

Horribly overdue update to unicode.txt

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2003-10/0924.html>

From: H. Peter Anvin (hpa_at_zytor.com)

Date: 10/05/03

To: linux-kernel@vger.kernel.org

Date: 4 Oct 2003 17:51:11 -0700

Hi all,

Here is a horribly overdue update to unicode.txt. It was circled as a draft back in 2000, and the change is mostly identical except for text changes (based on what the official Unicode changes were.)

The changes in summary:

- Straight-to-font is defined as F000–F7FF.
- F800–F804 are deprecated in favour of the new official Unicodes.
- Some additions for keyboards, now described in vendor-neutral language. Note that the looks of any partical glyphs is not specified.

Last update: 2003–10–04, version 1.2

This file is maintained by H. Peter Anvin <unicode@lanana.org> as part of the Linux Assigned Names And Numbers Authority (LANANA) project. The current version can be found at:

<http://www.lanana.org/docs/unicode/unicode.txt>

The Linux kernel code has been rewritten to use Unicode to map characters to fonts. By downloading a single Unicode-to-font table, both the eight-bit character sets and UTF-8 mode are changed to use the font as indicated.

This changes the semantics of the eight-bit character tables subtly. The four character tables are now:

Map symbol Map name Escape code (G0)

LAT1_MAP Latin-1 (ISO 8859-1) ESC (B

GRAF_MAP DEC VT100 pseudographics ESC (0

IBMP_C_MAP IBM code page 437 ESC (U

Linux–Kernel: Horribly overdue update to unicode.txt

USER_MAP User defined ESC (K

In particular, ESC (U is no longer "straight to font", since the font might be completely different than the IBM character set. This permits for example the use of block graphics even with a Latin–1 font loaded.

In accordance with the Unicode standard/ISO 10646 the range U+F000 to U+F8FF has been reserved for OS–wide allocation (the Unicode Standard refers to this as a "Corporate Zone", since this is inaccurate for Linux we call it the "Linux Zone"). U+F000 was picked as the starting point since it lets the direct–mapping area start on a large power of two (in case 1024– or 2048–character fonts ever become necessary). This leaves U+E000 to U+EFFF as End User Zone.

[v1.2]: The Unicodes range from U+F000 and up to U+F7FF have been hard–coded to map directly to the loaded font, bypassing the translation table. The user–defined map now defaults to U+F000 to U+FOF