PLANE 2</div> <div>0xDD</div>	<div>CJK IDEOGR.</div> <div>𑀀</div> <div>EXT. B</div> <div>0xDE</div>	<div>CJK IDEOGR.</div> <div>𑀀</div> <div>EXT. C</div> <div>0xDF</div>
---	--	--	---	---	--	---	--	---	--	---	---	--	--	---	---

<div>CJK IDEOGR.</div> <div>𑀀</div> <div>EXT. D</div> <div>0xE0</div>	<div>CJK IDEOGR.</div> <div>𑀀</div> <div>COMPAT. SUP.</div> <div>0xE1</div>	<div>ILLEGAL</div> <div>⊘</div> <div>NOT UNICODE</div> <div>0xE2</div>	<div>UNDEFINED</div> <div>3</div> <div>PLANE 3</div> <div>0xE3</div>	<div>ILLEGAL</div> <div>⊘</div> <div>NOT UNICODE</div> <div>0xE4</div>	<div>UNDEFINED</div> <div>14</div> <div>PLANE 14</div> <div>0xE5</div>	<div>TAGS</div> <div>𑀀</div> <div>TAGS</div> <div>0xE6</div>	<div>VI SYLLABLES</div> <div>VS</div> <div>SELECTORS A</div> <div>0xE7</div>	<div>ILLEGAL</div> <div>⊘</div> <div>NOT UNICODE</div> <div>0xE8</div>	<div>PRIVATE USE</div> <div>♄</div> <div>PLANE 15</div> <div>0xE9</div>	<div>ILLEGAL</div> <div>⊘</div> <div>NOT UNICODE</div> <div>0xEA</div>	<div>PRIVATE USE</div> <div>𑀀</div> <div>PLANE 16</div> <div>0xEB</div>	<div>ILLEGAL</div> <div>⊘</div> <div>NOT UNICODE</div> <div>0xEC</div>	<div>PRIVATE USE</div> <div>𑀀</div> <div>PLANE 17</div> <div>0xED</div>	<div>ILLEGAL</div> <div>⊘</div> <div>NOT UNICODE</div> <div>0xEE</div>	<div>ILLEGAL</div> <div>⊘</div> <div>NOT UNICODE</div> <div>0xEF</div>
---	---	--	--	--	--	--	--	--	---	--	---	--	---	--	--

Unicode Technical Reports



<http://www.unicode.org/reports/about-reports.html>

Unicode Character Database (UCD), TR#44 (UAX)

<http://www.unicode.org/Public/UCD/latest/ucd/UnicodeData.txt>

00E9;LATIN SMALL LETTER E WITH ACUTE;Ll;0;L;0065 0301;;;;N;LATIN SMALL LETTER E ACUTE;;00C9;;00C9

0. Codepoint	00E9	
1. Name	LATIN SMALL LETTER E WITH ACUTE	
2. General_Category	Ll	a lowercase letter
3. Canonical_Combining_Class	0	not reordered
4. Bidi_Class	L	left to right
5. Decomposition_Type,	0065 0301	
6. Numeric_Type, Numeric Value		
7. Numeric_Type, Numeric Value		
8. Numeric_Type, Numeric Value		
9. Bidi_Mirrored	N	Y if mirrored in a bidirectional text
10. Unicode_1_Name (Obsolete)	LATIN SMALL LETTER E ACUTE	name in Unicode 1.0
11. ISO_Comment (Obsolete)		
12. Simple_Uppercase_Mapping	00C9	
13. Simple_Lowercase_Mapping		already lowercase
14. Simple_Titlecase_Mapping	00C9	

Unicode Technical Committee Minutes

The [Unicode Technical Committee \(UTC\)](#) meets quarterly each year. Meeting minutes document the decisions, actions and voting record of the Full, Institutional, and Supporting Members of the Committee through numbered motions, consensus statements, and action items. Approved meeting minutes are ones that have been reviewed and approved by the UTC, preliminary minutes are ones posted for final public review prior to their approval at the next meeting of the UTC.

In addition, draft minutes are available via the [current document register](#). These are unapproved minutes from the most recent UTC and are subject to revision before final versions are posted.

UTC Minutes	Status	Location	Dates
UTC 143		San Jose, CA	May 4-8, 2015
UTC 142		San Jose, CA	Feb. 2-5, 2015
UTC 141		Sunnyvale, CA	Oct. 27-31, 2014
UTC 140	<i>Draft</i>	Redmond, WA	August 5-8, 2014
UTC 139	<i>Draft</i>	San Jose, CA	May 6-9, 2014
UTC 138	<i>Draft</i>	San Jose, CA	Feb 3-6, 2014
UTC 137	<i>Draft</i>	Cupertino, CA	November 4-7, 2013
UTC 136	<i>Approved</i>	Redmond, WA	July 29 - August 2, 2013
UTC 135	<i>Approved</i>	San Jose, CA	May 6 - 10, 2013
UTC 134	<i>Approved</i>	San Jose, CA	Jan 28 - Feb 1, 2013
UTC 133	<i>Approved</i>	Cupertino, CA	November 5 - 9, 2012
UTC 132	<i>Draft</i>	Redmond, WA	July 30 - August 6, 2012
UTC 131	<i>Approved</i>	San Jose, CA	May 7-11, 2012



Eg. Proposal to encode GREEK BYZANTINE DOUBLE SUSPENSION MARK

ISO/IEC JTC 1/SC 2/WG 2		L2/14-157
PROPOSAL SUMMARY FORM TO ACCOMPANY SUBMISSIONS FOR ADDITIONS TO THE REPERTOIRE OF ISO/IEC 10646¹		
Please fill all the sections A, B and C below.		
Please read Principles and Procedures Document (P & P) from http://www.dkuug.dk/JTC1/SC2/WG2/docs/principles.html for guidelines and details before filling this form.		
Please ensure you are using the latest Form from http://www.dkuug.dk/JTC1/SC2/WG2/docs/summaryform.html . See also http://www.dkuug.dk/JTC1/SC2/WG2/docs/roadmaps.html for latest Roadmaps.		
A. Administrative		
1. Title:	Proposal to encode GREEK BYZANTINE DOUBLE SUSPENSION MARK : ' ;	
2. Requester's name:	Dumbarton Oaks	
3. Requester type (Member body/Liaison/Individual contribution):	individual contribution	
4. Submission date:	2014-07-18	
5. Requester's reference (if applicable):		
6. Choose one of the following:		
This is a complete proposal:	yes	
(or) More information will be provided later:		

Byzantine seals

DOSeals 2:40.11 (10th c.): In this example, the typesetter tried, but failed, to get the two characters to align vertically (they used two legacy characters with incompatible kerning values)

40.11 Leo imperial protospatharios and ek prosopou of Aigaion Pelagos (X c.)
DO 58.106.2474.— D. 21 mm. W. 6.42 g. Oxidated.
Unpublished.
Obv. Patriarchal cross on three steps with fleurons (up to first arm). Along the border of dots, inscription: +ΚΕΡΟΗΘΕΙΤΩCΩΔΔΛ.
Rev. Inscription of five lines. *Border of dots.*
+ΛΕΟΝ.[Ρ'Α'CΠΑΘ'.]Ρ.ΣΕΚΠΡΟ.[ΩΠ.Τ.ΕΓΙ.]ΠΕΛΑΓ
+Κ(ύρι)ε βοήθει τῷ σῷ δούλ(ω) Λέον[τ(ι)] β(ασιλικῷ) (πρωτο)σπαθ[α]ρ(ίω)
(καὶ) ἐκ προ[σ]ώπ(ου) τ(οῦ) Ἐγί[ου] Πελάγ(ους).
The form Ἐγίου is not unique, cf. no. 40.7 above. The IXth-century seal of Nicholas ek prosopou (without mention of the province) was discovered in the neighborhood of Methymna (SBS 2 [1990] 167). This could well be an ek prosopou of the Aigaion Pelagos, like the owner of the present seal.



(l) DO 55.1.847.— D. 24 mm. W. 12.67 g. Blank too small for
(m) DO 55.1.848.— D. 23 mm. W. 11.34 g. Blank too small for
(n) DO 55.1.849.— D. 24 mm. W. 15.09 g. Blank too small for
straight sides. Channel off center.
(o) DO 55.1.850.— D. 23 mm. W. 8.12 g. Blank too small for
(p) DO 55.1.851.— D. 25 mm. W. 11.17 g. Blank too small for
straight sides. Channel off center.
(q) DO 55.1.852.— D. 23 mm. W. 11.99 g. Blank too small for
(r) DO 55.1.853.— D. 24 mm. W. 12.57 g. Blank too small for
(s) DO 55.1.854.— D. 26 mm. W. 12.38 g. Channel off center
(t) DO 55.1.855.— D. 24 mm. W. 8.91 g. Blank too small for
(u) DO 55.1.856.— D. 24 mm. W. 13.08 g. Blank too small for
(v) DO 55.1.857.— D. 29 mm. W. 14.19 g.
(w) Fogg 1946.— D. 31 mm. W. 19.97 g. Corroded and cracked
(x) Fogg 2312.— D. 25 mm. W. 10.51 g. Blank too small for
Although the inscriptions of many of the specimens above appear that all are from the same boulloterion. The reading inscriptions.
Ed. Zacos-Veglery, 1886(a) (= our specimen a); 1886(b) (= specimen c).
Obv. Cruciform invocative monogram (type VIII); in the center border.
Rev. Inscription of five lines. *Width border.*
+ΕΝΦΙΜ|ΙΑΝΟΒ.Α.CΠ|ΑΘ.ΣCΤΡΑ|ΤΙΓ.ΕΛΑ|ΔΟC
Κύριε βοήθει τῷ σῷ δούλ(ω) Εὐφίμιαν(ω) β(ασιλ)στρατιγ(ῷ) Ἑλλάδος.
An epigraphical oddity is the presence of double abbreviations. The existence of several seals with the channel well off-center. The blanks used by Euphemianos, some of which also present a mold with some straight (non-curved) sides. These phenomena



<http://www.unicodeconference.org>

<http://www.unicodeconference.org/conference-at-a-glance.htm>

1. The Unicode Consortium

2. Selected Unicode Specifications

3. Unicode in Practice

4. Unicode Hacks

Encodings



PNG: ...

JPEG: ...

BMP: ...

é
U+00E9 LATIN
SMALL LETTER E
WITH ACUTE

UTF-8 : C3 A9

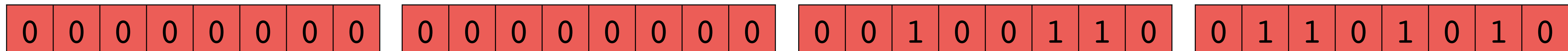
UTF-32: FF FE 00 00
E9 00 00 00

UTF-16: FF FE E9 00

0x0000

- Direct representation of the codepoint on 32 bits.
- Disadvantage: 4 bytes per character is space inefficient.
- Example with U+266A 🎵 « EIGHTH NOTE »

UTF-32



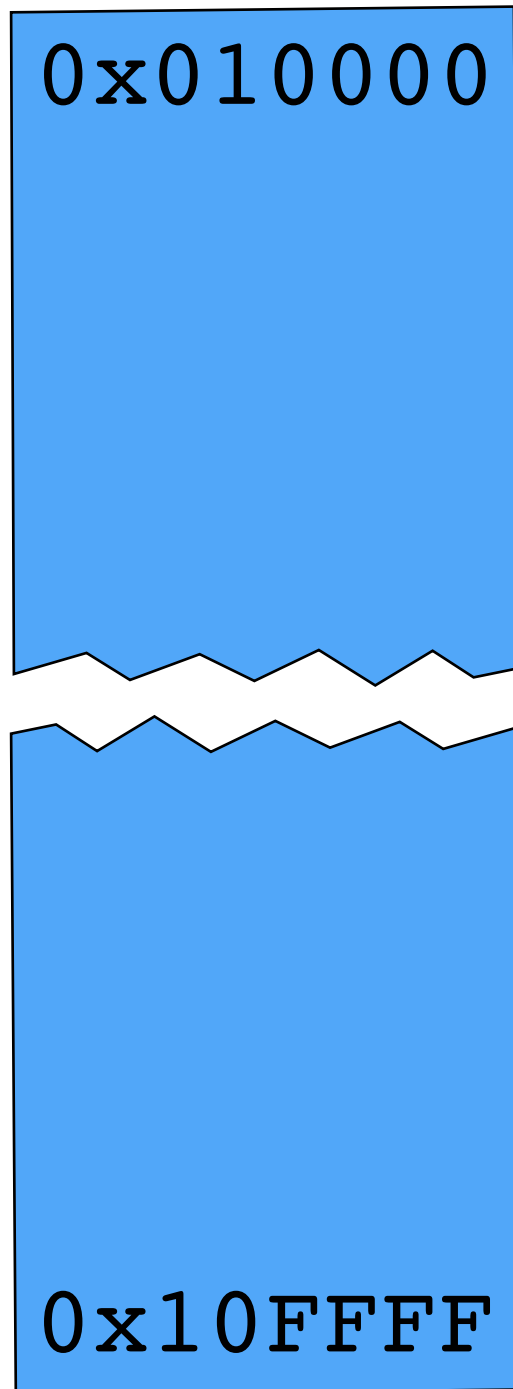
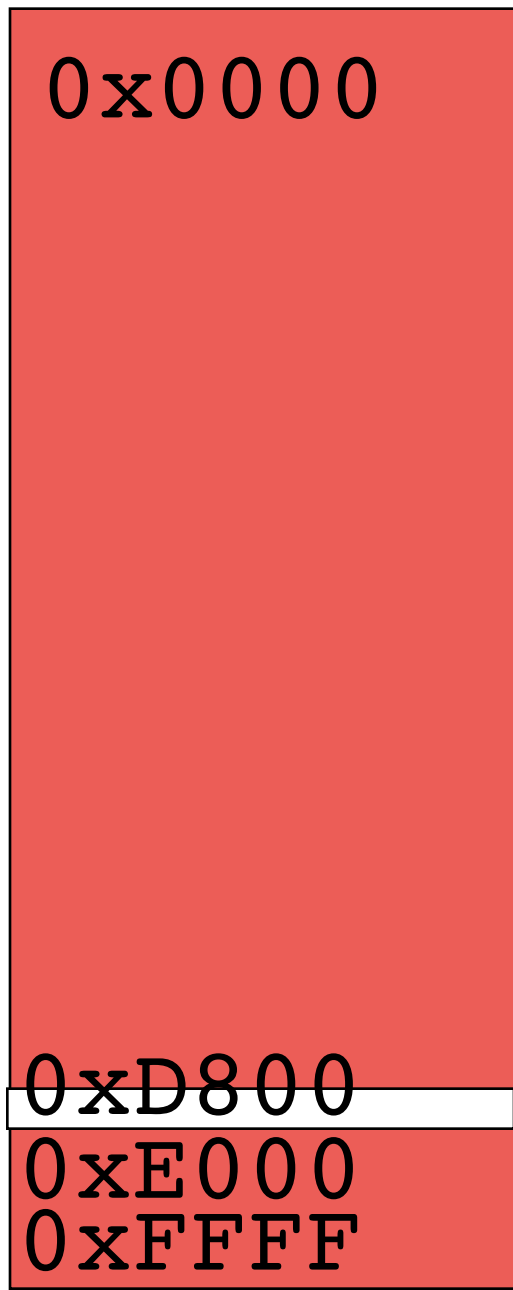
0x00

0x00

0x26

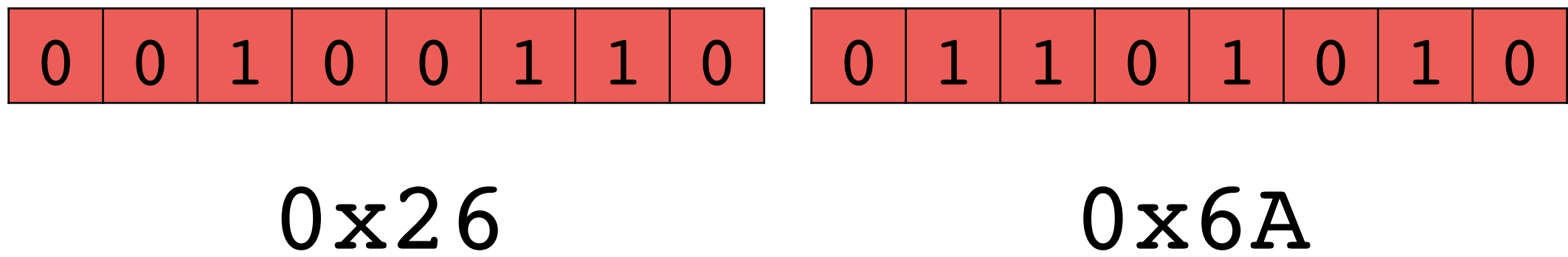
0x6A

0x10FFF

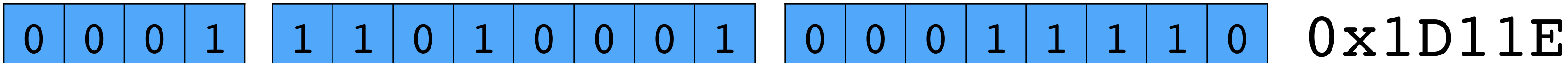


- Most common 63K characters encoded on single 16 bits code units.
- Example with U+266A ♪ « EIGHTH NOTE »

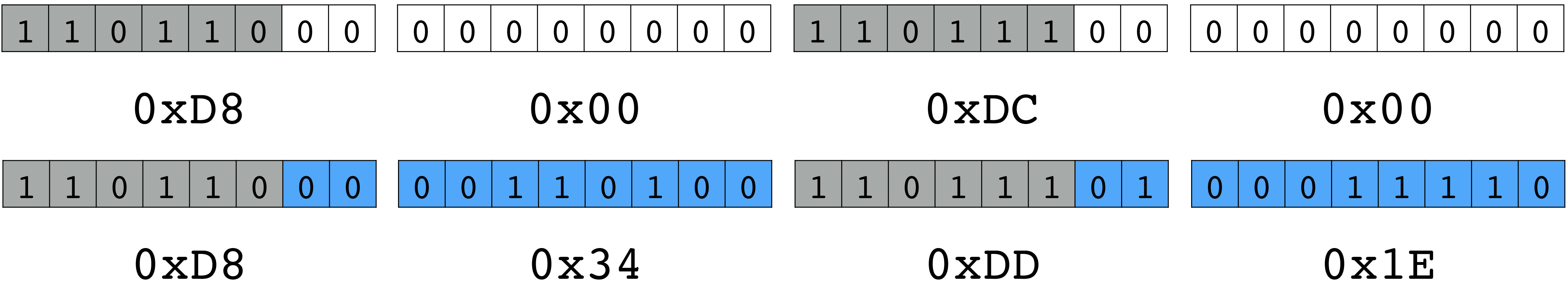
UTF-16

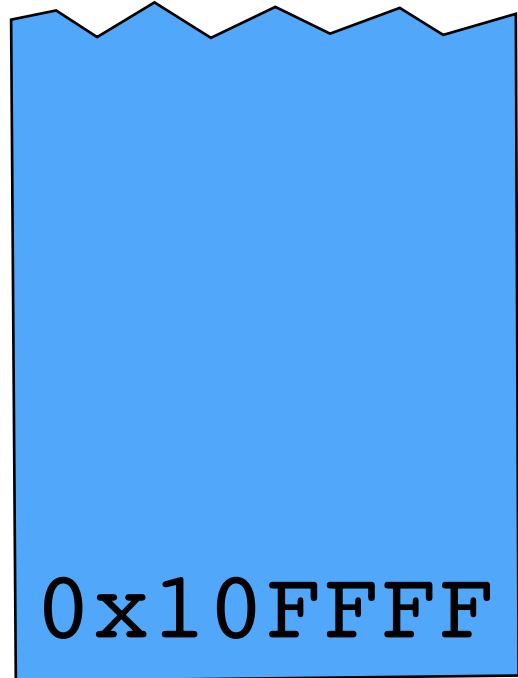
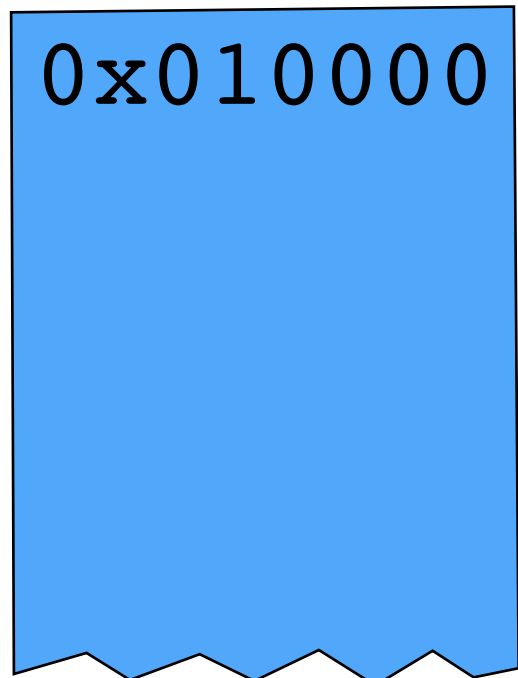
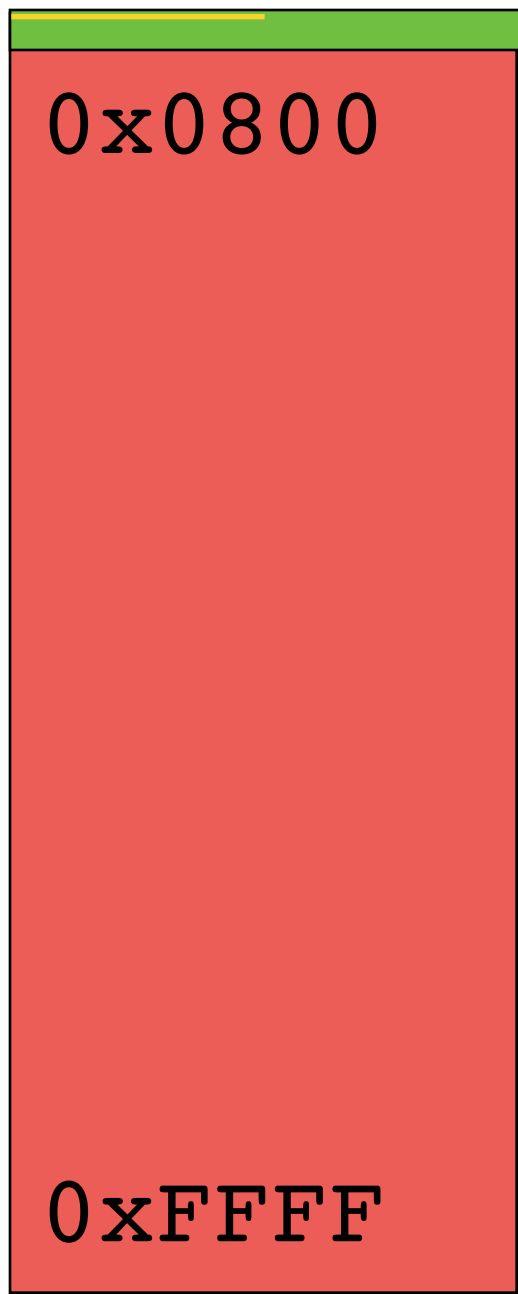


- Other non-BMP codepoints encode 20 bits in a pair of 16 bits surrogates.
- Example with U+1D11E 🎵 « MUSICAL SYMBOL G CLEF »



- Subtract 0x10000 (for a 20 bits space), fill surrogates with 2 times 10 bits





- 7-bits codepoints (« Basic Latin ») U+0041 A « LATIN CAPITAL LETTER A »

1 0 0 0 0 0 1 0x0041

0 1 0 0 0 0 0 1 0x41

UTF-8

- 11-bits codepoints, ie blocks « Latin 1 », « Cyrillic », « Arabic », ...
- Ex. U+036C ϕ « GREEK SMALL LETTER PHI »

0 1 1 1 1 0 0 0 1 1 0 0x03C6

1 1 0 0 1 1 1 1 1 0 0 0 1 1 0 0xCF 0x86

- 16-bits codepoints, ex. U+266A ♪ « EIGHTH NOTE »

0 0 1 0 0 1 1 0 1 1 0 1 0 0x266A

1 1 1 0 0 0 1 0 1 0 0 1 1 0 1 0 1 0 0xE2 0x99 0xAA

- 21-bits codepoints, ex. U+1D11E ♪ « MUSICAL SYMBOL G CLEF »

0 0 0 0 1 1 1 0 0 0 1 0 0 0 1 1 1 1 0 0x1D11E

1 1 1 1 0 0 0 0 1 0 0 1 1 0 1 1 0 0 0 0 1 0 0 1 0 0 1 0 0 1 1 1 1 0

0xF0

0x9D

0x84

0x9E

Figure 2-12. Unicode Encoding Schemes

<div>A</div> <div>00 00 00 41</div>	<div>Ω</div> <div>00 00 03 A9</div>	<div>語</div> <div>00 00 8A 9E</div>	<div>Ⅲ</div> <div>00 01 03 84</div>	UTF-32BE
<div>A</div> <div>41 00 00 00</div>	<div>Ω</div> <div>A9 03 00 00</div>	<div>語</div> <div>9E 8A 00 00</div>	<div>Ⅲ</div> <div>84 03 01 00</div>	UTF-32LE
<div>A</div> <div>00 41</div>	<div>Ω</div> <div>03 A9</div>	<div>語</div> <div>8A 9E</div>	<div>Ⅲ</div> <div>D8 00 DF 84</div>	UTF-16BE
<div>A</div> <div>41 00</div>	<div>Ω</div> <div>A9 03</div>	<div>語</div> <div>9E 8A</div>	<div>Ⅲ</div> <div>00 D8 84 DF</div>	UTF-16LE
<div>A</div> <div>41</div>	<div>Ω</div> <div>CE A9</div>	<div>語</div> <div>E8 AA 9E</div>	<div>Ⅲ</div> <div>F0 90 8E 84</div>	UTF-8

in Unicode Standard 7.0, page 41

Normalization: TR#15 (UAX)

Canonical Equivalence

Two code points sequences with:

- same appearance
- same meaning

Å
U+212B

A ͆
U+0041 U+030A

Compatibility Equivalence

Two code points sequences with:

- possibly distinct appearances
- the same meaning in some contexts

fi
U+FB01

f i
U+0066 U+0069

é ①
U+00E9 U+2460

Canonical decomposition

Compatibility decomposition

e ˙ ①
U+0065 U+0301 U+2460

NFD

NFKD

é ˙ 1
U+0065 U+0301 U+0031

Canonical composition

é ①
U+0065 U+2460

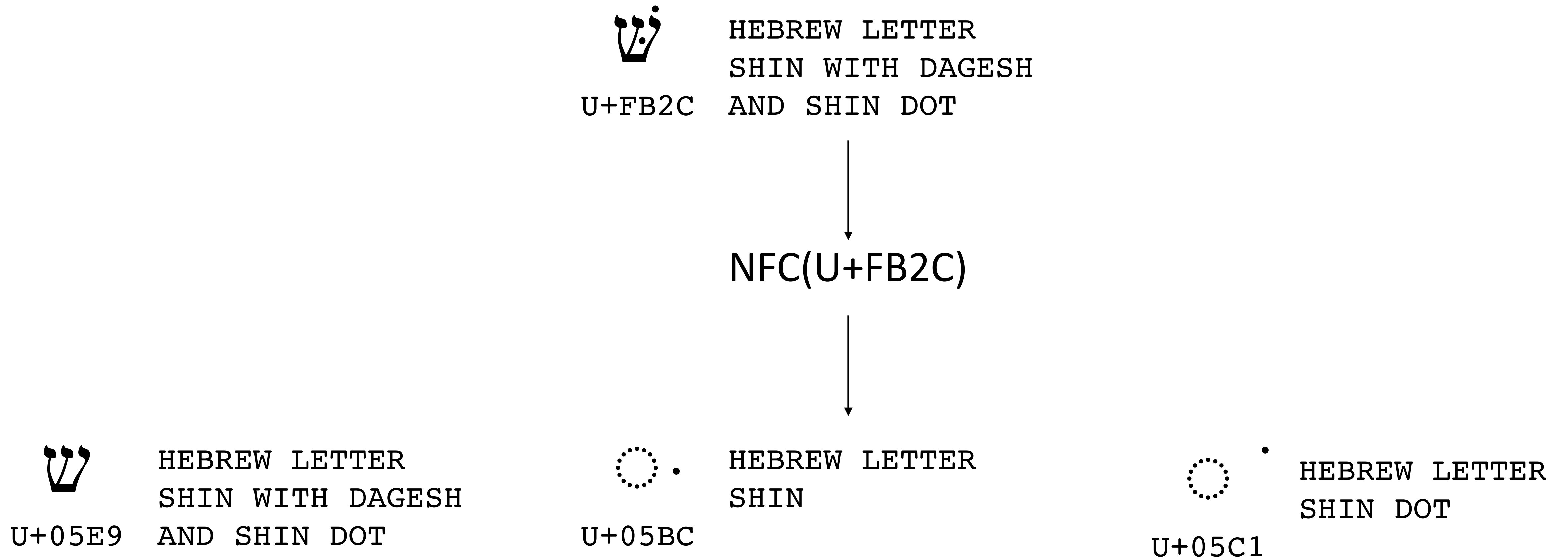
NFC

(most common)

NFKC

é 1
U+00E9 U+0031

NFC doesn't always compose



NFKD Maximum Expansion



U+FDFA
ARABIC
LIGATURE
SALLALLAHOU
ALAYHE
WASALLAM

```
>>> import unicodedata

>>> s = '\uFDFA'
>>> len(s)
1

>>> s_nfkd = unicodedata.normalize('NFKD', s)
>>> s_nfkd.encode('unicode-escape')
b'\\u0635\\u0644\\u0649 \\u0627\\u0644\\u0644\\u0647 \\u0639\\u0644\\u064a\\u0647 \\u0648\\u0633\\u0644\\u0645'

>>> len(s_nfkd)
18
```

Unicode Collation Algorithm (UCA)

- TR#10 (UTS)
- About **text comparison**
café < cafe ?
cafe < café ?
- **Language dependant**
- **Usage dependant**
German dictionary: öf < of
German phonebook: of < öf
- **Customizable**
lower first or upper first, ...
numeric ordering, ...
- **Context dependant**
Normal Accent Ordering
cote < coté < côte < côté
Backward Accent Ordering (FR)
cote < côte < coté < côté
- **Unstable over time**

Language Dependant Collation

German			Swedish
Åkersberga	1	2	Alingsås
Alingsås	2	4	Oskarshamn
Äpplebo	3	7	Utting
Oskarshamn	4	6	Üttfeld
Östersund	5	8	Zwickau
Üttfeld	6	1	Åkersberga
Utting	7	3	Äpplebo
Zwickau	8	5	Östersund

(Steven R. Loomis, Mark Davis)

DUCET (Default Unicode Collation Element Table)

<http://www.unicode.org/Public/UCA/latest/allkeys.txt>

Character	Collation Element	Name
0300 "`"	[.0000.0025.0002]	COMBINING GRAVE ACCENT
0061 "a"	[.190C.0020.0002]	LATIN SMALL LETTER A
0062 "b"	[.1925.0020.0002]	LATIN SMALL LETTER B
0063 "c"	[.193E.0020.0002]	LATIN SMALL LETTER C
0043 "C"	[.193E.0020.0008]	LATIN CAPITAL LETTER C
0064 "d"	[.1953.0020.0002]	LATIN SMALL LETTER D

alphanumeric
ordering

diacritic
ordering

case
ordering

Algorithm

NFD	Collation Element Array
cab	[.193E.0020.0002] [.190C.0020.0002] [.1925.0020.0002]
Cab	[.193E.0020.0008] [.190C.0020.0002] [.1925.0020.0002]
càb	[.193E.0020.0002] [.190C.0020.0002] [.0000.0025.0002] [.1925.0020.0002]
dab	[.1953.0020.0002] [.190C.0020.0002] [.1925.0020.0002]

NFD	Sort Key
cab	193E 190C 1925 0020 0020 0020 0002 0002 0002
Cab	193E 190C 1925 0020 0020 0020 0008 0002 0002
càb	193E 190C 1925 0020 0020 0025 0020 0002 0002 0002 0002
dab	1953 190C 1925 0020 0020 0020 0002 0002 0002

Case Folding

```
# The data supports both implementations that require simple case foldings  
# (where string lengths don't change), and implementations that allow full case folding  
# (where string lengths may grow). Note that where they can be supported, the  
# full case foldings are superior: for example, they allow "MASSE" and "Maße" to match.
```

```
00C9; C; 00E9; # LATIN CAPITAL LETTER E WITH ACUTE
```

```
00DF; F; 0073 0073; # LATIN SMALL LETTER SHARP S
```

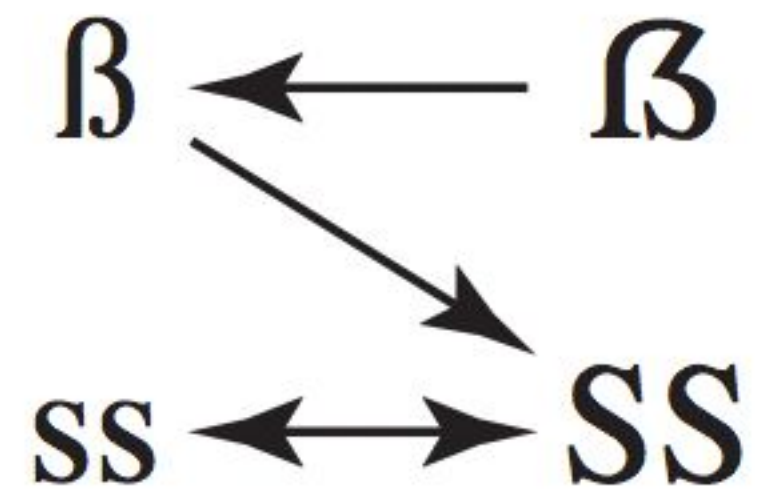
<http://www.unicode.org/Public/UNIDATA/CaseFolding.txt>

<http://userguide.icu-project.org/transforms/casemappings>

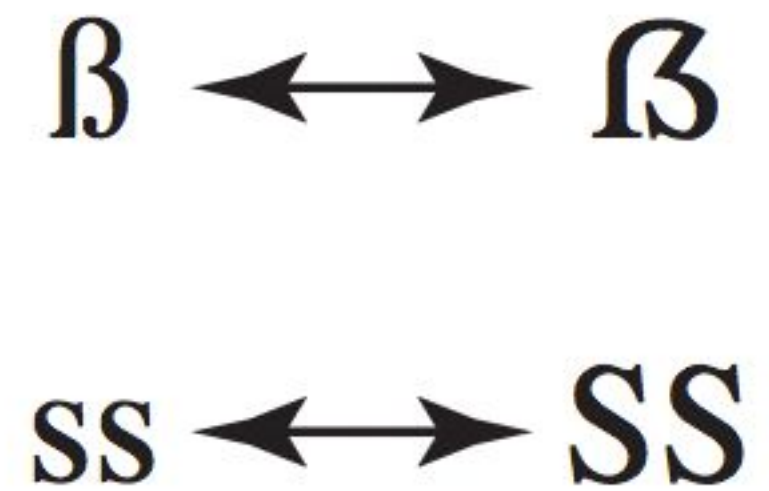
Case Conversion

Figure 5-16. Casing of German Sharp S

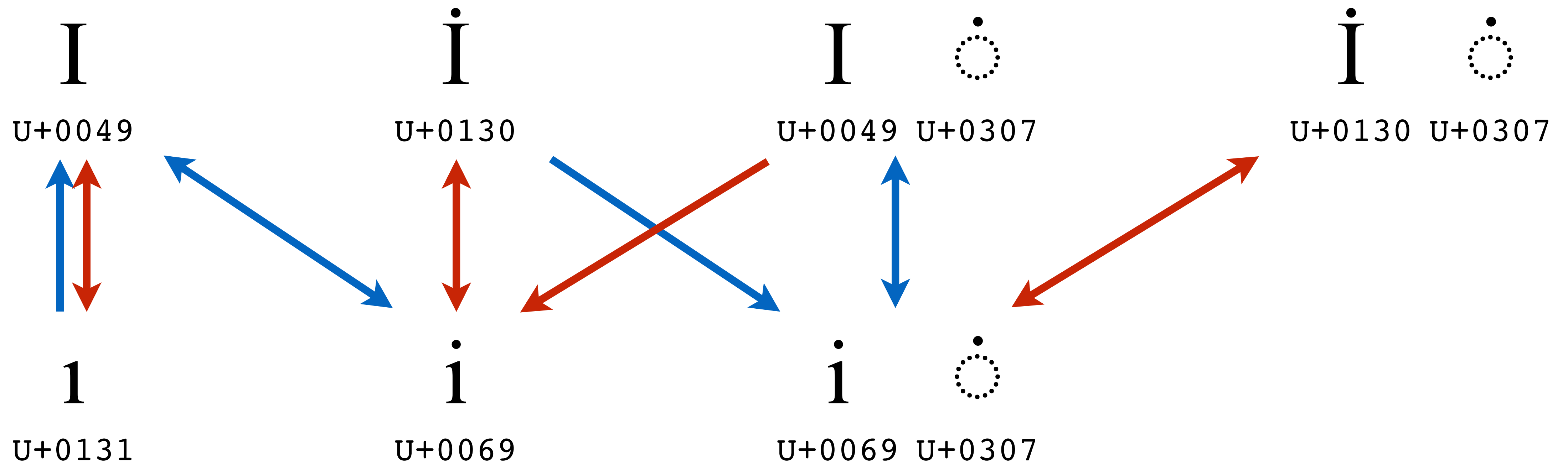
Default Casing



Tailored Casing



Case Conversion



Posix Locale

Turkish Locale

Emojis

絵 (e ≅ picture)

文 (mo ≅ writing)

字 (ji ≅ character)

- Early 2000s: Emoji became generally available on Japanese cell phones.
- Late 2000s, standardized and added into Unicode 6.0 (2010)
- Submit your own: <http://www.unicode.org/pending/proposals.html> and join rejected ones <http://www.unicode.org/alloc/nonapprovals.html>

Emoji Symbols: Background Data

Background data for Proposal for Encoding Emoji Symbols

N3xxx

Date: 2010-Apr-27

Authors:







Markus Scherer, Mark Davis, Kat Momoi, Darick Tong (Google Inc.)
Yasuo Kida, Peter Edberg (Apple Inc.)

This document reflects proposed Emoji symbols data as shown in FDAM8 which includes the disposition of FPDAM8 ballot comments and changes agreed during the San José WG2 meeting 56.

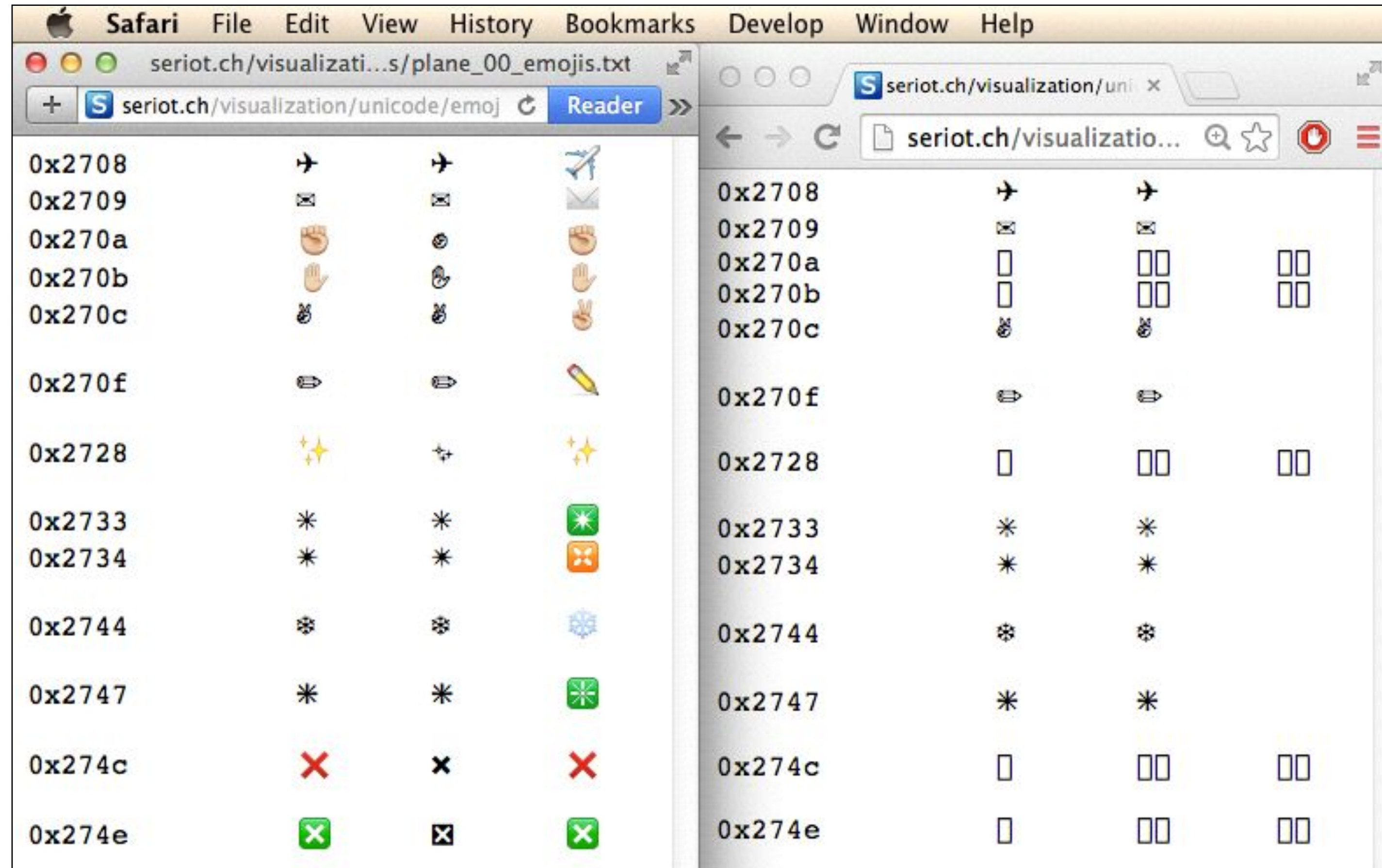
The carrier symbol images in this file point to images on other sites. The images are only for comparison and may change.

See the [chart legend](#) for an explanation of the data presentation in this chart.

In the HTML version of this document, each symbol row has an anchor to allow direct linking by appending [#e-4B0](#) (for example) to this page's URL in the address bar.

Internal ID	Symbol	Name & Annotations	DoCoMo	KDDI	SoftBank	Google
Enclosed alphanumeric symbols						
e-82C	 U+0023 U+20E3 <small>unified (Unicode 3.0)</small>	HASH KEY	# #123 'Sharp dial' シャープダイヤル 「shiyaapudaiyaru」 U+E6E0 SJIS-F985 JIS-7B69	 #818 # U+EB84 SJIS-F489 JIS-7B69	 #403 #old196 # U+E210 SJIS-F7B0	U+FE82C
e-837	 U+0030 U+20E3 <small>unified (Unicode 3.0)</small>	KEYCAP 0	0 #134 U+E6EB SJIS-F990 JIS-784B	 #325 四角数字0 U+E5AC SJIS-F7C9 JIS-784B	 #402 #old217 0 U+E225 SJIS-F7C5	U+FE837
e-82E	 U+0031 U+20E3 <small>unified (Unicode 3.0)</small>	KEYCAP 1	1 #125 '1' 1 U+E6E2 SJIS-F987 JIS-767D	 #180 四角数字1 U+E522 SJIS-F6FB JIS-767D	 #393 #old208 1 U+E21C SJIS-F7BC	U+FE82E
e-82F	 U+0032	KEYCAP 2	2 #126 '2' 2 U+E6E3 SJIS-F988 JIS-767E	 #181 四角数字2 U+E523 SJIS-F6FC JIS-767E	 #394 #old209 2 U+E21D SJIS-F7BD	U+FE82F

Aweful Support in Chrome



Emojis Evolution

- Discussions about Emojis Diversity in meetings minutes
<http://www.unicode.org/L2/L2014/14172r-emoji-enhancements.pdf>
<http://www.unicode.org/L2/L2014/14177.htm#140-C28>
- UTC Meeting [140-A47] Action Item for Mark Davis: Talk to Facebook and Twitter to see if they would like to get more involved.

Variation Selectors

- may modify some glyph appearance
- 16 VS in BMP: U+FE00 to U+FEFF
- 240 more VS in plane 14

U+FE0E U+FE0F				U+FE0E U+FE0F				U+FE0E U+FE0F				U+FE0E U+FE0F			
U+203C	!!	!!	!!	U+2600	☀️	☀️	☀️	U+2693	⚓	⚓	⚓	U+2733	✳️	✳️	✳️
U+2049	!?	!?	!?	U+2601	☁️	☁️	☁️	U+26A0	⚠️	⚠️	⚠️	U+2734	✳️	✳️	✳️
U+2139	❗	❗	❗	U+260E	☎️	☎️	☎️	U+26A1	⚡	⚡	⚡	U+2744	❄️	❄️	❄️
U+2194	↔️	↔️	↔️	U+2611	☑️	☑️	☑️	U+26AA	⦿	⦿	⦿	U+2747	✳️	✳️	✳️
U+2195	↕️	↕️	↕️	U+2614	☔️	☔️	☔️	U+26AB	⬤	⬤	⬤	U+274C	❌	❌	❌
U+2196	↖️	↖️	↖️	U+2615	☕️	☕️	☕️	U+26BD	⚽️	⚽️	⚽️	U+274E	❌	❌	❌
U+2197	↗️	↗️	↗️	U+261D	👉	👉	👉	U+26BE	⚾️	⚾️	⚾️	U+2753	❓	❓	❓
U+2198	↘️	↘️	↘️	U+263A	😊	😊	😊	U+26C4	🐼	🐼	🐼	U+2754	❓	❓	❓
U+2199	↙️	↙️	↙️	U+2648	♊️	♊️	♊️	U+26C5	☼	☼	☼	U+2755	❗	❗	❗
U+21A9	↺️	↺️	↺️	U+2649	♋️	♋️	♋️	U+26CE	♎️	♎️	♎️	U+2757	❗	❗	❗
U+21AA	↻️	↻️	↻️	U+264A	♌️	♌️	♌️	U+26D4	🚫	🚫	🚫	U+2764	❤️	❤️	❤️
U+231A	🕒	🕒	🕒	U+264B	♍️	♍️	♍️	U+26EA	🏛️	🏛️	🏛️	U+27A1	➡️	➡️	➡️
U+231B	🕒	🕒	🕒	U+264C	♎️	♎️	♎️	U+26F2	🍲	🍲	🍲	U+27BF	🔗	➡️	🔗
U+23E9	▶️	▶️	▶️	U+264D	♏️	♏️	♏️	U+26F3	🏌️	🏌️	🏌️	U+2934	↶️	↶️	↶️
U+23EA	◀️	◀️	◀️	U+264E	♐️	♐️	♐️	U+26F5	🚤	🚤	🚤	U+2935	↷️	↷️	↷️
U+23EB	⬆️	⬆️	⬆️	U+264F	♑️	♑️	♑️	U+26FA	🏠	🏠	🏠	U+2B05	⬅️	⬅️	⬅️
U+23EC	⬇️	⬇️	⬇️	U+2650	♒️	♒️	♒️	U+26FD	🗑️	🗑️	🗑️	U+2B06	⬆️	⬆️	⬆️
U+23F0	🕒	🕒	🕒	U+2651	♓️	♓️	♓️	U+2702	✂️	✂️	✂️	U+2B07	⬇️	⬇️	⬇️
U+23F3	🕒	🕒	🕒	U+2652	🌊	🌊	🌊	U+2705	✅	✅	✅	U+2B18	▀	▀	▀
U+25AA	▪️	▪️	▪️	U+2653	♈️	♈️	♈️	U+2708	✈️	✈️	✈️	U+2B1C	◻️	◻️	◻️
U+25AB	◻️	◻️	◻️	U+2660	♠️	♠️	♠️	U+2709	✉️	✉️	✉️	U+2B50	★	★	★
U+25B6	▶️	▶️	▶️	U+2663	♣️	♣️	♣️	U+270A	👉	👉	👉	U+2B55	⦿	⦿	⦿
U+25C0	◀️	◀️	◀️	U+2665	♥️	♥️	♥️	U+270B	👈	👈	👈	U+303D	⤴️	⤴️	⤴️
U+25FB	◻️	◻️	◻️	U+2666	♦️	♦️	♦️	U+270C	👉	👉	👉	U+3297	🎉	🎉	🎉
U+25FC	◼️	◼️	◼️	U+2668	🔥	🔥	🔥	U+270F	🖋️	🖋️	🖋️	U+3299	🎉	🎉	🎉
U+25FD	◻️	◻️	◻️	U+267B	♻️	♻️	♻️	U+2728	✨	✚	✨				
U+25FE	◼️	◼️	◼️	U+267F	♿️	♿️	♿️								

BPM Emojis variations with VS15 and VS16

Figure 1: Typical Cyrillic Small Letter Ve (boxed in black) and variant form (boxed in red). Source: Bible printed by Francysk Skaryna, Prague, circa 1519.

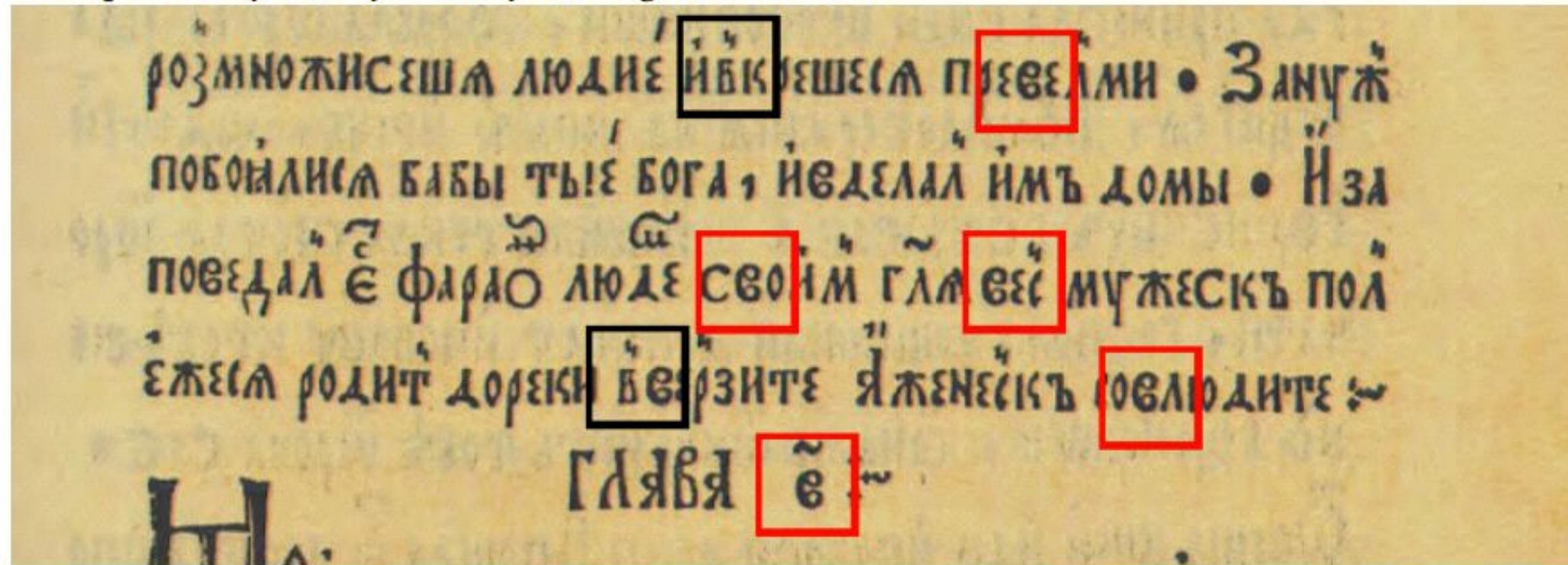


Figure 2: Typical Cyrillic Small Letter Ve (boxed in black) and variant form (boxed in red). Source: Bible printed by Francysk Skaryna, Prague, circa 1519.

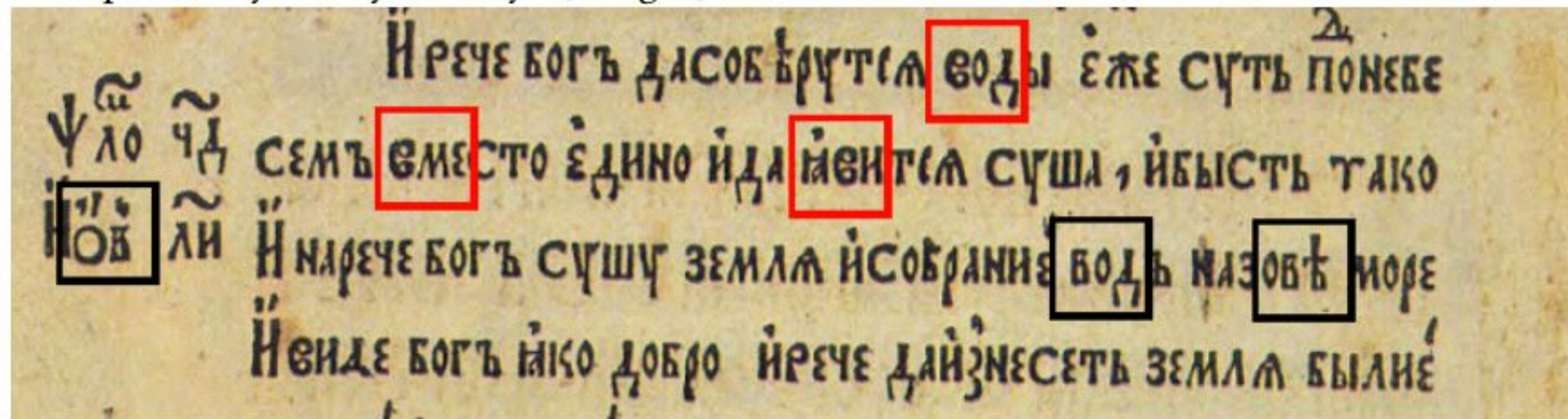





Table 1: Table of Proposed Variation Sequences				
Sequence	Glyph	Name	HIP	Belgrade
U+0432 U+FE00	ѡ	CYRILLIC SMALL LETTER VE VARIANT-1 ROUNDED VEDI	-	-
U+0432 U+FE0F	ѡ	CYRILLIC SMALL LETTER VE BASE FORM	в	E053
U+0434 U+FE00	ѡ̑	CYRILLIC SMALL LETTER DE VARIANT-1 LONG-LEGGED DOBRO	<д>	-
U+0434 U+FE0F	ѡ	CYRILLIC SMALL LETTER DE BASE FORM	д	E055
U+043E U+FE00	ѡ̣	CYRILLIC SMALL LETTER O VARIANT-1 NARROW ON	<о_>	E069
U+043E U+FE0F	ѡ	CYRILLIC SMALL LETTER O BASE FORM	о	E06A
U+0441 U+FE00	ѡ̑	CYRILLIC SMALL LETTER ES VARIANT-1 WIDE SLOVO	<с>	E167
U+0441 U+FE0F	ѡ̑	CYRILLIC SMALL LETTER ES BASE FORM	с	E06F
U+0442 U+FE00	ѡ̑	CYRILLIC SMALL LETTER TE VARIANT-1 TALL TVERDO	<т>	-
U+0442 U+FE01	ѡ̑	CYRILLIC SMALL LETTER TE VARIANT-2 OLD-STYLE TVERDO	< т >	-
U+0442 U+FE0F	ѡ̑	CYRILLIC SMALL LETTER TE BASE FORM	т	E070
U+044A U+FE00	ѡ̑	CYRILLIC SMALL LETTER HARD SIGN VARIANT-1 TALL HARD SIGN	<ѡ̑>	-
U+044A U+FE0F	ѡ̑	CYRILLIC SMALL LETTER HARD SIGN BASE FORM	ѡ̑	E080
U+0463 U+FE00	ѡ̑	CYRILLIC SMALL LETTER YAT VARIANT-1 TALL YAT	<ѡ̑>	-
U+0463 U+FE0F	ѡ̑	CYRILLIC SMALL LETTER YAT BASE FORM	ѡ̑	E086
U+A64B U+FE00	ѡ̑	CYRILLIC SMALL LETTER MONOGRAPH UK VARIANT-1 CHECKMARK-SHAPED UK	<ов>	-
U+A64B U+FE0F	ѡ̑	CYRILLIC SMALL LETTER MONOGRAPH UK BASE FORM	у	E072

Proposal to Use Standardized Variation Sequences to Encode Church Slavonic Glyph Variants in Unicode

Country Flags

0x1f1e6	+	0x1f1e7			
0x1f1e8	+	0x1f1f3			
0x1f1e9	+	0x1f1ea			
0x1f1ea	+	0x1f1f8			
0x1f1eb	+	0x1f1f7			
0x1f1ec	+	0x1f1e7			
0x1f1ee	+	0x1f1f9			
0x1f1ef	+	0x1f1f5			
0x1f1f0	+	0x1f1f7			
0x1f1f7	+	0x1f1fa			
0x1f1fa	+	0x1f1f8			

Unicode Common Locale Data Repository (CLDR) TR#35 (UTS)

Navigation

Unicode CLDR Project

CLDR Releases/Downloads

CLDR Survey Tool

CLDR Change Requests

CLDR Charts

CLDR Process

CLDR Specifications

Translation Guidelines

Unicode Extensions for BCP 47

Milestone Schedule

Date	Phase
2014-03-19	v25 Released
2014-09-18	v26 Released

See General Schedule

Internal Development

CLDR Development Site

New CLDR Developers

Handling Tickets (bugs/enhancements)

CLDR: Big Red Switch

Messages

Design Proposals

Direct Modifications to CLDR Data

Updating Codes

Updating DTDs

Editing the CLDR Spec

Sitemap

Copyright © 1991–2014 Unicode, Inc.
All Rights Reserved
[Terms of Use](#)

Unicode CLDR Project >

CLDR Releases/Downloads

Each release of the Unicode CLDR is a stable release and may be used as reference material or cited as a normative reference by other specifications. Each version, once published, is absolutely stable and will never change. Implementations may also apply [CLDR Corrigenda](#) to a release. Bug reports and feature requests for subsequent versions may be filed at [Bug Reports](#).

Downloads

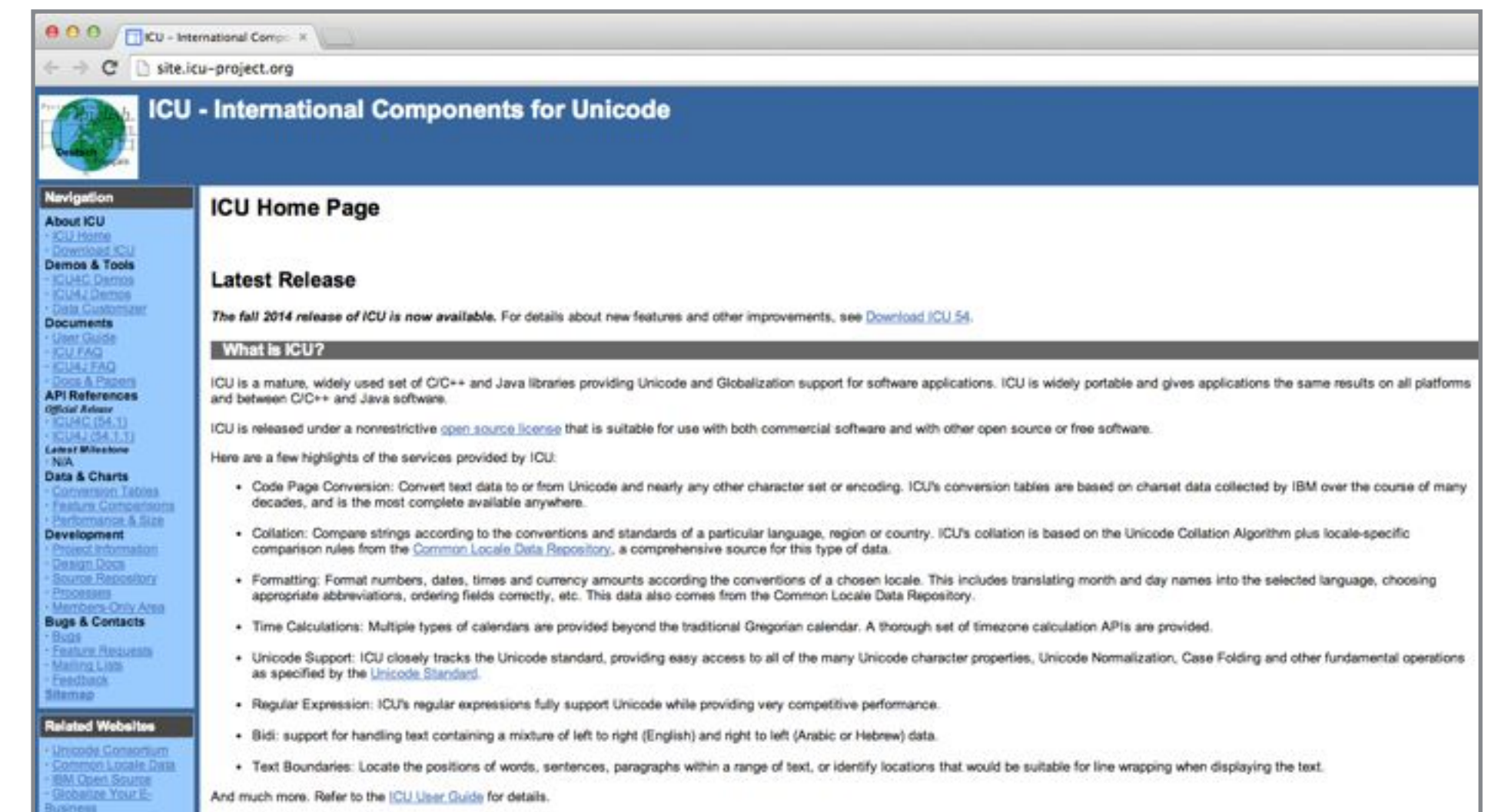
The following table lists the files for each released version. For license information, see the Unicode [Terms of Use](#); in particular, [Exhibit 1](#). The top two rows have permalinks for the latest version and the latest development version (snapshot). They are followed by the specific release versions.

No.	Date	Rel. Note	Data	Charts	Spec	Delta	SVN Tag	DTD Diffs
Latest		latest-version	latest-data	latest-charts	latest-ldml	latest-changes	latest	
Dev		dev-version	dev-data	dev-charts	dev-ldml	dev-changes	trunk	trunk
26	2014-09-18	v26	CLDR26	Charts26	LDML26	Δ26	release-26	ΔDTD26
25	2014-03-19	v25	CLDR25	Charts25	LDML25	Δ25	release-25	ΔDTD25
24	2013-09-18	v24	CLDR24	Charts24	LDML24	Δ24	release-24	ΔDTD24
23.1	2013-05-15	v23.1	CLDR23.1	Charts23.1	LDML23	Δ23.1	release-23-1	ΔDTD23.1
23	2013-03-15	v23	CLDR23	Charts23	LDML23	Δ23	release-23	ΔDTD23
22.1	2012-10-26	v22.1	CLDR22.1	Charts22.1	LDML22.1	Δ22.1	release-22-1	ΔDTD22.1
22	2012-09-10	v22	CLDR22	Charts22	LDML22	Δ22	release-22	ΔDTD22
21.0.2	2012-06-06	v21.0.2	via SVN		LDML21.0.1	Δ21.0.2	release-21-0-2	ΔDTD21.0.2
21.0.1	2012-03-21	v21.0.1	via SVN		LDML21.0.1	Δ21.0.1	release-21-0-1	ΔDTD21.0.1
21.0	2012-02-10	v21	CLDR21		LDML21	Δ21	release-21	ΔDTD21
2.0.1	2011-07-18	v2.0.1	CLDR2.0.1		LDML2.0.1	Δ2.0.1	release-2-0-1	ΔDTD2.0.1
2.0	2011-05-25	v2.0	CLDR2.0		LDML2.0	Δ2.0	release-2-0	ΔDTD2.0
1.9.1	2011-03-11	v1.9.1	CLDR1.9.1		LDML1.9	Δ1.9.1	release-1-9-1	ΔDTD1.9.1
1.9	2010-12-01	v1.9	CLDR1.9.0		LDML1.9	Δ1.9	release-1-9	ΔDTD1.9
1.8.1	2010-04-29	v1.8.1	CLDR1.8.1		LDML1.8.1	Δ1.8.1	release-1-8-1	ΔDTD1.8.1

- **Locale-specific patterns for formatting and parsing**
dates, times, timezones, numbers and currency values
- **Translations of names**
countries and regions, currencies, eras, months, weekdays, timezones, cities, time units, ...
- **Language & script information**
characters used; sorting & searching; writing direction; numbers spellings; segmentation, ...
- **Country information**
language usage, currency information, calendar preference and week conventions, ...

International Components for Unicode (ICU)

- Open-source project on top of CLDR
- Unicode text handling and regular expressions
character, word, and line boundaries
Language sensitive collation and searching
Normalization, upper and lowercase conversion
multi-calendar and time zones
parse and format dates, times, numbers, currencies
...



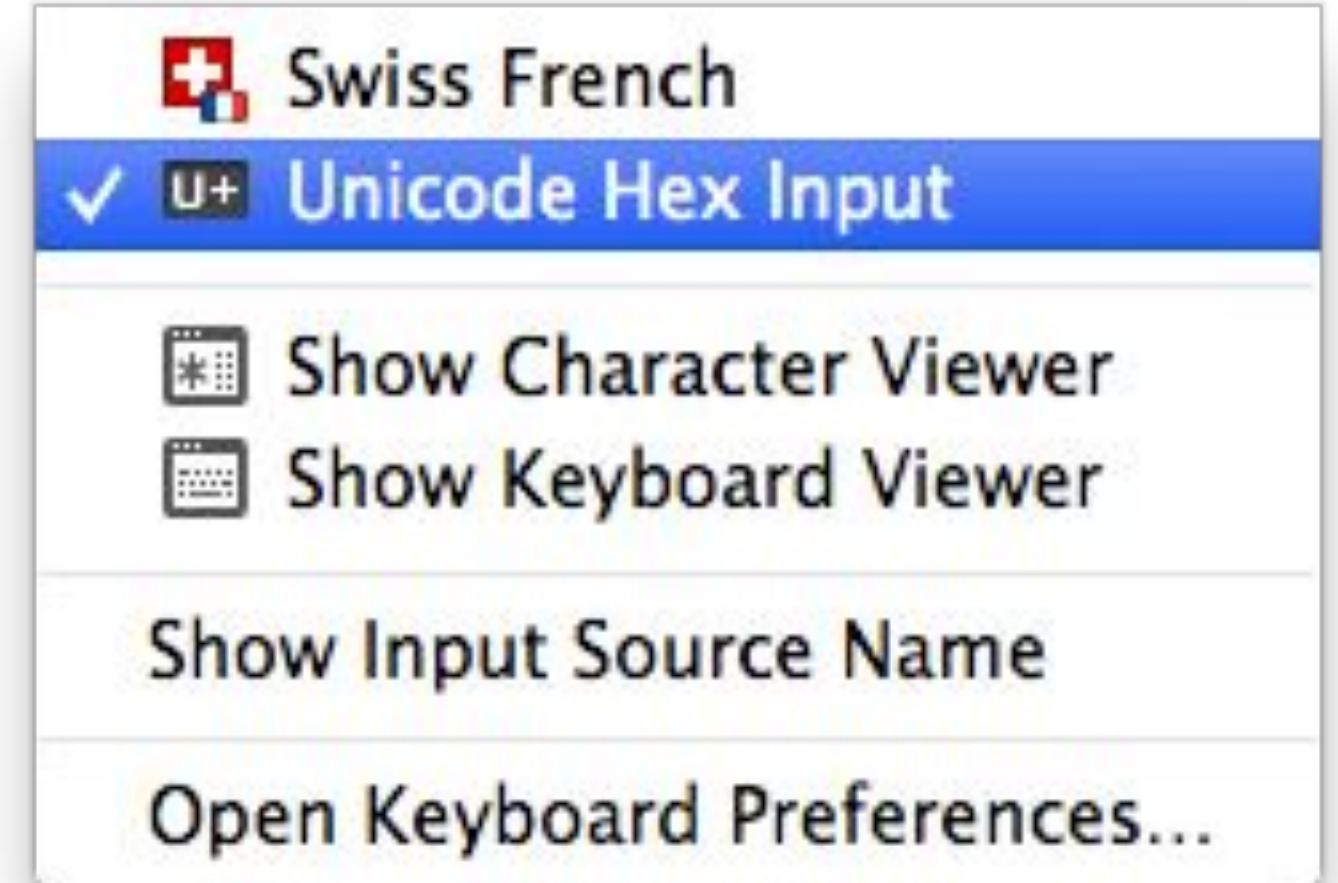
- Descends from Taligent (mid 1990s), which became part of IBM in 1996
- Included by Sun into JDK 1.1

More Specifications

- Text Segmentation TR#29 (UAX)
 - About when to words and lines, contextual
- Regular Expressions TR#18 (UTS)
- Bidirectional Algorithm TR#9 (UAX)
 - Arabic, Hebrew, ... display text from right to left but use left to right digits

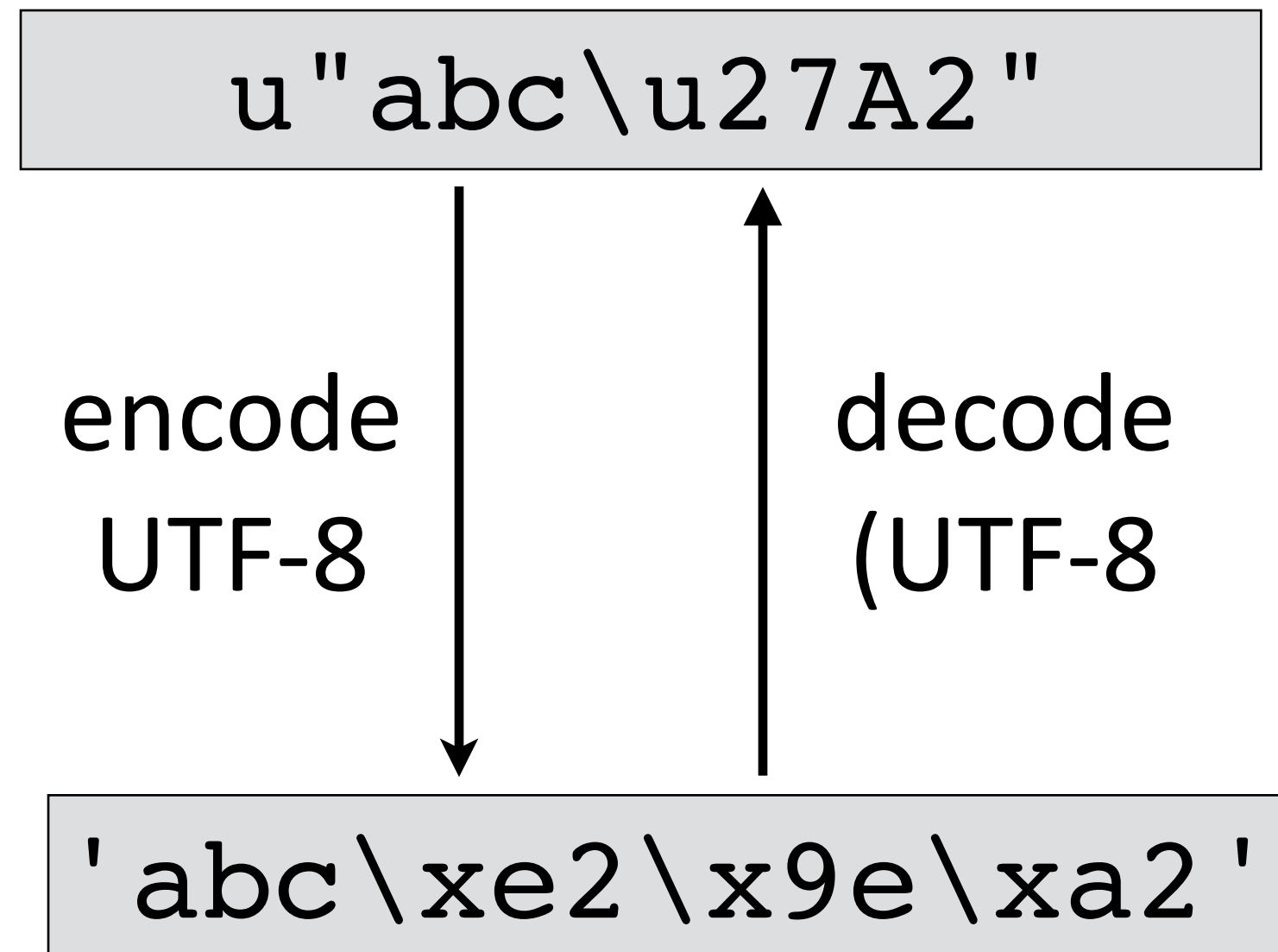
1. The Unicode Consortium
2. Selected Unicode Specifications
- 3. Unicode in Practice**
4. Unicode Hacks

OS X Unicode Hex Input alt XXXX (BMP only)



```
$ python3
>>> u = '\U0001F41B'
>>> print(u)
🐛
>>> import unicodedata
>>> unicodedata.name(u)
'BUG'
>>> u2 = unicodedata.lookup("BUG")
>>> print(u2)
🐛
```

Code Points \leftrightarrow Bytes



```
>>> u = u"abc\u27A2"  
>>> s = u.encode('utf-8')  
>>> s  
'abc\xe2\x9e\xa2'  
>>> u2 = s.decode('utf-8')  
>>> u2 == u  
True
```

C / C++

- Use `wchar_t*` ("wide char") instead of `char*`
Use the `wcs` functions instead of the `str` functions
`strcat` => `wcscat`
`strlen` => `wcslen`
- Convert `char` strings into `wchar_t` strings
`mbstowcs` multi byte string to wide char string
`wcstombs` wide char string to multi byte string
- Create a literal UCS-2 string:
`L"Hello"`

C

```
#include <stdio.h>
#include <locale.h>
#include <inttypes.h>

int main() {
    if (!setlocale(LC_CTYPE, "")) {
        fprintf(stderr, "Can't set the specified locale!\n");
        return 1;
    }

    wchar_t wc = 0x2190;

    printf("%ls %lc\n", L"Schöne Grüße \u2603", wc);

    return 0;
}
```

```
$ export LC_CTYPE=UTF-8
$ cc utf8.c
$ ./a.out
Schöne Grüße 🍵 ←
```

length of wchar_t (16 or 32 bits) is implementation-defined

Java

```
class Test {  
    public static void main (String[] argv) {  
        String s = "xxx \u2603";  
        System.out.println(s);  
    }  
}
```

```
$ javac Test.java
```

```
$ java -Dfile.encoding=UTF-8 Test
```

```
xxx 🍷
```

wide characters size is defined as 16 bits

Encoding Conversions

```
$ file utf8.txt  
utf8.txt: UTF-8 Unicode text
```

```
$ iconv -f utf8 -t utf-16le utf8.txt > utf-16le.txt
```

```
$ file latin1.txt  
latin1.txt: ISO-8859 text
```

Objective-C

```
NSString *s0 = @"A";  
NSString *s1 = @"\x61";  
NSString *s2 = @"\u2100";  
NSString *s3 = @"\U0001FF00";
```

```
NSString *s1 = @"\u2603";  
unichar uc = 0x2665;
```

```
NSLog(@"-- s1: %@ %C", s1, uc); // 🍷 ❤️
```

```
NSString *s2 = [NSString stringWithUTF8String:@"\xF0\x9D\x84\x9E"];  
NSLog(@"-- s2: %@", s2); // 🎵
```

```
NSData *data = [s2 dataUsingEncoding:NSUTF8StringEncoding];  
NSLog(@"-- data: %@", data); // <f09d849e>
```

Python 3

- ❌ Collation: still compare codepoints

```
>>> 'café' < 'caff'  
False
```

- ❌ Case Conversion restricted to 1:1 case mappings

```
>>> 'ß'.upper()  
'ß'
```

- ❌ Case conversion ignores locale

❌ Additionally, locale is global

```
>>> import locale  
>>> locale.setlocale(locale.LC_ALL, 'tr_TR')  
>>> s = "istanbul"  
>>> s.upper()  
'ISTANBUL'
```

Case Conversion – Locale

```
NSString *s = [NSString stringWithFormat:@"%istambul"];  
  
NSLocale *locale = [NSLocale localeWithLocaleIdentifier:@"%tr_TR"];  
  
NSString *s2 = [s uppercaseStringWithLocale:locale];  
  
// İSTAMBUL ✅
```

```
// U+1F600 GRINNING FACE
NSArray *a = @[@"A", @"\U0001F600", @"B"];
```

```
[a enumerateObjectsUsingBlock:^(NSString *s, NSUInteger idx, BOOL *stop) {
    NSLog(@"[%lu] %@\n", idx, s);
}];
```

```
/*
[0] A
[1] 😄
[2] B
*/
```



```
[a enumerateObjectsUsingBlock:^(NSString *s, NSUInteger idx, BOOL *stop) {
    NSLog(@"[%lu] %C\n", idx, [s characterAtIndex:0]);
    // idx == 1, s = [0xD83D, 0xDE00], and U+D83D is a high surrogate
}];
```

```
/*
[0] A
[2] B
*/
```



Swift

```
$ xcrun swift
1> import Foundation
2> var s1 = "ni\u{00F1}o" // precomposed
s1: String = "niño"
3> var s2 = "nin\u{0303}o" // decomposed
s2: String = "niño"
4> s1 == s2 // canonical equality
$R0: Bool = true
5> s1.isEqual(s2) // different bytes
$R1: Bool = false
```

Regex

```
$ python3
>>> import re
>>> reg = re.compile("\d")
>>> gen = ( chr(c) for c in range(0, 0xFFFF) if re.match(reg, chr(c)) )
>>> print(''.join(gen))
0123456789.\|_!@#%&'()*+,-./:;<=>?[\]^_`{|}~¡¢£¥¦§¨ª«¬®¯°±²³´µ¶·¸¹º»¼½¾¿ÀÁÂÃÄÅÆÇÈÉÊËÌÍÎÏÐÑÒÓÔÕÖ×ØÙÚÛÜÝÞßàáâãäåæçèéêëìíîïðñ
r s t u v w x y z [ \ ] ^ _ ` { | } ~ ¡ ¢ £ ¥ ¦ § ¨ © ª « ¬ ® ¯ ° ± ² ³ ´ µ ¶ · ¸ ¹ º » ¼ ½ ¾ ¿ À Á Â Ã Ä Å Æ Ç È É Ê Ë Ì Í Î Ï Ð Ñ Ò Ó Ô Õ Ö × Ø Ù Ú Û Ü Ý Þ ß à á â ã
ä å æ ç è é ê ë ì í î ï ð ñ ò ó ô õ ö ÷ ø ù ú û ü ý þ ÿ 0 1 2 3 4 5 6 7 8 9
>>> reg = re.compile("\d", re.ASCII)
```

Regex

```
$ jsc
>>> /a.c/.test('abc')
true
>>> /a.c/.test('a🐛c')
false
>>> /a...c/.test('a🐛c')
true
```

How well do you know your tools?

- illegal code points
- length? (code points? bytes?)
- equality, equivalence, norm.
- reversing strings
- character at index
- iterating over all symbols
- substring
- regex
- bi-directional text
- text segmentation

1. The Unicode Consortium
2. Selected Unicode Specifications
3. Unicode in Practice
- 4. Unicode Hacks**



+ Follow

Yo **Unicode** I herd U like *typefaces* so we put some codepoints in your Supplementary Multilingual Plane so you can *encode* fonts in your *fonts*.

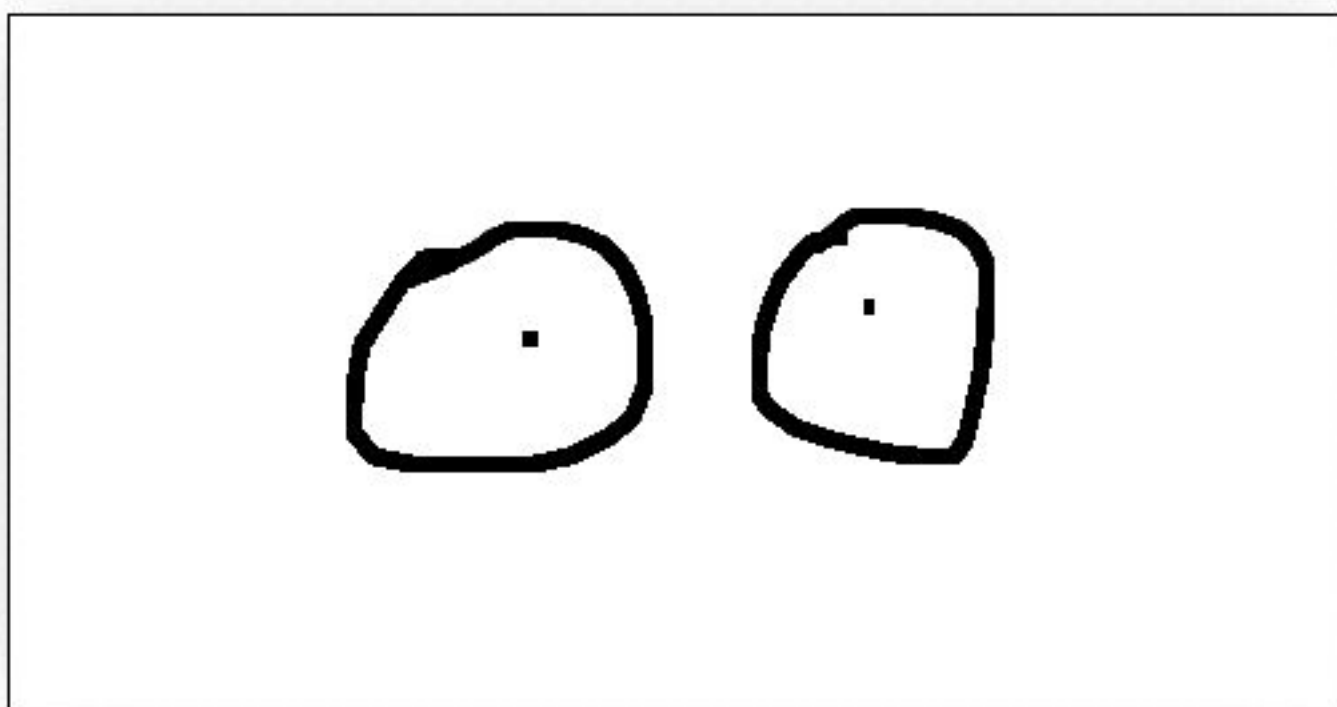


RETWEETS

823

10:03 AM - 14

drawbox



Draw something in the left box!

And let shapecatcher help you to find the most similar unicode characters!
Currently, there are 11817 unicode character glyphs in the database. Japanese, Korean and Chinese characters are currently not supported.

195

3,069

2.7k

Flattr

Tweet

Like

Share

Follow @Shapecatcher

→ Recognize ✖ Clear



Domino tile horizontal-01-01: ☐

Unicode hexadecimal: 0x1f039
In block: [Domino Tiles](#)
Rate this suggestion: [good](#) | [bad](#)
[More Info](#)

score: 0.843336



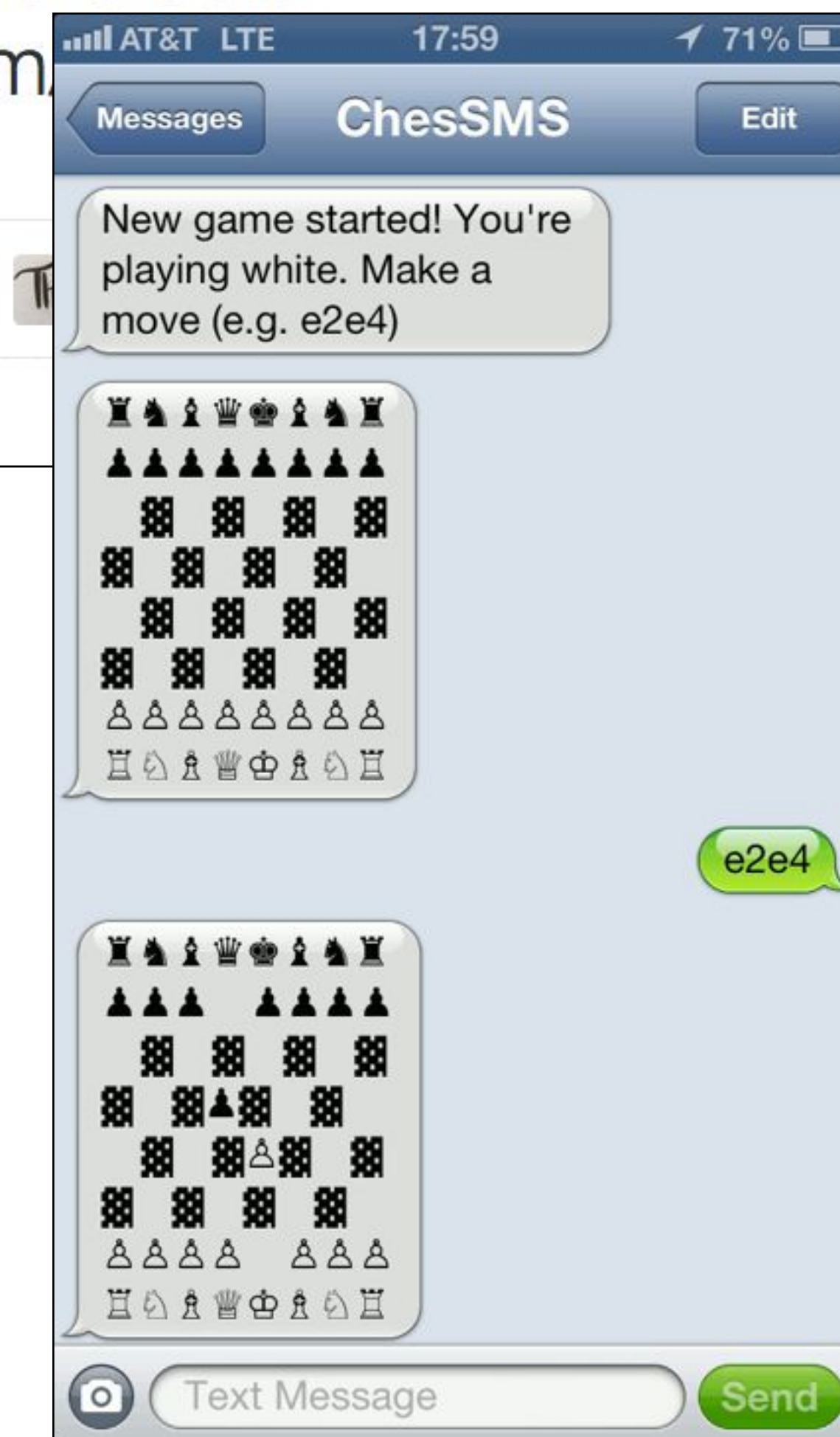
Wall Street Journal

@WSJ



+ Follow

████████████████████ Last 12 months of the U.S. unemployment rate, which rose to 9% in April. More data: <http://on.wsj.com>





Nicolas Seriot
@nst021

```
$ python unibinary.py -s "嫠韋罔哢一七一—  
北——俚—予——丐——佂—丸——剿印哆  
嶺啖嶸哆刊丐市䟽———佂—È丄僂——噴市  
嶠市市È丄✖丄乐—丁——丁——佂—丐崙市  
嶠仆嚶——聿倨么夙乌宀罍刑聿佂伟崙——✖  
丁丐市咀—下丁譚姿嫫仿✖丄—仿䟽—✖✖" >  
m
```



RETWEETS

3

FAVORITE

1

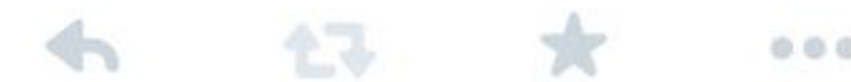


8:29 PM - 17 Jan 2013



Nicolas Seriot
@nst021

```
$ chmod +x m  
$ ./m  
Hello world
```



RETWEETS

2

FAVORITE

1



8:29 PM - 17 Jan 2013

Pack 289+ ASCII chars or 209+ bytes into 140 characters.

<https://github.com/nst/UniBinary>

Unicode Security



« Unicode is just too complex to ever be secure. »
– Bruce Schneier, 2000

<https://www.schneier.com/crypto-gram-0007.html#9>

- TR#36 Unicode Security Considerations
- TR#39 Unicode Security Mechanisms
- Chris Weber's <http://websec.github.io/unicode-security-guide/>

Illegal Sequences

- Illegal UTF-8 sequences include:

- overlong encoding

1	1	0	0	0	0	0	0	1	0	0	0	0	0	0	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

0xC0 0x41


- unexpected continuation byte

1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

0xC0 0x00

- Illegal UTF-16 sequences include unpaired surrogates such as:
 - [0xD800–0xDBFF] not followed by [0xDC00–0xFFFF]
 - [0xDC00–0xFFFF] not preceded by [0xD800–0xDBFF]

Exploiting Transformations

- Exploitation of normalization to add / remove characters and bypass filters
- Non-characters: U+FFFE, U+FFFF, U+1FFFE, U+1FFFF, U+10FFFE, U+10FFFF
- Non-character code points must not be simply deleted (as allowed by Unicode < 5.2 C7) but replaced by  U+FFFD REPLACEMENT CHARACTER.

``

- Unassigned code points (eg. U+2073)



Posted on [June 18, 2013](#) by [Mikael Goldmann](#)

Creative usernames and Spotify account hijacking

Pwning an account

A bunch of us dropped whatever we were working on and scurried to try to understand what was going wrong and how to fix it. From the forum post we knew that taking over an account went something like this:

1. Find a user account to hijack. For the sake of this example let us hijack the account belonging to user bigbird.
2. Create a new spotify account with username `BIGBIRD` (in python this is the string `u'\u1d2e\u1d35\u1d33\u1d2e\u1d35\u1d3f\u1d30'`).
3. Send a request for a password reset for your new account.
4. A password reset link is sent to the email you registered for your new account. Use it to change the password.
5. Now, instead of logging in to account with username `BIGBIRD`, try logging in to account with username bigbird with the new password.
6. Success! Mission accomplished.

From the log lines associated with the hijacking of the forum manager's account it appeared to be a problem with how we derived a *canonical username* from the username the user chooses at registration, but we were still pretty much in the dark. We had no option except to disable account creation until we could prevent the attack.

<https://labs.spotify.com/2013/06/18/creative-usernames/>

Visual Spoofing

AA A A Δ A A

www.google.com – U+0067 LATIN SMALL LETTER G

www.google.com – U+0261 LATIN SMALL LETTER SCRIPT G

8 – U+09EA BENGALI DIGIT FOUR

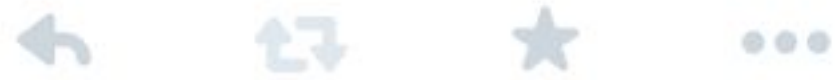
9 – U+0B68 ORIYA DIGIT TWO



Nicolas Seriot

@nst021

Here is a nice little Core Text crasher for OS X:
\$ python -c "print u'\u0647\u0020\u0488\u0488\u0488'"



RETWEETS

38

FAVORITES

34



10:49 AM - 25 Mar 2013

```
$ gdb Twitter
```

```
(gdb) r
```

```
Starting program: /Applications/Twitter.app/Contents/MacOS/Twitter
```

```
Program received signal EXC_BAD_ACCESS, Could not access memory.
```

```
Reason: KERN_INVALID_ADDRESS at address: 0x00000001084e8008
```

```
0x00007fff9432ead2 in vDSP_sveD ()
```

```
(gdb) bt
```

```
#0 0x00007fff9432ead2 in vDSP_sveD ()
```

```
#1 0x00007fff934594fe in TStorageRange::SetStorageSubRange ()
```

```
#2 0x00007fff93457d5c in TRun::TRun ()
```

```
#3 0x00007fff934579ee in CTGlyphRun::CloneRange ()
```

```
#4 0x00007fff93466764 in TLine::SetLevelRange ()
```

```
#5 0x00007fff93467e2c in TLine::SetTrailingWhitespaceLevel ()
```

```
#6 0x00007fff93467d58 in TRunReorder::ReorderRuns ()
```

```
#7 0x00007fff93467bfe in TTypesetter::FinishLineFill ()
```

```
#8 0x00007fff934858ae in TFramesetter::FrameInRect ()
```

```
#9 0x00007fff93485110 in TFramesetter::CreateFrame ()
```

```
#10 0x00007fff93484af2 in CTFramesetterCreateFrame ()
```

```
...
```



MAIN MENU MY STORIES: 25 FORUMS SUBSCRIBE JOBS

INFINITE LOOP / THE APPLE ECOSYSTEM

Rendering bug crashes OS X, iOS apps with string of Arabic characters (Updated)

CoreText bug crashes any iOS 6 and OS X programs that use the API.

by Andrew Cunningham and Dan Goodin Aug 29 2013, 9:30pm CEST

Share Tweet 149

LATEST FEATURE



FEATURE STORY (1 PAGE)

The Register®

Biting the hand that feeds IT

Data Centre Software Networks Security Business Hardware Science Bootnotes Video Forums Weekend Edition

Operating Systems Applications Developer Verity Stob

SOFTWARE > OPERATING SYSTEMS

Anatomy of a killer bug: How just 5 characters can murder iPhone, Mac apps

What evil lurks in the Unicode of Death ... oh, a buffer overrun

By Chris Williams, 4 Sep 2013

93

RELATED STORIES

Analysis There has been much sniggering into sleeves after wags found they could upset iOS 6 iPhones and iPads, and Macs running OS X 10.8, by sending a simple rogue text message or email.

A bug is triggered when the CoreText component in vulnerable Apple operating

MOST READ MOST COMMENTED

YARR! Pirates walk the plank: DMCA magnets sink in Google results

Whisper tracks its users. So we tracked down its LA office. This is what happened next

Xperia Z3: Crikey, Sony – ANOTHER flagship phondleslab?

Ex-US Navy fighter pilot MIT prof: Drones beat humans - I should know

Apple flings iOS 8.1 at world+dog: Our AMAZEBALLS 9-step installation guide

U+202E RIGHT-TO-LEFT OVERRIDE

```
$ python3 -c "print('ABC\u202EDEF')"  
ABCFED  
# copy-paste gets crazy
```

```
$ python3 -c "print('x\u202Efdp.doc')"  
xcod.pdf  
# double click a .pdf, open a .doc
```

HFS+

Important: The terms used in this Q&A, precomposed and decomposed, roughly correspond to Unicode Normal Forms C and D, respectively. However, most volume formats do not follow the exact specification for these normal forms. For example, HFS Plus (Mac OS Extended) uses a variant of Normal Form D in which U+2000 through U+2FFF, U+F900 through U+FAFF, and U+2F800 through U+2FAFF are not decomposed (this avoids problems with round trip conversions from old Mac text encodings). It's likely that your volume format has similar oddities.

Apple Technical Q&A QA1173

- Terminal.app (and most apps) output NFC UTF-8.
- The filenames you write are different from the ones you read.

HFS+

```
$ echo ü; echo ü | xxd
ü
00000000: c3bc 0a # NFC
$ touch ü; ls; ls | xxd
ü
00000000: 75cc 880a # NFD
```

```
$ touch "Bücher"
$ ls Bü<TAB> # no completion
$ ls Bu<TAB> # completion
```

OS X Bash

```
$ mkdir /tmp/test
$ cd /tmp/test
$ touch `printf « a\xef\xbb\xbf`^
# or "a\uFEFFb".encode('utf-8')
$ ls a*
a?b
$ touch ab
$ ls a*
a?b
# where did ab go?!
```

OS X Finder

```
$ echo -e "\xFF\xFE" > x.txt # UTF-16LE BOM
$ xattr -w com.apple.TextEncoding "utf-16le" x.txt
$ qlmanage -p x.txt # or QuickLook with Finder
```

```
[ERROR] An uncaught exception was raised outside of any generator: *** -[NSConcreteTextStorage attribute:atIndex:longestEffectiveRange:inRange:]: Range or index out of bounds
2014-10-24 10:53:08.474 qlmanage[5268:11f] *** Terminating app due to uncaught exception 'NSRangeException', reason: '*** -[NSConcreteTextStorage attribute:atIndex:longestEffectiveRange:inRange:]: Range or index out of bounds'
*** First throw call stack:
(
    0   CoreFoundation          0x00007fff89ebe25c __exceptionPreprocess + 172
    1   libobjc.A.dylib          0x00007fff87934e75 objc_exception_throw + 43
    2   CoreFoundation          0x00007fff89ebe10c +[NSException raise:format:] + 204
    3   AppKit                  0x00007fff81a83a7a -[NSConcreteTextStorage attribute:atIndex:longestEffectiveRange:inRange:] + 118
    4   AppKit                  0x00007fff81951ded -[NSMutableAttributedString(NSMutableAttributedStringKitAdditions) fixGlyphInfoAttributeInRange:] + 204
    5   AppKit                  0x00007fff81951cd8 -[NSMutableAttributedString(NSMutableAttributedStringKitAdditions) fixAttributesInRange:] + 39
    6   AppKit                  0x00007fff81a838e1 -[NSTextStorage processEditing] + 109
    7   AppKit                  0x00007fff81a7f742 -[NSTextStorage endEditing] + 110
    8   AppKit                  0x00007fff81c5db4f _NSReadAttributedStringFromURLOrData + 14525
    9   AppKit                  0x00007fff81c5e3a5 -[NSAttributedString(NSAttributedStringKitAdditions) initWithURL:options:documentAttributes:
```

```
# watch your Finder go nuts!!!
$ cd; touch `printf "\x41\xe9"`
# NFC( "Aé" )
$ open .
# fixed in OS X 10.10
```

Conclusion

- Unicode is cool. Unicode is hard.
- Everything dealing with Unicode is a bug nest.
- You cannot just ignore Unicode, you're using it.
- Most APIs should use strings instead of a single char.



seriot.ch

twitter.com/nst021

linkedin.com/in/nseriot