NAME
ot2kpx − extract kerning information from OpenType fonts

SYNOPSIS
ot2kpx [−*afm* | −*kpx* | −*lua*] **<fontfile>**

DESCRIPTION
ot2kpx extract the kerning data from OpenType fonts and prints it to `stdout`, either in Adobe's KPX format (for adding to an *afm* file) or as a Luatex custom feature, for use with the `\directlua` command.

OPTIONS AND ARGUMENTS
−help
> Print a short help text and exit.

−version
> Print **ot2kpx**'s version and exit.

−afm, **−kpx**
> Output the kerning data in Adobe's KPX format, as used in *afm* files. This is the default output format.

−lua
> Output the kerning data as a Luatex custom font feature, to be included in a `\directlua` command.

<fontfile>
> The OpenType font (both *otf* and *ttf* format are supported).

RESTRICTIONS
− **ot2kpx** doesn't implement all of the OpenType specification. Things that are missing include: support for font files containing multiple fonts, LookupTables with LookupTypes other than 2, "kern" tables with format other than 0 and ValueRecords with other types of data than just XAdvance data.

Most of these limitations won't matter, since the missing features are rare (the only fonts I know of that use them are the non-western fonts that come with Adobe Reader). Furthermore, many of these features define (according to the OpenType specification) *"subtle, device-dependent adjustments at specific font sizes or device resolutions"*; since there's no way to express such adjustments in *afm* format, ignoring them seems to be the only option anyway.

− **ot2kpx** collects kerning data first from the "kern" table, then from all LookupTables associated with the "kern" feature; if a kerning pair occurs multiple times, the first value seen is chosen. There are (or may be) several issues with this approach:

− The OpenType specification says that fonts in *otf* format shouldn't use the "kern" table at all, just the lookups from the "GPOS" table. Many such fonts do, however, contain a "kern" table, but no "GPOS" table; so we use the "kern" table anyway.

− Instead of reading all LookupTables, it might be better to let the user specify a script and language and process only the LookupTables for those values. However, at least in the fonts I checked, all script/language combinations eventually point to the *same* "kern" LookupTables, so this approach wouldn't make any difference (apart from further complicating the code).

AUTHOR
Marc Penninga <marcpenninga@gmail.com>

COPYRIGHT
Copyright (C) 2005−2026 Marc Penninga.

LICENSE
This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 2 of the License, or (at your option) any later version. A copy of the GNU General Public License is included with **ot2kpx**; see the file *GPLv2.txt*.

DISCLAIMER

   This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

VERSION

   This document describes **ot2kpx** version 20260210.

RECENT CHANGES

   (See the source code for the rest of the story.)

   2019−05−20 Added the −*version* option.

   2019−04−15 Added the −*lua* command-line option to get output in Luatex's custom feature format.