ASCII and Unicode quotation marks

by Markus Kuhn

Summary: Please do not use the ASCII grave accent (0x60) as a left quotation mark together with the ASCII apostrophe (0x27) as the corresponding right quotation mark (as in `quote'). Your text will otherwise appear rather strange with most modern fonts (e.g., on Windows and Mac systems). Only old X Window System fonts and some old video terminals show ASCII 0x60/0x27 as left and right quotation marks, while most modern systems follow the ISO and Unicode standards instead. If you can use only ASCII's typewriter characters, then use the apostrophe character (0x27) as both the left and right quotation mark (as in 'quote'). If you can use Unicode characters, nice directional quotation marks are available in the form of characters U+2018, U+2019, U+201C, and U+201D (as in 'quote' or "quote").

Background

The Unicode and ISO 10646 standards define the following characters:

U+0022	QUOTATION MARK	"	neutral (vertical), used as opening or closing quotation mark; preferred characters in English for paired quotation marks are U+201C and U+201D
U+0027	APOSTROPHE	'	neutral (vertical) glyph having mixed usage; preferred character for apostrophe is U+2019; preferred characters in English for paired quotation marks are U+2018 and U+2019
U+0060	GRAVE ACCENT	`	
U+00B4	ACUTE ACCENT	´	
U+2018	LEFT SINGLE QUOTATION MARK	'	
U+2019	RIGHT SINGLE QUOTATION MARK	'	this is the preferred character to use for apostrophe
U+201C	LEFT DOUBLE QUOTATION MARK	"	
U+201D	RIGHT DOUBLE QUOTATION MARK	"	

ASCII and ISO 8859 were only designed to support the very restricted typographic style available to typewriter users. The two ASCII characters

0x22	QUOTATION MARK	"
0x27	APOSTROPHE	'

are supposed to represent the neutral (vertical) glyphs commonly used on typewriters. They should **not** be used as directional quotation marks.

ISO 8859 and Unicode fonts are supposed to show the two accent characters

| 0x60 | GRAVE ACCENT | ` |
| 0xB4 | ACUTE ACCENT | ´ |

as mutually symmetric shapes.

The problem

Unfortunately, the X Window System fonts contained for a long time the following mutually symmetric glyphs:

| 0x27 | APOSTROPHE | ' |
| 0x60 | GRAVE ACCENT | ` |

These shapes were even sanctioned by an early US version of the ISO 646 standard (ANSI X3.4, also known as ASCII), which defined 0x27 as "apostrophe (closing single quotation mark; acute accent)", but they should already have been changed when the fonts were extended to cover ISO 8859-1, which added a separate acute accent at 0xB4. One obviously cannot have both 0x27/0x60 and 0x60/0xB4 as mutually symmetric glyph pairs and have at the same time a different shape for 0x27 and 0xB4. Since 0x60/0xB4 were defined to be accents by the modern standards, their symmetric shape got priority, except that this had not been fixed in the X fonts until 2004 (somewhat earlier in the versions that come with XFree86).

The old X fonts encouraged some authors of Unix software and documentation to abuse 0x60 together with 0x27 as directional quotation marks. This practice looked somewhat acceptable like

> ‛ quotation ’

if displayed with old X fonts, but it looked rather ugly like

> ˋ quotation ˈ

in most other modern display environments (e.g., with the correctly designed Windows and Mac TrueType fonts, but also on many classic 1970s/1980s video terminals, such as those by Siemens/Nixdorf and many other manufacturers).

For example, 0x60 and 0x27 look under Windows NT 4.0 with the TrueType font Lucida Console (size 14) like this:
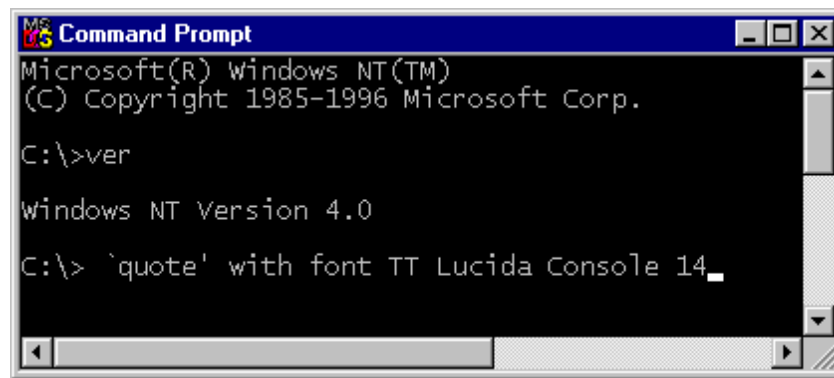
Unicode and ISO 10646 make a very clear distinction between the undirected typewriter-style ASCII single quotation mark and apostrophe U+0027 as in

<p style="text-align:center">' quotation '</p>

and the typographic directed quotation marks U+2018 and U+2019 as in

<p style="text-align:center">‘ quotation ’</p>

Unicode 2.1 explicitly says that U+2019 is the preferred punctuation apostrophe, as in "We've been here before." The Unicode standard also notes:

> "For historical reasons, U+0027 is a particularly overloaded character. In ASCII it is used to represent a punctuation mark (such as right single quotation mark, left single quotation mark, apostrophe punctuation, vertical line, or prime) or a modifier letter (such as apostrophe modifier or acute accent.) (Punctuation marks generally break words; modifier letters generally are considered part of a word.) In many systems it is always represented as a straight vertical line and can never represent a curly apostrophe or right quotation mark."

What to do?

If you are the author of some Unix software, then please check, whether you use the ASCII character 0x60 (`) as a left quotation mark as in `quote'. Change it such that you use instead the character 0x27 (') on both sides, as in 'quote'. If you work in an environment where the UTF-8 encoding is already used everywhere (e.g., Plan9 and most modern GNU/Linux installations), you could even decide to use proper directional quotation marks, as in ‘quote’ or "quote".

Check your source code directories with

```
grep \` *
```

to find out, where modifications are necessary. Then use (with proper care!) something like

```
perl -pi.bak -e "s/\`/'/g;" file1 file2 ...
```

to make the necessary substitutions automatically, or make the edits manually instead.

The use of 0x60 (grave accent) as a special control character in the Unix shell (to denote command

substitution as in `command` or better $(command)), in Perl, in Lisp, or in TeX/troff (to denote a proper left single quotation mark) does not have to be changed and remains unaffected. Donald Knuth's TeXbook (chapter 2, page 3, end of second paragraph) has actually warned TeX users already since 1986 that the apostrophe and grave accent shapes can show up as required by ISO and Unicode and not as used in the rest of the TeXbook. The Unix m4 macro processor is probably the only widely used tool that uses the `quote' combination as part of its input syntax; however, even that could be modified via changequote.

Why should we fix this?

There are quite a number of reasons, why the old X fonts had to be fixed, and with them the associated ASCII backquote practice:

- Obviously, grave accent and acute accent have to be mutually symmetric, which was not the case in the old X fonts.

- The Unicode 4.0 standard says explicitly that U+0027 be a "neutral (vertical) glyph having mixed usage" and shows the entire ASCII section like this:

- The ISO 10646, ISO 8859 and ISO 646/ECMA-6 standards also show the vertical typewriter apostrophe for U+0027 and have U+0060 and U+00B4 as mutually symmetric accents.

- The code table in ANSI X3.4:1986 ("ASCII"), which has been printed using the OCR-B font, also shows the vertical typewriter apostrophe. Historically, the originally proposed use of 0x60 in the international 7-bit coded character set was as a grave accent (ISO TC 97/SC 2 meeting, October 29-31, 1963), and only later its meaning was extended in the US implementation of the standard to also cover the use as a left single quotation mark (CACM 8(4)207-214, 1965).

- Most European keyboards have keycap labels for the apostrophe and both accents. These have

always looked like in the ISO and Unicode standards. The photo below shows the relevant keys highlighted on a standard German PC keyboard, which has the acute/grave accent key left and the number-sign/apostrophe key below the backspace key:

It can cause quite some confusion for users, if the keycap labels and the glyph shapes in the fonts disagree, as they did in the old X fonts.

- Microsoft and Apple fonts also follow the modern standards and disagree with the old X fonts. X11 users really should not be mislead about how the characters they use will appear on other standards conforming systems. Otherwise they will not realize that for example every user of a Windows web browser (screenshot: Internet Explorer 5) sees "backquotes" as in
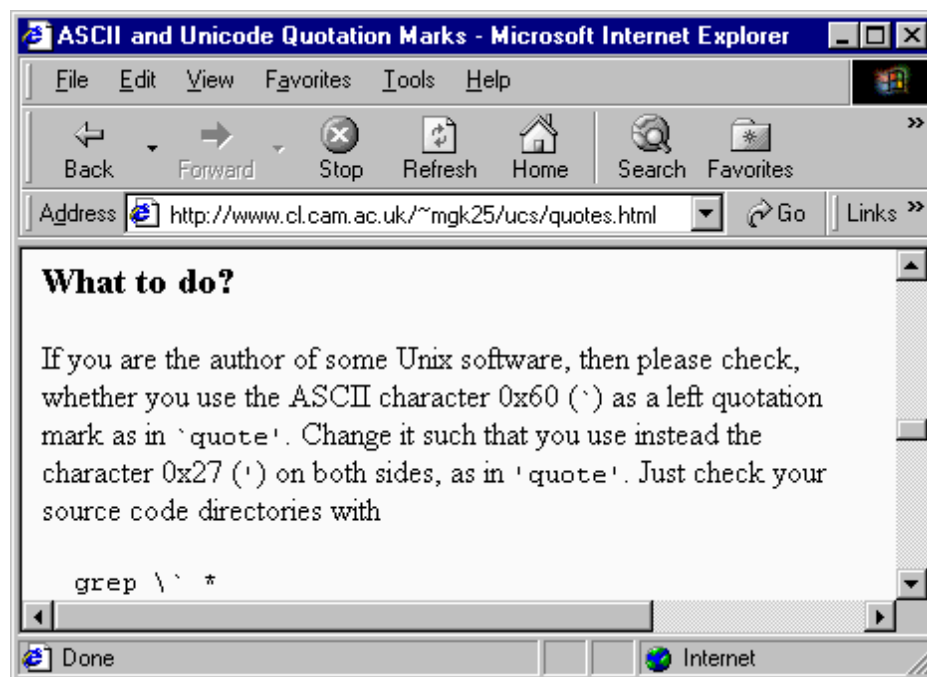
- Since XFree86 4.0 added TrueType font support, users of GNU/Linux systems have increasingly used modern fonts with the straight 0x27 glyph, and get funny quotation marks

with older software that tries to do show directional quotation marks with ASCII (most notably various GNU packages).

- The characters 0x27 (apostrophe) and 0x22 (quotation mark) are often used to abbreviate minutes and seconds or feet and inches, which is yet another reason, why 0x27 should just be a single-stroke version of 0x22, and not a curly directional quotation mark.

Updated X Window System core BDF fonts have been available since 1998, in which the apostrophe and grave accent are now corrected, along with a number of other bugs. They replaced the old fonts in XFree86 since version 4.0 and in the X.Org sample implementation since X11R6.8.

Related hints

PostScript

PostScript has a somewhat complicated history of how it maps the ASCII bytes to glyphs. In PostScript fonts, each glyph is identified not by a code position, but by a *glyph name* such as "quotesingle". After the publication of the Unicode Standard, Adobe released an official PostScript Glyph Name to Unicode Mapping table. When a PostScript interpreter displays text, it uses an *encoding vector* to map the 8-bit byte values found in text strings onto the glyph names found in fonts.

Unicode		glyph image	PostScript			
			glyph name	encoding vector		
position	name			Std	ISOLatin1	CE
U+0022	QUOTATION MARK	"	quotedbl	0x22	0x22	0x22
U+0027	APOSTROPHE	'	quotesingle	0xA9	—	0x27
U+0060	GRAVE ACCENT	`	grave	0xC1	0x91	0x60
U+00B4	ACUTE ACCENT	´	acute	0xC2	0x92/0xB4	0xB4
U+2018	LEFT SINGLE QUOTATION MARK	'	quoteleft	0x60	0x60	0x91
U+2019	RIGHT SINGLE QUOTATION MARK	'	quoteright	0x27	0x27	0x92
U+201C	LEFT DOUBLE QUOTATION MARK	"	quotedblleft	0xAA	—	0x93
U+201D	RIGHT DOUBLE QUOTATION MARK	"	quotedblright	0xBA	—	0x94

PostScript provides several predefined 8-bit encoding vectors. Authors of printer drivers can easily add their own. As the above table shows, the original PostScript standard encoding followed a practice similar to the old X fonts, with all its problems, namely it mapped the ASCII bytes 0x60 and 0x27 to curly opening and closing quotation marks ("quoteleft" and "quoteright" in PostScript glyph-name terminology, or U+2018 and U+2019 in Unicode).

When ISO 8859-1 emerged, Adobe added to PostScript another predefined encoding vector called `ISOLatin1Encoding`. This was meant to be ISO 8859-1 compatible, but it remained at 0x60 and 0x27 unchanged from the old `StandardEncoding` vector, and therefore it does not actually print the ISO 8859-1 characters 0x27 and 0x60 correctly, which correspond to Unicode characters U+0027 and U+0060 and should be represented by the PostScript glyphs "grave" and "quotesingle". The authors of Adobe's PostScript Language Reference, Third Edition (Addison-Wesley, ISBN 0-201-37922-8) acknowledge this in section E.5, footnote 3, page 783, where they note that the "`ISOLatin1Encoding` encoding vector deviates from the ISO 8859-1 standard" and that an application that wants to "conform exactly to the ISO standard should create a modified encoding vector". The newer CE encoding vector (Central European, matching Windows CP1250), which is now also described in the PostScript Language Reference, correctly maps 0x27 to "quotesingle" and 0x60 to "grave".

If you write a PostScript driver, please use the official Unicode to PostScript mapping table to map ASCII, ISO 8859 and ISO 10646 characters to PostScript glyphs, as the updated Type 1 renderer in XFree86 4.0 does. Do not use the `ISOLatin1Encoding` encoding vector to print ISO 8859-1 text, without changing it first to map 0x27 to "quotesingle" and 0x60 to "grave". (In addition, you may also want to map 0x2D = HYPHEN-MINUS to the PostScript glyph "hyphen" instead of the "minus" mapping used by `ISOLatin1Encoding`).

TeX

The font `cmtt10` in TeX's Computer Modern family follows the example of the PostScript standard encoding by providing a straight double quotation mark and directional single quotation marks on the ASCII positions 0x22, 0x60, and 0x27. It also provides a straight single quotation mark, grave accent, and acute accent on code positions 0x0d, 0x12, and 0x13, respectively, but it lacks directional double quotation marks:

U+0022 QUOTATION MARK	`"`	"
U+0027 APOSTROPHE	`\char"0D`	'
U+0060 GRAVE ACCENT	`\char"12`	`
U+00B4 ACUTE ACCENT	`\char"13`	´
U+2018 LEFT SINGLE QUOTATION MARK	`` ` ``	'
U+2019 RIGHT SINGLE QUOTATION MARK	`'`	'

Therefore, to demonstrate the result of abusing ASCII's straight quotation mark and graph accent as directional quotation marks in a document written in LaTeX, you can write `\texttt{\char"12 quote\char"0D}`. The non-typewriter fonts in Computer Modern lack both single and double straight quotation marks.

Use LaTeX's upquote package (`\usepackage{upquote}`) to map in the verbatim modes the ASCII characters 0x27 and 0x60 to the correct glyphs.

References

- Michael Everson: On the apostrophe and quotation mark, with a note on Egyptian transliteration characters, Working Group Document ISO/IEC JTC1/SC2/WG2 N2043, 1999-07-24

- Adobe: Unicode and Glyph Names, 1997–2003.

- UTF-8 and Unicode FAQ for Unix/Linux

- Unicode fonts and tools for X11

- Bruno Haible explains how to output nice Unicode quotation marks in a portable way using GNU gettext.

- The Unicode Standard, Version 4.0, Addison-Wesley, 2003, ISBN 0321185781.

- Jukka Korpela: Character histories: notes on some ASCII code positions.

- Markus Kuhn: Apostrophe and acute accent confusion. This is a page on the frequent error of misusing the U+00B4 or U+0060 acute and grave accent as an apostrophe instead of the appropriate apostrophe character U+0027 or better U+2019. This is today a frequent mistake, made by users of German, Swedish, Spanish and other PC keyboards, where the acute accent key is easier to reach than the (shifted) apostrophe key. The acute/grave key should always be non-spacing, to make it less likely that it is misused for entering wrong apostrophes.

- David A. Wheeler: Curling Quotes in HTML, SGML, and XML.

created 1999-12-19 – last modified 2007-12-11 – http://www.cl.cam.ac.uk/~mgk25/ucs/quotes.html