

The `xepersian-hm` package

Fixing kashida in xepersian

Hossein Movahhedian*

Released 2020-04-20 v0.5a

Kashida feature in `xepersian` has problems with some fonts such as **HM Series** fonts available at <https://dma8hm1334.bitbucket.io> and **X Series 2** fonts available at http://wiki.irmug.com/index.php/X_Series_2. The `xepersian-hm` package fixes these problems.

The file `texmf-dist/doc/xelatex/xepersian-hm/kashida-example.tex` can be used as a simple example of the usage of the package.

Please use the Bitbucket issue tracker: <https://bitbucket.org/dma8hm1334/xepersian-hm/issues> to report a bug, request a feature or if you have a comment.

I will do my best to fix all the bugs you report, but, unfortunately, time is a big hurdle to overcome; so, my apologies in advance for those which I cannot make time to fix.

1 Package loading and options

Please note that `xepersian-hm` loads `xepersian` automatically, so you may only pass options to the package using the command `\PassOptionsToPackage` before `\documentclass`. For example:

```
\PassOptionsToPackage{Kashida=off,RTLdocument=on}{xepersian}  
\documentclass{report}  
\usepackage{xcolor}  
\usepackage[Kashida,kashidastretch=0.14 em plus 0.5 em]{xepersian-hm}
```

The options available in `xepersian-hm` are:

- **Kashida**: which implements kashida feature with two possible values:
 - **glyph**: which uses the **Kashida** character to stretch the text. Please note that this feature is still experimental and is not regarded as stable. If you are going to use this option be ready for unpredictable results.
 - **hrule**: which uses a horizontal rule (`\hrule`) to stretch the text.

For example `Kashida=glyph` inserts a stretched **Kashida** glyph where ever it is needed.

*E-mail: dma8hm1334@gmail.com

- **linebreakpenalty**: which specifies the amount of penalty for preventing bad line-breaking. You may strictly specify the value of this option. For example 'linebreakpenalty=8' or you may use the default values. The available default values of penalties which are put into the horizontal list output are the negative of the followings:

min:	0
low:	8
medium:	15
high:	25
max:	10000

- **kashidastretch**: which specifies the amount of extra stretching for some combinations of characters. You may strictly specify the value of this option. For example 'kashidastretch=0.14 em plus 0.5 em' or you may use the default values. The available default values are:

kayhan	0.14 em plus 0.5 em
khorrāmshahr	0.131 em plus 0.5 em
kayhannavaar	0.129 em plus 0.5 em
kayhanpook	0.133 em plus 0.5 em
kayhansayeh	0.135 em plus 0.5 em
khoramshahr	0.128 em plus 0.5 em
khorrāmshahr	0.13 em plus 0.5 em
niloofar	0.132 em plus 0.5 em
paatch	0.127 em plus 0.5 em
riyaz	0.125 em plus 0.5 em
roya	0.142 em plus 0.5 em
shafigh	0.143 em plus 0.5 em
shafighKurd	0.126 em plus 0.5 em
shafighUzbek	0.123 em plus 0.5 em
shiraz	0.122 em plus 0.5 em
sols	0.124 em plus 0.5 em
tabriz	0.119 em plus 0.5 em
titr	0.12 em plus 0.5 em
titre	0.121 em plus 0.5 em
traffic	0.124 em plus 0.5 em
vahid	0.134 em plus 0.5 em
vosta	0.136 em plus 0.5 em
yaghut	0.138 em plus 0.5 em
yagut	0.137 em plus 0.5 em
yas	0.126 em plus 0.5 em
yekan	0.141 em plus 0.5 em
yermook	0.139 em plus 0.5 em
zar	0.116 em plus 0.5 em
ziba	0.119 em plus 0.5 em
default	0.14 em plus 0.5 em
noskip	0 em plus 0.5 em

For example: 'kashidastretch=titr' is equivalent to 'kashidastretch=0.12 em plus 0.5 em'

2 Commands

`\discouragebadlinebreaks{text}` *text* may not contain paragraph breaks. This command puts a penalty after each Persian word. The value of the penalty is set by the option `linebreakpenalty`. The default value is 8.

3 xepersian-hm Implementation

3.1 File: xepersian-hm.sty

```

1 (*package)
2 \RequirePackage{l3keys2e}
3 \RequirePackage{graphicx}[2019-11-30]
4 \RequirePackage{zref-savepos}[2020-03-03]
5 \RequirePackage{xepersian}
6 \ExplSyntaxOn
7 \ProvidesExplPackage {xepersian-hm} {2020-04-20} {0.5a} { Fixes~issues~in~xepersian~package }
8
9 \bool_set_false:N \l_xephm_kashida_hm_fix_bool
10
11 \tl_new:N \l_xephm_gap_filler_tl
12
13 \bool_set_false:N \l_xephm_linebreakpenalty_bool
14
15 \int_new:N \l_xephm_line_break_penalty_int
16 \int_new:N \l_xephm_min_penalty_int
17 \int_new:N \l_xephm_low_penalty_int
18 \int_new:N \l_xephm_med_penalty_int
19 \int_new:N \l_xephm_high_penalty_int
20 \int_new:N \l_xephm_max_penalty_int
21
22 \int_set:Nn \l_xephm_min_penalty_int { 0 }
23 \int_set:Nn \l_xephm_low_penalty_int { 8 }
24 \int_set:Nn \l_xephm_med_penalty_int { 15 }
25 \int_set:Nn \l_xephm_high_penalty_int { 25 }
26 \int_set:Nn \l_xephm_max_penalty_int { 10000 }
27
28 \tl_set:Nn \l_xephm_stretch_glyph_tl { glyph }
29 \tl_set:Nn \l_xephm_stretch_hrulerule_tl { hrulerule }
30
31 \tl_set:Nn \l_xephm_kayhan_tl { kayhan }
32 \tl_set:Nn \l_xephm_khorramshahr_tl { khorramshahr }
33 \tl_set:Nn \l_xephm_kayhannavaar_tl { kayhannavaar }
34 \tl_set:Nn \l_xephm_kayhanpook_tl { kayhanpook }
35 \tl_set:Nn \l_xephm_kayhansayeh_tl { kayhansayeh }
36 \tl_set:Nn \l_xephm_khoramshahr_tl { khoramshahr }
37 \tl_set:Nn \l_xephm_khorramshahr_tl { khorramshahr }
38 \tl_set:Nn \l_xephm_niloofar_tl { niloofar }
39 \tl_set:Nn \l_xephm_paatch_tl { paatch }

```

```

40 \tl_set:Nn \l_xephm_riyaz_tl { riyaz }
41 \tl_set:Nn \l_xephm_roya_tl { roya }
42 \tl_set:Nn \l_xephm_shafigh_tl { shafigh }
43 \tl_set:Nn \l_xephm_shafighKurd_tl { shafighKurd }
44 \tl_set:Nn \l_xephm_shafighUzbek_tl { shafighUzbek }
45 \tl_set:Nn \l_xephm_shiraz_tl { shiraz }
46 \tl_set:Nn \l_xephm_sols_tl { sols }
47 \tl_set:Nn \l_xephm_tabriz_tl { tabriz }
48 \tl_set:Nn \l_xephm_titr_tl { titr }
49 \tl_set:Nn \l_xephm_titre_tl { titre }
50 \tl_set:Nn \l_xephm_traffic_tl { traffic }
51 \tl_set:Nn \l_xephm_vahid_tl { vahid }
52 \tl_set:Nn \l_xephm_vosta_tl { vosta }
53 \tl_set:Nn \l_xephm_yaghut_tl { yaghut }
54 \tl_set:Nn \l_xephm_yagut_tl { yagut }
55 \tl_set:Nn \l_xephm_yas_tl { yas }
56 \tl_set:Nn \l_xephm_yekan_tl { yekan }
57 \tl_set:Nn \l_xephm_yermook_tl { yermook }
58 \tl_set:Nn \l_xephm_zar_tl { zar }
59 \tl_set:Nn \l_xephm_ziba_tl { ziba }
60 \tl_set:Nn \l_xephm_default_tl { default }
61 \tl_set:Nn \l_xephm_noskip_tl { noskip }
62
63 \keys_define:nn { xepersian-hm }
64 {
65   Kashida .code:n =
66   {
67     \tl_set:Nn \l_tmpa_tl { #1 }
68     \tl_case:NnTF \l_tmpa_tl
69     {
70       \l_xephm_stretch_glyph_tl { \tl_set:Nn \l_xephm_gap_filler_tl { \l_xephm_stretch_glyph_
71       \l_xephm_stretch_hruler_tl { \tl_set:Nn \l_xephm_gap_filler_tl { \l_xephm_stretch_hruler_
72       } } { \tl_set:Nn \l_xephm_filler_default_tl { #1 } }
73       \bool_set_true:N \l_xephm_kashida_hm_fix_bool
74       \tex_input:D { kashida-xepersian-hm.def }
75     } ,
76
77   linebreakpenalty .code:n =
78   {
79     \int_set:Nn \l_tmpa_int { #1 }
80     \int_case:nnTF \l_tmpa_int
81     {
82       \l_xephm_min_penalty_int { \int_set:Nn \l_xephm_line_break_penalty_int { \l_xephm_min_p
83       \l_xephm_low_penalty_int { \int_set:Nn \l_xephm_line_break_penalty_int { \l_xephm_low_p
84       \l_xephm_med_penalty_int { \int_set:Nn \l_xephm_line_break_penalty_int { \l_xephm_med_p
85       \l_xephm_high_penalty_int { \int_set:Nn \l_xephm_line_break_penalty_int { \l_xephm_high
86       \l_xephm_max_penalty_int { \int_set:Nn \l_xephm_line_break_penalty_int { \l_xephm_max_p
87     } } { \int_set:Nn \l_xephm_line_break_penalty_int { #1 } }
88     \bool_set_true:N \l_xephm_linebreakpenalty_bool
89   } ,
90
91   kashidastretch .code:n =
92   {
93     \tl_set:Nn \l_tmpb_tl { #1 }

```

```

94 \tl_case:NnTF \l_tmpb_tl
95 {
96   \l_xephm_kayhan_tl      { \tl_set:Nn \l_xephm_hskip_default_tl { 0.14 em plus 0.5 em }
97   \l_xephm_khorramshahr_tl { \tl_set:Nn \l_xephm_hskip_default_tl { 0.131 em plus 0.5 em }
98   \l_xephm_kayhannavaar_tl { \tl_set:Nn \l_xephm_hskip_default_tl { 0.129 em plus 0.5 em }
99   \l_xephm_kayhanpook_tl   { \tl_set:Nn \l_xephm_hskip_default_tl { 0.133 em plus 0.5 em }
100  \l_xephm_kayhansayeh_tl   { \tl_set:Nn \l_xephm_hskip_default_tl { 0.135 em plus 0.5 em }
101  \l_xephm_khoramshahr_tl   { \tl_set:Nn \l_xephm_hskip_default_tl { 0.128 em plus 0.5 em }
102  \l_xephm_khorramshahr_tl { \tl_set:Nn \l_xephm_hskip_default_tl { 0.13 em plus 0.5 em }
103  \l_xephm_niloofar_tl      { \tl_set:Nn \l_xephm_hskip_default_tl { 0.132 em plus 0.5 em }
104  \l_xephm_paatch_tl        { \tl_set:Nn \l_xephm_hskip_default_tl { 0.127 em plus 0.5 em }
105  \l_xephm_riyaz_tl         { \tl_set:Nn \l_xephm_hskip_default_tl { 0.125 em plus 0.5 em }
106  \l_xephm_roya_tl          { \tl_set:Nn \l_xephm_hskip_default_tl { 0.142 em plus 0.5 em }
107  \l_xephm_shafigh_tl       { \tl_set:Nn \l_xephm_hskip_default_tl { 0.143 em plus 0.5 em }
108  \l_xephm_shafighKurd_tl   { \tl_set:Nn \l_xephm_hskip_default_tl { 0.126 em plus 0.5 em }
109  \l_xephm_shafighUzbek_tl  { \tl_set:Nn \l_xephm_hskip_default_tl { 0.123 em plus 0.5 em }
110  \l_xephm_shiraz_tl        { \tl_set:Nn \l_xephm_hskip_default_tl { 0.122 em plus 0.5 em }
111  \l_xephm_sols_tl          { \tl_set:Nn \l_xephm_hskip_default_tl { 0.124 em plus 0.5 em }
112  \l_xephm_tabriz_tl        { \tl_set:Nn \l_xephm_hskip_default_tl { 0.119 em plus 0.5 em }
113  \l_xephm_titr_tl          { \tl_set:Nn \l_xephm_hskip_default_tl { 0.12 em plus 0.5 em }
114  \l_xephm_titre_tl         { \tl_set:Nn \l_xephm_hskip_default_tl { 0.121 em plus 0.5 em }
115  \l_xephm_traffic_tl       { \tl_set:Nn \l_xephm_hskip_default_tl { 0.124 em plus 0.5 em }
116  \l_xephm_vahid_tl         { \tl_set:Nn \l_xephm_hskip_default_tl { 0.134 em plus 0.5 em }
117  \l_xephm_vosta_tl         { \tl_set:Nn \l_xephm_hskip_default_tl { 0.136 em plus 0.5 em }
118  \l_xephm_yaghut_tl        { \tl_set:Nn \l_xephm_hskip_default_tl { 0.138 em plus 0.5 em }
119  \l_xephm_yagut_tl         { \tl_set:Nn \l_xephm_hskip_default_tl { 0.137 em plus 0.5 em }
120  \l_xephm_yas_tl           { \tl_set:Nn \l_xephm_hskip_default_tl { 0.126 em plus 0.5 em }
121  \l_xephm_yekan_tl         { \tl_set:Nn \l_xephm_hskip_default_tl { 0.141 em plus 0.5 em }
122  \l_xephm_ymook_tl         { \tl_set:Nn \l_xephm_hskip_default_tl { 0.139 em plus 0.5 em }
123  \l_xephm_zar_tl           { \tl_set:Nn \l_xephm_hskip_default_tl { 0.116 em plus 0.5 em }
124  \l_xephm_ziba_tl          { \tl_set:Nn \l_xephm_hskip_default_tl { 0.119 em plus 0.5 em }
125  \l_xephm_default_tl       { \tl_set:Nn \l_xephm_hskip_default_tl { 0.14 em plus 0.5 em }
126  \l_xephm_noskip_tl        { \tl_set:Nn \l_xephm_hskip_default_tl { 0 em plus 0.5 em }
127  } { } { \tl_set:Nn \l_xephm_hskip_default_tl { #1 } }
128 } ,
129 kashidastretch .default:n = \tl_set:Nn \l_xephm_hskip_default_tl { 0 em plus 0.5 em } ,
130 }
131
132 \ProcessKeysOptions { xepersian-hm }
133
134 \ExplSyntaxOff
135 \endinput
136 \</package>

```

3.2 File: kashida-xepersian-hm.def

```

137 < *kashida-xepersian-hm>
138 \ExplSyntaxOn
139 \ProvidesExplFile {kashida-xepersian-hm.def} {2020-04-20} {0.5a} { Fixes~implementation-of~Kashida-
140
141 \int_const:Nn \c_xephm_zwj_int {"200D} % zero-width joiner
142 \int_const:Nn \c_xephm_lrm_int {"200E} % left-right-mark
143 \int_const:Nn \c_xephm_two_int {2} % 2
144 \int_const:Nn \c_xephm_four_int {4} % 4
145 \int_const:Nn \c_xephm_ksh_int {"0640} % kashida

```

```

146
147 \int_new:N \l_xephm_counter_int
148
149 \newXeTeXintercharclass \c_xephm_d_charclass % dual-joiner class
150 \newXeTeXintercharclass \c_xephm_l_charclass % lam
151 \newXeTeXintercharclass \c_xephm_r_charclass % right-joiner
152 \newXeTeXintercharclass \c_xephm_a_charclass % alef
153 \newXeTeXintercharclass \c_xephm_y_charclass % yeh
154
155 \tl_new:N \l_xephm_hskip_zero_tl
156 \tl_new:N \l_xephm_hskip_default_tl
157
158 \dim_new:N \l_xephm_diff_pos_dim
159
160 %% Partly adapted from LaTeX2e source
161 \cs_new:Nn \xephm_line_break: {
162   \if_mode_vertical:
163     \GenericError{
164       \space\space\space\space\space\space\space\space\space\space\space\space\space\space\space\space
165     }{
166       LaTeX Error: Theres no line here to end
167     }{
168       See the LaTeX manual or LaTeX Companion for explanation.
169     }{
170       Your command was ignored.\MessageBreak
171       Type \space I <command> <return> \space to replace it~
172       with another command,\MessageBreak
173       or \space <return> \space to continue without it.}
174   \else:
175     \l_tmpa_skip \tex_lastskip:D
176     \tex_unskip:D
177     \tex_penalty:D -\l_xephm_line_break_penalty_int
178     \dim_compare:nT { \l_tmpa_skip > \c_zero_skip }
179     { \skip_horizontal:N \l_tmpa_skip \tex_ignorespaces:D }
180   \fi:
181 }
182
183 \NewDocumentCommand { \discouragebadlinebreaks } { m }
184 {
185   \xephm_put_line_breaks:n { #1 }
186 }
187
188 \tl_new:N \l_xephm_line_break_tl
189
190 \cs_new_protected:Nn \xephm_put_line_breaks:n
191 {
192   \tl_set:Nn \l_xephm_line_break_tl { #1 }
193   \regex_replace_all:nnN { ([\])+ } { \ \0\ \c{xephm_line_break:}\ } \l_xephm_line_break_tl
194   \tl_use:N \l_xephm_line_break_tl
195 }
196
197 %% Partly adapted from the code provided by David Carlisle in:
198 %% https://tex.stackexchange.com/questions/356709/how-to-know-the-width-and-fill-the-glue-space-between
199 \cs_new:Npn \xephm_kashida_glyph #1

```

```

200 {
201   \bool_if:NT \l_xephm_kashida_hm_fix_bool
202   {
203     \c_xephm_lrm_int\tex_penalty:D 10000
204     \mode_leave_vertical:
205     \tex_global:D \tex_advance:D \l_xephm_counter_int \c_one_int
206
207     \tl_set:Nx \l_xephm_pos_tl { pos_\tex_romannumeral:D \l_xephm_counter_int }
208     \tl_set:Nx \l_xephm_zref_tl { zref_\tex_romannumeral:D \l_xephm_counter_int }
209
210     \zsavexposx{x_i\l_xephm_zref_tl}
211     \tl_set:Nx \l_tmpa_tl
212     {
213       \iow_now:cx { @auxout }
214       {
215         \ExplSyntaxOn
216         \tl_gset:Nn \exp_after:wN \token_to_str:N \cs:w get_x_i\l_xephm_pos_tl \cs_end: { \zposx{
217         \ExplSyntaxOff
218       }
219     }
220     \l_tmpa_tl
221     \skip_horizontal:n { #1 }
222     \zsavexposx{x_f\l_xephm_zref_tl}
223     \tl_set:Nx \l_tmpa_tl
224     {
225       \iow_now:cx { @auxout }
226       {
227         \ExplSyntaxOn
228         \tl_gset:Nn \exp_after:wN \token_to_str:N \cs:w get_x_f\l_xephm_pos_tl \cs_end: { \zposx{
229         \ExplSyntaxOff
230       }
231     }
232     \l_tmpa_tl
233     \exp_after:wN
234     \if_meaning:w
235     \cs:w get_x_i\l_xephm_pos_tl \cs_end: \tex_relax:D
236     \else:
237     \dim_set:Nn \l_xephm_diff_pos_dim
238     {
239       \dim_eval:n { \cs:w get_x_i\l_xephm_pos_tl \cs_end: sp - \cs:w get_x_f\l_xephm_pos_tl \cs
240     }
241     \dim_compare:nTF { \l_xephm_diff_pos_dim == 0sp }
242     { }
243     { \llap { \resizebox { \l_xephm_diff_pos_dim \tex_relax:D } { \height } { \c_xephm_ksh_int
244     \fi:
245   }
246 }
247
248 %% Partly adapted from the code provided by Jonathan Kew in:
249 %% https://tug.org/pipermail/xetex/2009-February/012307.html.
250 %% Somebody notified me that the code in 'kashida-xepersian.def' from xepersian
251 %% package is an exact copy of Jonathan Kew's code. Being unaware of this, in
252 %% the earlier versions of this package I made a mistake and acknowledged
253 %% Vafa Khalighi instead of Jonathan Kew. A sincere thank you to Jonathan Kew

```

```

254 %% for his excellent code.
255 \cs_new:Npn \xephm_kashida_hrulerule #1
256 {
257   \bool_if:NT \l_xephm_kashida_hm_fix_bool
258   {
259     \c_xephm_lrm_int\c_xephm_zwj_int\tex_penalty:D 10000
260     \tex_leaders:D \tex_hrulerule:D height \XeTeXglyphbounds \c_xephm_two_int
261     \int_use:N \XeTeXcharglyph \c_xephm_ksh_int depth \XeTeXglyphbounds \c_xephm_four_int
262     \int_use:N \XeTeXcharglyph \c_xephm_ksh_int \skip_horizontal:n { #1 }
263     \c_xephm_zwj_int
264   }
265 }
266
267 \XeTeXinterchartokenstate = 1
268
269 \clist_set:Nn \l_xephm_a_clist { 0622,0623,0625,0627 } %
270 \clist_map_inline:Nn \l_xephm_a_clist
271 {
272   \XeTeXcharclass "#1 \c_xephm_a_charclass
273 }
274
275 \clist_set:Nn \l_xephm_d_clist { 0626,0628,062A,062B,062C,062D,062E,0633,0634,0635,0636,0637,0638,0639 } %
276 \clist_map_inline:Nn \l_xephm_d_clist
277 {
278   \XeTeXcharclass "#1 \c_xephm_d_charclass
279 }
280
281 \clist_set:Nn \l_xephm_l_clist { 0644 } %
282 \clist_map_inline:Nn \l_xephm_l_clist
283 {
284   \XeTeXcharclass "#1 \c_xephm_l_charclass
285 }
286
287 \clist_set:Nn \l_xephm_r_clist { 0624,0629,062F,0630,0631,0632,0648,0698 } % , , , , , ,
288 \clist_map_inline:Nn \l_xephm_r_clist
289 {
290   \XeTeXcharclass "#1 \c_xephm_r_charclass
291 }
292
293 \clist_set:Nn \l_xephm_y_clist { 0649,064A,06CC } % , ,
294 \clist_map_inline:Nn \l_xephm_y_clist
295 {
296   \XeTeXcharclass "#1 \c_xephm_y_charclass
297 }
298
299 \msg_new:nnx { xepersian-hm } { value-not-available-for-kashida-option }
300 {
301   Sorry,~ value~ '\l_xephm_gap_filler_tl'~ is~ not~ available~ for~ kashida~ option~ yet!
302 }
303
304 \str_case:VnTF \l_xephm_gap_filler_tl
305 {
306   \l_xephm_stretch_glyph_tl
307   {

```



```

308 \XeTeXinterchartoks \c_xephm_y_charclass \c_xephm_y_charclass =
309 {
310     \bool_if:NTF \l_xephm_kashida_hm_fix_bool
311     { \c_xephm_zwj_int \xephm_kashida_glyph \l_xephm_hskip_default_tl \c_xephm_zwj_int }
312     { \c_xephm_zwj_int \xephm_kashida_glyph \l_xephm_hskip_zero_tl \c_xephm_zwj_int }
313 }
314 \XeTeXinterchartoks \c_xephm_d_charclass \c_xephm_y_charclass =
315 {
316     \bool_if:NTF \l_xephm_kashida_hm_fix_bool
317     { \c_xephm_zwj_int \xephm_kashida_glyph \l_xephm_hskip_default_tl \c_xephm_zwj_int }
318     { \c_xephm_zwj_int \xephm_kashida_glyph \l_xephm_hskip_zero_tl \c_xephm_zwj_int }
319 }
320 \XeTeXinterchartoks \c_xephm_y_charclass \c_xephm_d_charclass =
321 { \c_xephm_zwj_int \xephm_kashida_glyph \l_xephm_hskip_zero_tl \c_xephm_zwj_int }
322 \XeTeXinterchartoks \c_xephm_d_charclass \c_xephm_d_charclass =
323 { \c_xephm_zwj_int \xephm_kashida_glyph \l_xephm_hskip_zero_tl \c_xephm_zwj_int }
324 \XeTeXinterchartoks \c_xephm_l_charclass \c_xephm_d_charclass =
325 { \c_xephm_zwj_int \xephm_kashida_glyph \l_xephm_hskip_zero_tl \c_xephm_zwj_int }
326 \XeTeXinterchartoks \c_xephm_d_charclass \c_xephm_l_charclass =
327 { \c_xephm_zwj_int \xephm_kashida_glyph \l_xephm_hskip_zero_tl \c_xephm_zwj_int }
328 \XeTeXinterchartoks \c_xephm_l_charclass \c_xephm_l_charclass =
329 { \c_xephm_zwj_int \xephm_kashida_glyph \l_xephm_hskip_zero_tl \c_xephm_zwj_int }
330 \XeTeXinterchartoks \c_xephm_d_charclass \c_xephm_r_charclass =
331 { \c_xephm_zwj_int \xephm_kashida_glyph \l_xephm_hskip_zero_tl \c_xephm_zwj_int }
332 \XeTeXinterchartoks \c_xephm_d_charclass \c_xephm_a_charclass =
333 { \c_xephm_zwj_int \xephm_kashida_glyph \l_xephm_hskip_zero_tl \c_xephm_zwj_int }
334 \XeTeXinterchartoks \c_xephm_l_charclass \c_xephm_r_charclass =
335 { \c_xephm_zwj_int \xephm_kashida_glyph \l_xephm_hskip_zero_tl \c_xephm_zwj_int }
336 \XeTeXinterchartoks \c_xephm_l_charclass \c_xephm_a_charclass =
337 { }
338 }
339 \l_xephm_stretch_hruler_tl
340 {
341     \XeTeXinterchartoks \c_xephm_y_charclass \c_xephm_y_charclass =
342     {
343         \bool_if:NTF \l_xephm_kashida_hm_fix_bool
344         { \xephm_kashida_hruler \l_xephm_hskip_default_tl }
345         { \xephm_kashida_hruler \l_xephm_hskip_zero_tl }
346     }
347     \XeTeXinterchartoks \c_xephm_d_charclass \c_xephm_y_charclass =
348     {
349         \bool_if:NTF \l_xephm_kashida_hm_fix_bool
350         { \xephm_kashida_hruler \l_xephm_hskip_default_tl }
351         { \xephm_kashida_hruler \l_xephm_hskip_zero_tl }
352     }
353     \XeTeXinterchartoks \c_xephm_y_charclass \c_xephm_d_charclass =
354     { \xephm_kashida_hruler \l_xephm_hskip_zero_tl }
355     \XeTeXinterchartoks \c_xephm_d_charclass \c_xephm_d_charclass =
356     { \xephm_kashida_hruler \l_xephm_hskip_zero_tl }
357     \XeTeXinterchartoks \c_xephm_l_charclass \c_xephm_d_charclass =
358     { \xephm_kashida_hruler \l_xephm_hskip_zero_tl }
359     \XeTeXinterchartoks \c_xephm_d_charclass \c_xephm_l_charclass =
360     { \xephm_kashida_hruler \l_xephm_hskip_zero_tl }
361     \XeTeXinterchartoks \c_xephm_l_charclass \c_xephm_l_charclass =

```

```

362         { \xephm_kashida_hrulerule \l_xephm_hskip_zero_tl }
363 \XeTeXinterchartoks \c_xephm_d_charclass \c_xephm_r_charclass =
364 { \xephm_kashida_hrulerule \l_xephm_hskip_zero_tl }
365 \XeTeXinterchartoks \c_xephm_d_charclass \c_xephm_a_charclass =
366 { \xephm_kashida_hrulerule \l_xephm_hskip_zero_tl }
367 \XeTeXinterchartoks \c_xephm_l_charclass \c_xephm_r_charclass =
368 { \xephm_kashida_hrulerule \l_xephm_hskip_zero_tl }
369 \XeTeXinterchartoks \c_xephm_l_charclass \c_xephm_a_charclass =
370 { }
371 }
372 } { } { \msg_error:nn { xepersian-hm } { value-not-available-for-kashida-option } }
373
374 \NewDocumentCommand \KashidaHMFixOn {} { \bool_set_true:N \l_xephm_kashida_hm_fix_bool }
375 \NewDocumentCommand \KashidaHMFixOff {} { \bool_set_false:N \l_xephm_kashida_hm_fix_bool }
376
377 \tex_let:D \KashidaOn \KashidaHMFixOn
378 \tex_let:D \KashidaOff \KashidaHMFixOff
379
380 \tl_set:Nn \l_xephm_hskip_zero_tl { 0 em plus 0.5 em }
381
382 \bool_if:NTF \l_xephm_kashida_hm_fix_bool
383 {
384   \tl_if_empty:NT \l_xephm_hskip_default_tl { \tl_set:Nn \l_xephm_hskip_default_tl { 0.14 em plus 0.5 em } }
385 }
386 {
387   \tl_set:NV \l_xephm_hskip_default_tl \l_xephm_hskip_zero_tl
388 }
389
390 \ExplSyntaxOff
391
392 \makeatletter
393 \newif\if@Kashida@on
394 \newif\if@Kashida@XB@fix
395 \makeatother
396
397 \endinput
398 </kashida-xepersian-hm>

```