

The **xepersian-hm** package
Fixing kashida in xepersian
Source code documentation

Hossein Movahhedian*

Released 2020-09-04 v1.1a

HM: *HM is an initialism for “Hossein Movahhedian” – the author’s name and family.*

*E-mail: dma8hm1334@gmail.com

Contents

1	XePersianHM Implementation	3
1.1	File: xepersian-hm.sty	3
1.2	File: xepersian-hm-kashida.tex	9
1.3	File: xepersian-hm-common-kashida.tex	13
2	Acknowledgments	14
3	Change History	14
	[2020-03-17 v0.1]	14
	[2020-03-19 v0.2]	14
	[2020-03-23 v0.3]	14
	[2020-03-26 v0.4]	15
	[2020-04-20 v0.5a]	15
	[2020-04-26 v0.5b]	15
	[2020-05-01 v0.5c]	15
	[2020-05-07 v0.5d]	15
	[2020-05-12 v0.5e]	15
	[2020-05-20 v0.5f]	15
	[2020-05-31 v1.0a]	15
	[2020-06-04 v1.0b]	15
	[2020-07-20 v1.0c]	16
	[2020-08-05 v1.0d]	16
	[2020-09-04 v1.1a]	16
	Index	17

1 XePersianHM Implementation

1.1 File: xepersian-hm.sty

```
1 <*xepersian-hm-sty>
2 \NeedsTeXFormat{LaTeX2e}
3 \RequirePackage{expl3}[2020-03-06]
4 \@ifpackagelater{expl3}{2020-03-06}
5 {}
6 {%
7   \PackageError{xepersian-hm}{Support package l3kernel too old}
8   {%
9     Please install an up to date version of l3kernel\MessageBreak
10    using your TeX package manager or from CTAN.\MessageBreak
11    \MessageBreak
12    Loading xtemplate will abort!%
13    }%
14  \endinput
15 }
16 \RequirePackage{l3keys2e}
17 \RequirePackage{graphicx}[2019-11-30]
18 \RequirePackage{zref-savepos}[2020-03-03]
19 \RequirePackage{xcolor}[2016/05/11]
20 \RequirePackage{xepersian}
21 \ProvidesExplPackage {xepersian-hm} {2020-09-04} {1.1a} { Fixes-issues-in-xepersian-package }
22
23 \def\XePersianHM{XePersianHM}
24
25 \box_new:N \l_xephm_ksh_box
26
27 \int_const:Nn \c_xephm_ksh_int {"0640} % kashida
28 \int_const:Nn \c_xephm_lrm_int {"200E} % left-right-mark
29 \int_const:Nn \c_xephm_zwj_int {"200D} % zero-width joiner
30
31 \int_const:Nn \c_xephm_two_int {2}
32 \int_const:Nn \c_xephm_four_int {4}
33
34 \tl_const:Nn \c_xephm_skip_a_tl { 0 em plus 0.5 em }
35 \tl_const:Nn \c_xephm_skip_b_tl { 0.14 em plus 5.5 em }
36
37 \int_new:N \l_xephm_counter_int
38
39 \int_new:N \l_xephm_kashida_slot_int
40
41 \int_new:N \l_xephm_line_break_penalty_int
42
43 \int_new:N \l_xephm_min_penalty_int
44 \int_new:N \l_xephm_low_penalty_int
45 \int_new:N \l_xephm_med_penalty_int
46 \int_new:N \l_xephm_high_penalty_int
47 \int_new:N \l_xephm_max_penalty_int
48
49 \tl_new:N \l_xephm_line_break_tl
50
```

```

51 \tl_new:N \l_xephm_main_font_full_tl
52 \tl_new:N \l_xephm_main_font_name_tl
53
54 \tl_new:N \l_xephm_font_full_tl
55 \tl_new:N \l_xephm_font_name_tl
56
57 \tl_new:N \l_xephm_skip_default_tl
58
59 \tl_new:N \l_xephm_active_ligs_tl
60
61 \tl_new:N \l_xephm_gap_filler_tl
62
63 \tl_new:N \l_xephm_use_color_tl
64 \tl_new:N \l_xephm_color_tl
65 \tl_new:N \l_xephm_color_rgb_tl
66
67 \dim_new:N \l_xephm_diff_pos_dim
68
69 \bool_set_false:N \l_xephm_kashida_fix_bool
70
71 \bool_set_false:N \l_xephm_ligature_bool
72 \bool_set_false:N \l_xephm_linebreakpenalty_bool
73 \bool_set_false:N \l_xephm_color_bool
74
75 \int_set:Nn \l_xephm_min_penalty_int { 0 }
76 \int_set:Nn \l_xephm_low_penalty_int { 8 }
77 \int_set:Nn \l_xephm_med_penalty_int { 15 }
78 \int_set:Nn \l_xephm_high_penalty_int { 25 }
79 \int_set:Nn \l_xephm_max_penalty_int { 10000 }
80
81 \tl_set:Nn \l_xephm_stretch_glyph_tl { glyph }
82 \tl_set:Nn \l_xephm_stretch_leaders_glyph_tl { leaders+glyph }
83 \tl_set:Nn \l_xephm_stretch_leaders_hrulerule_tl { leaders+hrule }
84 \tl_set:Nn \l_xephm_stretch_off_tl { Off }
85 \tl_set:Nn \l_xephm_stretch_on_tl { On }
86
87 \tl_set:Nn \l_xephm_fnt_kayhan_tl { kayhan }
88 \tl_set:Nn \l_xephm_fnt_kayhannavaar_tl { kayhannavaar }
89 \tl_set:Nn \l_xephm_fnt_kayhanpook_tl { kayhanpook }
90 \tl_set:Nn \l_xephm_fnt_kayhansayeh_tl { kayhansayeh }
91 \tl_set:Nn \l_xephm_fnt_khoramshahr_tl { khoramshahr }
92 \tl_set:Nn \l_xephm_fnt_khorramshahr_tl { khorrasmshahr }
93 \tl_set:Nn \l_xephm_fnt_niloofar_tl { niloofar }
94 \tl_set:Nn \l_xephm_fnt_paatch_tl { paatch }
95 \tl_set:Nn \l_xephm_fnt_riyaz_tl { riyaz }
96 \tl_set:Nn \l_xephm_fnt_roya_tl { roya }
97 \tl_set:Nn \l_xephm_fnt_shafigh_tl { shafigh }
98 \tl_set:Nn \l_xephm_fnt_shafighKurd_tl { shafighKurd }
99 \tl_set:Nn \l_xephm_fnt_shafighUzbek_tl { shafighUzbek }
100 \tl_set:Nn \l_xephm_fnt_shiraz_tl { shiraz }
101 \tl_set:Nn \l_xephm_fnt_sols_tl { sols }
102 \tl_set:Nn \l_xephm_fnt_tabriz_tl { tabriz }
103 \tl_set:Nn \l_xephm_fnt_titr_tl { titr }
104 \tl_set:Nn \l_xephm_fnt_titre_tl { titre }

```

```

105 \tl_set:Nn \l_xephm_fnt_traffic_tl      { traffic }
106 \tl_set:Nn \l_xephm_fnt_vahid_tl      { vahid }
107 \tl_set:Nn \l_xephm_fnt_vosta_tl      { vosta }
108 \tl_set:Nn \l_xephm_fnt_yaghut_tl     { yaghut }
109 \tl_set:Nn \l_xephm_fnt_yagut_tl      { yagut }
110 \tl_set:Nn \l_xephm_fnt_yas_tl        { yas }
111 \tl_set:Nn \l_xephm_fnt_yekan_tl      { yekan }
112 \tl_set:Nn \l_xephm_fnt_yermook_tl    { yermook }
113 \tl_set:Nn \l_xephm_fnt_zar_tl        { zar }
114 \tl_set:Nn \l_xephm_fnt_ziba_tl       { ziba }
115 \tl_set:Nn \l_xephm_fnt_default_tl    { default }
116 \tl_set:Nn \l_xephm_fnt_noskip_tl     { noskip }
117
118 \tl_set:Nn \l_xephm_lig_aalt_tl        { aalt } % Access All Alternatives
119 \tl_set:Nn \l_xephm_lig_ccmp_tl        { cmp } % Glyph Composition/Decomposition
120 \tl_set:Nn \l_xephm_lig_dlig_tl        { dlig } % Discretionary Ligatures
121 \tl_set:Nn \l_xephm_lig_fina_tl        { fina } % Final (Terminal) Forms
122 \tl_set:Nn \l_xephm_lig_init_tl        { init } % Initial Forms
123 \tl_set:Nn \l_xephm_lig_locl_tl        { locl } % Localized Forms
124 \tl_set:Nn \l_xephm_lig_medi_tl        { medi } % Medial Forms
125 \tl_set:Nn \l_xephm_lig_rlig_tl        { rlig } % Required Ligatures
126 \tl_set:Nn \l_xephm_lig_default_tl     { default }
127
128 \tl_set:Nn \l_xephm_col_default_tl     { magenta }
129
130 \clist_set:Nn \l_xephm_lig_aalt_clist   { } % Access All Alternatives
131 \clist_set:Nn \l_xephm_lig_ccmp_clist   { } % Glyph Composition/Decomposition
132 \clist_set:Nn \l_xephm_lig_dlig_clist   { FDF2 = , FDF3 = , FDFB = } % Discretionary Ligatures
133 \clist_set:Nn \l_xephm_lig_fina_clist   { } % Final (Terminal) Forms
134 \clist_set:Nn \l_xephm_lig_init_clist   { } % Initial Forms
135 \clist_set:Nn \l_xephm_lig_locl_clist   { } % Localized Forms
136 \clist_set:Nn \l_xephm_lig_medi_clist   { } % Medial Forms
137 \clist_set:Nn \l_xephm_lig_rlig_clist   { } % Required Ligatures
138 \clist_set:Nn \l_xephm_lig_default_clist { }
139
140 \clist_set:Nn \l_xephm_lig_names_clist
141 {
142   \l_xephm_lig_aalt_tl , { \l_xephm_lig_aalt_clist } ,
143   \l_xephm_lig_ccmp_tl , { \l_xephm_lig_ccmp_clist } ,
144   \l_xephm_lig_dlig_tl , { \l_xephm_lig_dlig_clist } ,
145   \l_xephm_lig_fina_tl , { \l_xephm_lig_fina_clist } ,
146   \l_xephm_lig_init_tl , { \l_xephm_lig_init_clist } ,
147   \l_xephm_lig_locl_tl , { \l_xephm_lig_locl_clist } ,
148   \l_xephm_lig_medi_tl , { \l_xephm_lig_medi_clist } ,
149   \l_xephm_lig_rlig_tl , { \l_xephm_lig_rlig_clist } ,
150 }
151
152 \msg_new:nnn { xepersian-hm } { error-kashida-character-is-not-available-in-the-
main-font }
153 {
154   Sorry,~ kashida~ character~ is~ not~ available~ in~ the~ main~ font~#!
155 }
156
157 \msg_new:nnn { xepersian-hm } { error-value-not-available-for-kashida-option }

```

```

158 {
159   Sorry,~ value~ '#1'~ is~ not~ available~ for~ 'Kashida'~ option~ yet~!
160 }
161
162 \msg_new:nnn { xepersian-hm } { error-specify-value-for-kashida-option }
163 {
164   Sorry,~ you~ must~ specify~ a~ value~ for~ 'Kashida'~ option~ yet~!
165 }
166
167 \msg_new:nnn { xepersian-hm } { warning-experimental-feature }
168 {
169   Please~ note~ that~ the~ feature~ '#1'~ is~ still~ experimental~
170   and~ is~ not~ regarded~ as~ stable.
171 }
172
173 \msg_new:nnn { xepersian-hm } { hm-series-font-not-found }
174 {
175   Either~ the~ font~'#1'~ is~ not~ installed~ on~ your~ system~ or~ does~ not~
176   belong~ to~ HM-Series-fonts.~
177   Please~ note~ that~ the~ option~ 'Kashida=leaders+glyph'~ is~ currently~ only~
178   supported~ by~ HM-Series-fonts.~
179   If~ you~ know~ of~ any~ other~ font~ that~ supports~ this~ option,~ please~
180   let~ me~ know~ to~ add~ it~ to~ the~ list~ of~ corresponding~ fonts.~
181 }
182
183 \keys_define:nn { xepersian-hm }
184 {
185   Kashida .code:n =
186   {
187     \tl_set:Nn \l_tmpa_tl { #1 }
188     \tl_case:NnTF \l_tmpa_tl
189     {
190       \l_xephm_stretch_glyph_tl
191       {
192         \msg_warning:nnn { xepersian-hm } { warning-experimental-feature } { Kashida=
193         \tl_set:Nx \l_xephm_gap_filler_tl { \l_xephm_stretch_glyph_tl }
194         \AtBeginDocument
195         {
196           \tl_set:Nx \l_xephm_main_font_full_tl { \tex_fontname:D \tex_the:D \tex_f
197           \tl_set:Nx \l_xephm_main_font_name_tl { \l_xephm_main_font_full_tl }
198           \regex_replace_once:nnN { ^"([~/]+)/.* } { \1 } \l_xephm_main_font_name_t
199         }
200         \bool_set_true:N \l_xephm_kashida_fix_bool
201       }
202       \l_xephm_stretch_leaders_glyph_tl
203       {
204         \tl_set:Nx \l_xephm_gap_filler_tl { \l_xephm_stretch_leaders_glyph_tl }
205         \bool_set_true:N \l_xephm_kashida_fix_bool
206       }
207       \l_xephm_stretch_leaders_hrule_tl
208       {
209         \tl_set:Nx \l_xephm_gap_filler_tl { \l_xephm_stretch_leaders_hrule_tl }
210         \bool_set_true:N \l_xephm_kashida_fix_bool
211       }

```

```

212     \l_xephm_stretch_off_tl
213     {
214         \tl_set:Nx \l_xephm_gap_filler_tl { \l_xephm_stretch_off_tl }
215         \bool_set_false:N \l_xephm_kashida_fix_bool
216     }
217     \l_xephm_stretch_on_tl
218     {
219         \tl_set:Nx \l_xephm_gap_filler_tl { \l_xephm_stretch_leaders_glyph_tl }
220         \bool_set_true:N \l_xephm_kashida_fix_bool
221     }
222     } { } { \tl_set:Nx \l_xephm_gap_filler_tl { #1 } }
223     \tl_if_empty:NT \l_xephm_gap_filler_tl { \msg_error:nn { xepersian-hm } { error-
specify-value-for-kashida-option } }
224     } ,
225
226     linebreakpenalty .code:n =
227     {
228         \int_set:Nn \l_tmpa_int { #1 }
229         \int_case:nnTF \l_tmpa_int
230         {
231             \l_xephm_min_penalty_int { \int_set:Nn \l_xephm_line_break_penalty_int { \l_xephm_min_penalty_int } }
232             \l_xephm_low_penalty_int { \int_set:Nn \l_xephm_line_break_penalty_int { \l_xephm_low_penalty_int } }
233             \l_xephm_med_penalty_int { \int_set:Nn \l_xephm_line_break_penalty_int { \l_xephm_med_penalty_int } }
234             \l_xephm_high_penalty_int { \int_set:Nn \l_xephm_line_break_penalty_int { \l_xephm_high_penalty_int } }
235             \l_xephm_max_penalty_int { \int_set:Nn \l_xephm_line_break_penalty_int { \l_xephm_max_penalty_int } }
236         } { } { \int_set:Nn \l_xephm_line_break_penalty_int { #1 } }
237         \bool_set_true:N \l_xephm_linebreakpenalty_bool
238     } ,
239
240     kashidastretch .code:n =
241     {
242         \tl_set:Nn \l_tmpa_tl { #1 }
243         \tl_case:NnTF \l_tmpa_tl
244         {
245             \l_xephm_fnt_kayhan_tl { \tl_set:Nn \l_xephm_skip_default_tl { 0.14 em pl } }
246             \l_xephm_fnt_kayhannavaar_tl { \tl_set:Nn \l_xephm_skip_default_tl { 0.129 em pl } }
247             \l_xephm_fnt_kayhanpook_tl { \tl_set:Nn \l_xephm_skip_default_tl { 0.133 em pl } }
248             \l_xephm_fnt_kayhansayeh_tl { \tl_set:Nn \l_xephm_skip_default_tl { 0.135 em pl } }
249             \l_xephm_fnt_khoramshahr_tl { \tl_set:Nn \l_xephm_skip_default_tl { 0.128 em pl } }
250             \l_xephm_fnt_khorramshahr_tl { \tl_set:Nn \l_xephm_skip_default_tl { 0.13 em pl } }
251             \l_xephm_fnt_niloofar_tl { \tl_set:Nn \l_xephm_skip_default_tl { 0.132 em pl } }
252             \l_xephm_fnt_paatch_tl { \tl_set:Nn \l_xephm_skip_default_tl { 0.127 em pl } }
253             \l_xephm_fnt_riyaz_tl { \tl_set:Nn \l_xephm_skip_default_tl { 0.125 em pl } }
254             \l_xephm_fnt_roya_tl { \tl_set:Nn \l_xephm_skip_default_tl { 0.142 em pl } }
255             \l_xephm_fnt_shafigh_tl { \tl_set:Nn \l_xephm_skip_default_tl { 0.143 em pl } }
256             \l_xephm_fnt_shafighKurd_tl { \tl_set:Nn \l_xephm_skip_default_tl { 0.126 em pl } }
257             \l_xephm_fnt_shafighUzbek_tl { \tl_set:Nn \l_xephm_skip_default_tl { 0.123 em pl } }
258             \l_xephm_fnt_shiraz_tl { \tl_set:Nn \l_xephm_skip_default_tl { 0.122 em pl } }
259             \l_xephm_fnt_sols_tl { \tl_set:Nn \l_xephm_skip_default_tl { 0.124 em pl } }
260             \l_xephm_fnt_tabriz_tl { \tl_set:Nn \l_xephm_skip_default_tl { 0.119 em pl } }
261             \l_xephm_fnt_titr_tl { \tl_set:Nn \l_xephm_skip_default_tl { 0.12 em pl } }
262             \l_xephm_fnt_titre_tl { \tl_set:Nn \l_xephm_skip_default_tl { 0.121 em pl } }
263             \l_xephm_fnt_traffic_tl { \tl_set:Nn \l_xephm_skip_default_tl { 0.124 em pl } }
264             \l_xephm_fnt_vahid_tl { \tl_set:Nn \l_xephm_skip_default_tl { 0.134 em pl } }

```

```

265         \l_xephm_fnt_vosta_tl      { \tl_set:Nn \l_xephm_skip_default_tl { 0.136 em pl
266         \l_xephm_fnt_yaghut_tl     { \tl_set:Nn \l_xephm_skip_default_tl { 0.138 em pl
267         \l_xephm_fnt_yagut_tl      { \tl_set:Nn \l_xephm_skip_default_tl { 0.137 em pl
268         \l_xephm_fnt_yas_tl        { \tl_set:Nn \l_xephm_skip_default_tl { 0.126 em pl
269         \l_xephm_fnt_yekan_tl      { \tl_set:Nn \l_xephm_skip_default_tl { 0.141 em pl
270         \l_xephm_fnt_yermook_tl    { \tl_set:Nn \l_xephm_skip_default_tl { 0.139 em pl
271         \l_xephm_fnt_zar_tl        { \tl_set:Nn \l_xephm_skip_default_tl { 0.116 em pl
272         \l_xephm_fnt_ziba_tl       { \tl_set:Nn \l_xephm_skip_default_tl { 0.119 em pl
273         \l_xephm_fnt_default_tl     { \tl_set:Nn \l_xephm_skip_default_tl { 0.14 em pl
274         \l_xephm_fnt_noskip_tl     { \tl_set:Nn \l_xephm_skip_default_tl { 0 em pl
275     } { } { \tl_set:Nn \l_xephm_skip_default_tl { #1 } }
276 } ,
277 kashidastretch .default:n = \tl_set:Nn \l_xephm_skip_default_tl { 0 em plus 0.5 em } ,
278
279 ligatures .code:n =
280 {
281     \tl_set:Nn \l_tmpa_tl { #1 }
282     \tl_case:NnTF \l_tmpa_tl
283     {
284         \l_xephm_lig_aalt_tl      { \tl_set:Nx \l_xephm_active_ligs_tl { \l_xephm_lig_aalt
285         \l_xephm_lig_ccmp_tl     { \tl_set:Nx \l_xephm_active_ligs_tl { \l_xephm_lig_ccmp
286         \l_xephm_lig_dlig_tl     { \tl_set:Nx \l_xephm_active_ligs_tl { \l_xephm_lig_dlig
287         \l_xephm_lig_fina_tl     { \tl_set:Nx \l_xephm_active_ligs_tl { \l_xephm_lig_fina
288         \l_xephm_lig_init_tl     { \tl_set:Nx \l_xephm_active_ligs_tl { \l_xephm_lig_init
289         \l_xephm_lig_locl_tl     { \tl_set:Nx \l_xephm_active_ligs_tl { \l_xephm_lig_locl
290         \l_xephm_lig_medi_tl     { \tl_set:Nx \l_xephm_active_ligs_tl { \l_xephm_lig_medi
291         \l_xephm_lig_rlig_tl     { \tl_set:Nx \l_xephm_active_ligs_tl { \l_xephm_lig_rlig
292         \l_xephm_lig_default_tl  { \tl_set:Nx \l_xephm_active_ligs_tl { \l_xephm_lig_defau
293     } { } { \tl_set:Nn \l_xephm_active_ligs_tl { #1 } }
294     \bool_set_true:N \l_xephm_ligature_bool
295 } ,
296 ligatures .default:n = \tl_set:Nn \l_xephm_active_ligs_tl { \l_xephm_lig_default_tl } ,
297
298 color .code:n =
299 {
300     \tl_set:Nn \l_tmpa_tl { #1 }
301     \tl_if_empty:NTF \l_tmpa_tl
302     {
303         \tl_set:Nx \l_xephm_color_tl { \l_xephm_col_default_tl }
304     }
305     {
306         \tl_set:Nx \l_xephm_color_tl { \l_tmpa_tl }
307     }
308     \bool_set_true:N \l_xephm_color_bool
309 } ,
310
311 }
312
313 \ProcessKeysOptions { xepersian-hm }
314
315 \bool_if:NTF \l_xephm_kashida_fix_bool
316 {
317     \tex_input:D { xepersian-hm-kashida.tex }
318

```



```

319 \NewDocumentCommand \KashidaHMFixOn {} { \bool_set_true:N \l_xephm_kashida_fix_bool }
320 \NewDocumentCommand \KashidaHMFixOff {} { \bool_set_false:N \l_xephm_kashida_fix_bool }
321
322 \tex_let:D \KashidaOn \KashidaHMFixOn
323 \tex_let:D \KashidaOff \KashidaHMFixOff
324
325 \tl_if_empty:NT \l_xephm_skip_default_tl { \tl_set:Nn \l_xephm_skip_default_tl { 0.14 em
326 }
327 {
328 \tl_set:NV \l_xephm_skip_default_tl \c_xephm_skip_a_tl
329 }
330
331 \makeatletter
332 \newif\if@Kashida@on
333 %% Becuase Vafa Khalighi has copied the above code (injecting the character uni+200E) in xepersian
334 23.0
335 %% (https://tug.org/svn/texlive/trunk/Master/texmf-dist/tex/xelatex/xepersian/kashida-
336 xepersian.def?revision=55165&view=co),
337 %% the following line of code is not needed in xepersian anymore.
338 %% % \newif\if@Kashida@XB@fix
339 \makeatother
340
341 \endinput
342 </xepersian-hm-sty>

```

1.2 File: xepersian-hm-kashida.tex

```

341 <*xepersian-hm-kashida.tex>
342 \ProvidesExplFile {xepersian-hm-kashida.tex} {2020-09-04} {1.1a} { Fixes-implementation-of-Ka
343
344 \tex_input:D { xepersian-hm-common-kashida.tex }
345
346 \newXeTeXintercharclass \c_xephm_d_charclass % dual-joiner class
347 \newXeTeXintercharclass \c_xephm_l_charclass % lam
348 \newXeTeXintercharclass \c_xephm_r_charclass % right-joiner
349 \newXeTeXintercharclass \c_xephm_a_charclass % alef
350 \newXeTeXintercharclass \c_xephm_y_charclass % yeh
351
352 \tl_set:Nn \l_xephm_use_color_tl
353 {
354 \bool_if:NTF \l_xephm_color_bool
355 {
356 \colorlet{default}{\l_xephm_color_tl}
357 }
358 {
359 \colorlet{default}{.}
360 }
361 \color{default}
362 }
363
364 %% Partly adapted from the code provided by David Carlisle in:
365 %% https://tex.stackexchange.com/questions/356709/how-to-know-the-width-and-fill-
366 the-glue-space-between-two-characters-when-using/356721#356721
367 \cs_new:Npn \xephm_kashida_glyph #1
368 {

```

```

368 \bool_if:NT \l_xephm_kashida_fix_bool
369 {
370   \c_xephm_lrm_int\tex_penalty:D 10000
371   \mode_leave_vertical:
372   \tex_global:D \tex_advance:D \l_xephm_counter_int \c_one_int
373
374   \tl_set:Nx \l_xephm_pos_tl { p\tex_romannumeral:D \l_xephm_counter_int }
375   \tl_set:Nx \l_xephm_zref_tl { z\tex_romannumeral:D \l_xephm_counter_int }
376
377   \zsaveposx{x_i\l_xephm_zref_tl}
378   \tl_set:Nx \l_tmpa_tl
379     {
380       \iow_now:cx { @auxout }
381       {
382         \token_to_str:N \gdef \exp_after:wN \token_to_str:N \cs:w xi\l_xephm_pos_tl \cs_end
383       }
384     }
385   \l_tmpa_tl
386   \skip_horizontal:n { #1 }
387   \zsaveposx{x_f\l_xephm_zref_tl}
388   \tl_set:Nx \l_tmpa_tl
389     {
390       \iow_now:cx { @auxout }
391       {
392         \token_to_str:N \gdef \exp_after:wN \token_to_str:N \cs:w xf\l_xephm_pos_tl \cs_end
393       }
394     }
395   \l_tmpa_tl
396   \exp_after:wN
397   \if_meaning:w
398     \cs:w xi\l_xephm_pos_tl \cs_end: \tex_relax:D
399   \else:
400     \dim_set:Nn \l_xephm_diff_pos_dim
401     {
402       \dim_eval:n { \cs:w xi\l_xephm_pos_tl \cs_end: sp - \cs:w xf\l_xephm_pos_tl \cs_end
403     }
404     \dim_compare:nTF { \l_xephm_diff_pos_dim == 0sp }
405     { }
406     { \llap { \resizebox { \l_xephm_diff_pos_dim \tex_relax:D } { \height } { \l_xephm_us
407   \fi:
408 }
409 }
410
411 \cs_new:Npn \xephm_kashida_leaders #1
412 {
413   \bool_if:NT \l_xephm_kashida_fix_bool
414   {
415     \tl_if_eq:NNTF \l_xephm_gap_filler_tl \l_xephm_stretch_leaders_glyph_tl
416     {
417       \tl_set:Nx \l_xephm_font_full_tl { \tex_fontname:D \tex_the:D \tex_font:D }
418       \tl_set:Nx \l_xephm_font_name_tl { \l_xephm_font_full_tl }
419       \tl_set:Nx \l_xephm_font_init_tl { \l_xephm_font_name_tl }
420       \regex_replace_once:nnN { ^"[\? (HM) [\_ \ ] (X|F) .* } { \1\2 } \l_xephm_font_init_tl\
421       \tl_set:Nn \l_tmpa_tl { HMF }

```

```

422     \tl_set:Nn \l_tmpb_tl { HMX }
423     \bool_if:nTF { \str_if_eq_p:NN { \l_xephm_font_init_tl } { \l_tmpa_tl } || \str_if_
424         {
425         \hbox_set:Nn \l_xephm_ksh_box { \l_xephm_use_color_tl \XeTeXglyph\XeTeXglyphin
426         \c_xephm_zwj_int \tex_penalty:D 10000
427         \tex_leaders:D \copy\l_xephm_ksh_box \skip_horizontal:n { #1 }
428         \c_xephm_zwj_int
429         }
430         {
431         \msg_error:nxx { xepersian-hm } { hm-series-font-not-found } { \l_xephm_font_na
432         }
433     }
434     {
435     %% Partly adapted from the code provided by Jonathan Kew in:
436     %% https://tug.org/pipermail/xetex/2009-February/012307.html.
437     %% Somebody notified me that the code in 'kashida-xepersian.def' from xepersian
438     %% package is an exact copy of Jonathan Kew's code. Being unaware of this, in
439     %% the earlier versions of this package I made a mistake and acknowledged
440     %% Vafa Khalighi instead of Jonathan Kew. A sincere thank you to Jonathan Kew
441     %% for his excellent code.
442     \c_xephm_lrm_int\c_xephm_zwj_int
443     {\l_xephm_use_color_tl\tex_penalty:D 10000
444     \tex_leaders:D \tex_hrulerule:D height \XeTeXglyphbounds \c_xephm_two_int
445     \int_use:N \XeTeXcharglyph \c_xephm_ksh_int depth \XeTeXglyphbounds \c_xephm_four_i
446     \int_use:N \XeTeXcharglyph \c_xephm_ksh_int \skip_horizontal:n { #1 }
447     }
448     \c_xephm_zwj_int
449     }
450     }
451 }
452
453 \XeTeXinterchartokenstate = 1
454
455 \clist_set:Nn \l_xephm_a_clist { 0622,0623,0625,0627 } %
456 \clist_map_inline:Nn \l_xephm_a_clist
457 {
458     \XeTeXcharclass "#1 \c_xephm_a_charclass
459 }
460
461 \clist_set:Nn \l_xephm_d_clist { 0626,0628,062A,062B,062C,062D,062E,0633,0634,0635,0636,0637,
462 \clist_map_inline:Nn \l_xephm_d_clist
463 {
464     \XeTeXcharclass "#1 \c_xephm_d_charclass
465 }
466
467 \clist_set:Nn \l_xephm_l_clist { 0644 } %
468 \clist_map_inline:Nn \l_xephm_l_clist
469 {
470     \XeTeXcharclass "#1 \c_xephm_l_charclass
471 }
472
473 \clist_set:Nn \l_xephm_r_clist { 0624,0629,062F,0630,0631,0632,0648,0698 } % , , , , , ,
474 \clist_map_inline:Nn \l_xephm_r_clist
475 {

```

```

476 \XeTeXcharclass "#1 \c_xephm_r_charclass
477 }
478
479 \clist_set:Nn \l_xephm_y_clist { 0649,064A,06CC } % ,,
480 \clist_map_inline:Nn \l_xephm_y_clist
481 {
482   \XeTeXcharclass "#1 \c_xephm_y_charclass
483 }
484
485 \tl_if_eq:NNTF \l_xephm_gap_filler_tl \l_xephm_stretch_glyph_tl {
486   \XeTeXinterchartoks \c_xephm_y_charclass \c_xephm_y_charclass = {
487     \bool_if:NTF \l_xephm_kashida_fix_bool
488     { \c_xephm_zwj_int \xephm_kashida_glyph \l_xephm_skip_default_tl \c_xephm_zwj_int }
489     { \c_xephm_zwj_int \xephm_kashida_glyph \c_xephm_skip_a_tl \c_xephm_zwj_int }
490   }
491   \XeTeXinterchartoks \c_xephm_d_charclass \c_xephm_y_charclass = {
492     \bool_if:NTF \l_xephm_kashida_fix_bool
493     { \c_xephm_zwj_int \xephm_kashida_glyph \l_xephm_skip_default_tl \c_xephm_zwj_int }
494     { \c_xephm_zwj_int \xephm_kashida_glyph \c_xephm_skip_a_tl \c_xephm_zwj_int }
495   }
496   \XeTeXinterchartoks \c_xephm_y_charclass \c_xephm_d_charclass = { \c_xephm_zwj_int \xephm_k
497   \XeTeXinterchartoks \c_xephm_d_charclass \c_xephm_d_charclass = { \c_xephm_zwj_int \xephm_k
498   \XeTeXinterchartoks \c_xephm_l_charclass \c_xephm_d_charclass = { \c_xephm_zwj_int \xephm_k
499   \XeTeXinterchartoks \c_xephm_d_charclass \c_xephm_l_charclass = { \c_xephm_zwj_int \xephm_k
500   \XeTeXinterchartoks \c_xephm_l_charclass \c_xephm_l_charclass = { \c_xephm_zwj_int \xephm_k
501   \XeTeXinterchartoks \c_xephm_d_charclass \c_xephm_r_charclass = { \c_xephm_zwj_int \xephm_k
502   \XeTeXinterchartoks \c_xephm_d_charclass \c_xephm_a_charclass = { \c_xephm_zwj_int \xephm_k
503   \XeTeXinterchartoks \c_xephm_l_charclass \c_xephm_r_charclass = { \c_xephm_zwj_int \xephm_k
504   \XeTeXinterchartoks \c_xephm_l_charclass \c_xephm_a_charclass = { }
505 }
506 {
507   \bool_if:nTF {
508     \tl_if_eq_p:NN \l_xephm_gap_filler_tl \l_xephm_stretch_leaders_glyph_tl ||
509     \tl_if_eq_p:NN \l_xephm_gap_filler_tl \l_xephm_stretch_leaders_hrule_tl
510   }
511   {
512     \XeTeXinterchartoks \c_xephm_y_charclass \c_xephm_y_charclass = {
513       \bool_if:NTF \l_xephm_kashida_fix_bool
514       { \xephm_kashida_leaders \l_xephm_skip_default_tl }
515       { \xephm_kashida_leaders \c_xephm_skip_a_tl }
516     }
517     \XeTeXinterchartoks \c_xephm_d_charclass \c_xephm_y_charclass = {
518       \bool_if:NTF \l_xephm_kashida_fix_bool
519       { \xephm_kashida_leaders \l_xephm_skip_default_tl }
520       { \xephm_kashida_leaders \c_xephm_skip_a_tl }
521     }
522     \XeTeXinterchartoks \c_xephm_y_charclass \c_xephm_d_charclass = { \xephm_kashida_leaders
523     \XeTeXinterchartoks \c_xephm_d_charclass \c_xephm_d_charclass = { \xephm_kashida_leaders
524     \XeTeXinterchartoks \c_xephm_l_charclass \c_xephm_d_charclass = { \xephm_kashida_leaders
525     \XeTeXinterchartoks \c_xephm_d_charclass \c_xephm_l_charclass = { \xephm_kashida_leaders
526     \XeTeXinterchartoks \c_xephm_l_charclass \c_xephm_l_charclass = { \xephm_kashida_leaders
527     \XeTeXinterchartoks \c_xephm_d_charclass \c_xephm_r_charclass = { \xephm_kashida_leaders
528     \XeTeXinterchartoks \c_xephm_d_charclass \c_xephm_a_charclass = { \xephm_kashida_leaders
529     \XeTeXinterchartoks \c_xephm_l_charclass \c_xephm_r_charclass = { \xephm_kashida_leaders

```



```

582         or \space <return> \space to continue without it.}
583 \else:
584     \l_tmpa_skip \tex_lastskip:D
585     \tex_unskip:D
586     \tex_penalty:D -\l_xephm_line_break_penalty_int
587     \dim_compare:nT { \l_tmpa_skip > \c_zero_skip }
588     { \skip_horizontal:N \l_tmpa_skip \tex_ignorespaces:D }
589 \fi:
590 }
591
592 \NewDocumentCommand { \discouragebadlinebreaks } { 0{\l_xephm_line_break_penalty_int} 0{\c_
593 {
594     \IfNoValueF {#1}
595     { \int_set:Nn \l_xephm_line_break_penalty_int {#1} }
596     \IfNoValueF {#2}
597     { \tl_set:Nn \l_xephm_skip_default_tl {#2} }
598     \xephm_put_line_breaks:n { #3 }
599 }
600
601 \cs_new_protected:Nn \xephm_put_line_breaks:n
602 {
603     \tl_set:Nn \l_xephm_line_break_tl { #1 }
604     \regex_replace_all:nnN { ([\])+ } { \ \ 0 \ \ \c{xephm_line_break:}\ } \l_xephm_line_brea
605     \tl_use:N \l_xephm_line_break_tl
606 }
607 }
608
609 \endinput
610 </xepersian-hm-common-kashida-tex>

```

2 Acknowledgments

In the first place I have to thank Donald Knuth for inventing TeX. During the development of this package I referred to Stack Exchange network of question-and-answer (Q&A) websites to solve problems for which I am grateful. I also would like to thank the developer teams of TeX's friends especially LaTeX, LuaTeX and XeTeX teams.

3 Change History

2020-03-17 v0.1

- First standalone version.

2020-03-19 v0.2

- Some minor changes.

2020-03-23 v0.3

- Added documentation

2020-03-26 v0.4

- Added DTX and INS files

2020-04-20 v0.5a

- Added `Kashida=glyph` to the package options
- Added `\discouragebadlinebreaks` to prevent bad line breaks in a text (less than a paragraph).

2020-04-26 v0.5b

- If the `kashida` glyph is used to stretch the text, then the main font is checked for availability of the `kashida` character.
- Two optional arguments added in `\discouragebadlinebreaks` macro.

2020-05-01 v0.5c

- The documentation is divided into two parts: User documentation and Source code documentation.
- The example file is divided into two parts: An example for the option `Kashida=glyph` and the other one for `Kashida=hrule`.

2020-05-07 v0.5d

- Added a new section to the user documentation titled: Brief outline of the issues.
- An error is triggered if the value of `Kashida` option is not specified.

2020-05-12 v0.5e

- Disturbance of ligatures caused by activation of `kashida` is fixed.

2020-05-20 v0.5f

- Reduced the length of items created by `zref` package in `\jobname.aux` file.
- If the option `Kashida=glyph` is used, a warning message is added to the log file which emphasizes that the option is experimental.

2020-05-31 v1.0a

- The option `'Kashida=hrule'` is changed to `'Kashida=leaders+hrule'`.
- The option `'Kashida=leaders+glyph'` is added. Since with this option `kashida` glyph leaders are used to stretch the text, nothing is saved in `.aux` files.

2020-06-04 v1.0b

- Added two options `Kashida=On` and `Kashida=Off`.

2020-07-20 v1.0c

- Fixed the problem with the option `ligatures=default`.

2020-08-05 v1.0d

- In the new version of LaTeX3, the name of some of the XeTeX control sequences has changed. These changes are applied in this version of `xepersion-hm`.

2020-09-04 v1.1a

- Added the `color` option.
- Added a few sections to the User Documentation.
- Split the file containing `kashida` code.

To Do's

To do

References:

(Actually, this is not a “References” nor a “Literature”, but the most important although not a complete list of “Resources Used” to develop this package.)

- [1] Donald E. Knuth, *The T_EX book*, Addison-Wesley, 1986.
- [2] Victor Eijkhout, *T_EX BY TOPIC*, Addison-Wesley, 2013.
- [3] Paul W. Abrahams, Kathryn A. Hargreaves, and Karl Berry, *T_EX for the Impatient*, Addison-Wesley, 2013.
- [4] Leslie Lamport, *L^AT_EX, A document preparation System*, Addison-Wesley, 1986.
- [5] Frank Mittelbach and Michel Goossens with Johannes Braams, David Carlisle, and Chris Rowley, *The L^AT_EX Companion*, Addison-Wesley, second edition, 2004.
- [6] Package `latex`: The LaTeX Team, *The L^AT_EX 2_ε Sources*, [CTAN:macros/latex/base/source2e.pdf](#), 2020-02-02
- [7] Package `l3kernel`: The LaTeX3 Team, *The L^AT_EX3 Sources*, [CTAN:macros/latex/contrib/l3kernel/source3.pdf](#), 2020-07-17
- [8] Package `l3kernel`: The LaTeX3 Team, *The L^AT_EX3 Interfaces*, [CTAN:macros/latex/contrib/l3kernel/interface3.pdf](#), 2020-07-17
- [9] Package `xetexref`: Will Robertson, Khaled Hosny, and Karl Berry, X_ƎT_EX reference guide, [CTAN:info/xetexref/xetex-reference.pdf](#), 2019-12-09
- [10] Package `xetex`: Jonathan Kew, About X_ƎT_EX, [CTAN:systems/doc/xetex/XeTeX-notes.pdf](#), 2005-10-17

- [11] Package `xetex`: Michel Goossens, The X_YTeX Companion, <http://xml.web.cern.ch/XML/lgc2/xetexmain.pdf>, 2009-08-19
- [12] Website: Stack Exchange: Hot Questions, T_EX-L^AT_EX Q&A for users of TeX, LaTeX, ConTeXt, and related typesetting systems, tex.stackexchange.com

Index

The *italic* numbers denote the pages where the corresponding entry is described, numbers underlined point to the definition, all others indicate the places where it is used.

	Symbols		
<code>\[</code>	420	
<code>_</code>	420	
	Numbers		
<code>\1</code>	198, 420	
<code>\2</code>	420	
<code>\u</code>	420, 563	
	A		
<code>\AtBeginDocument</code>	194	
	B		
bool commands:			
<code>\bool_if:NTF</code>	315, 354, 368, 413, 487, 492, 513, 518, 542, 567	
<code>\bool_if:nTF</code>	423, 507, 553	
<code>\bool_set_false:N</code>	69, 71, 72, 73, 215, 320	
<code>\bool_set_true:N</code>	200, 205, 210, 220, 237, 294, 308, 319	
box commands:			
<code>\box_new:N</code>	25	
	C		
<code>\char</code>	563	
clist commands:			
<code>\clist_count:N</code>	546	
<code>\clist_item:Nn</code>	551, 552	
<code>\clist_map_inline:Nn</code>	456, 462, 468, 474, 480, 558	
<code>\clist_new:N</code>	544	
<code>\clist_put_left:Nn</code>	555	
<code>\clist_set:Nn</code>	130, 131, 132, 133, 134, 135, 136, 137, 138, 140, 455, 461, 467, 473, 479, 552	
<code>\l_tmpa_clist</code>	552, 555	
<code>\color</code>	361	
			<code>\colorlet</code> 356, 359
			<code>\copy</code> 427
			cs commands:
			<code>\cs:w</code> 382, 392, 398, 402
			<code>\cs_end:</code> 382, 392, 398, 402
			<code>\cs_new:Nn</code> 570
			<code>\cs_new:Npn</code> 366, 411
	D		
			<code>\def</code> 23
			dim commands:
			<code>\dim_compare:nTF</code> 404, 587
			<code>\dim_eval:n</code> 402
			<code>\dim_new:N</code> 67
			<code>\dim_set:Nn</code> 400
			<code>\discouragebadlinebreaks</code> 592
	E		
			else commands:
			<code>\else:</code> 399, 583
			<code>\endinput</code> 14, 339, 537
			exp commands:
			<code>\exp_after:wN</code> 382, 392, 396
	F		
			fi commands:
			<code>\fi:</code> 407, 589
	G		
			<code>\gdef</code> 382, 392
			<code>\GenericError</code> 572
	H		
			hbox commands:
			<code>\hbox_set:Nn</code> 425
			<code>\height</code> 406
			HM commands:
			HM: 1

	I	\ProvidesExplPackage	21
if commands:			
\if_meaning:w	397		
\if_mode_vertical:	571		
int commands:			
\int_case:nnTF	229		
\int_const:Nn	27, 28, 29, 31, 32		
\int_eval:n	550		
\int_new:N			
.	37, 39, 41, 43, 44, 45, 46, 47, 545		
\int_set:Nn			
.	75, 76, 77, 78, 79, 228, 231, 232, 233, 234, 235, 236, 546, 549, 550		
\int_step_inline:nnnn	547		
\int_use:N	445, 446		
\c_one_int	372		
\l_tmpa_int	228, 229, 549, 550, 551		
\l_tmpb_int	550, 552		
iow commands:			
\iow_now:Nn	380, 390		
	K		
\KashidaHMFixOff	320, 323		
\KashidaHMFixOn	319, 322		
\KashidaOff	323		
\KashidaOn	322		
keys commands:			
\keys_define:nn	183		
	L		
\llap	406		
	M		
\makeatletter	331		
\makeatother	337		
\MessageBreak	9, 10, 11, 579, 581		
mode commands:			
\mode_leave_vertical:	371		
msg commands:			
\msg_error:nn	223		
\msg_error:nnn	431, 533		
\msg_new:nnn	152, 157, 162, 167, 173		
\msg_warning:nnn	192		
	N		
\NeedsTeXFormat	2		
\NewDocumentCommand	319, 320, 592		
\newif	332, 336		
\newXeTeXintercharclass			
.	346, 347, 348, 349, 350		
	P		
\PackageError	7		
\ProcessKeysOptions	313		
\ProvidesExplFile	342, 540		
	R		
regex commands:			
\regex_replace_once:nnN	198, 420		
\relax	420		
\RequirePackage	3, 16, 17, 18, 19, 20		
\resizebox	406		
	S		
seq commands:			
\seq_pop_left:NN	561, 562		
\seq_set_split:Nnn	560		
\l_tmpa_seq	560, 561, 562		
skip commands:			
\skip_horizontal:N	588		
\skip_horizontal:n	386, 427, 446		
\l_tmpa_skip	584, 587, 588		
\c_zero_skip	587		
\space	573, 580, 582		
str commands:			
\str_if_eq_p:NN	423		
	T		
TeX and L ^A T _E X 2 _ε commands:			
\@ifpackagelater	4		
@if@Kashida@on	332		
@if@Kashida@XB@fix	336		
tex commands:			
\tex_advance:D	372		
\tex_font:D	196, 417		
\tex_fontname:D	196, 417		
\tex_global:D	372		
\tex_hrule:D	444		
\tex_ignorespaces:D	588		
\tex_input:D	317, 344		
\tex_lastskip:D	584		
\tex_leaders:D	427, 444		
\tex_let:D	322, 323		
\tex_penalty:D	370, 426, 443, 586		
\tex_relax:D	398, 406		
\tex_romannumeral:D	374, 375		
\tex_the:D	196, 417		
\tex_unskip:D	585		
tl commands:			
\tl_case:NnTF	188, 243, 282		
\tl_const:Nn	34, 35, 563		
\tl_if_empty:NTF	223, 301, 325		
\tl_if_eq:NNTF	415, 485		
\tl_if_eq_p:NN	508, 509, 553		
\tl_new:N	49,		
.	51, 52, 54, 55, 57, 59, 61, 63, 64, 65		
\tl_set:Nn	81, 82, 83, 84, 85,		
.	87, 88, 89, 90, 91, 92, 93, 94, 95, 96,		

97, 98, 99, 100, 101, 102, 103, 104,	
105, 106, 107, 108, 109, 110, 111,	
112, 113, 114, 115, 116, 118, 119,	
120, 121, 122, 123, 124, 125, 126,	
128, 187, 193, 196, 197, 204, 209,	
214, 219, 222, 242, 245, 246, 247,	
248, 249, 250, 251, 252, 253, 254,	
255, 256, 257, 258, 259, 260, 261,	
262, 263, 264, 265, 266, 267, 268,	
269, 270, 271, 272, 273, 274, 275,	
277, 281, 284, 285, 286, 287, 288,	
289, 290, 291, 292, 293, 296, 300,	
303, 306, 325, 328, 352, 374, 375,	
378, 388, 417, 418, 419, 421, 422, 551	
\tl_use:N	563
\l_tmpa_tl	187, 188, 242, 243,
281, 282, 300, 301, 306, 378, 385,	
388, 395, 421, 423, 551, 553, 561, 563	
\l_tmpb_tl	422, 423, 562, 563
token commands:	
\token_to_str:N	382, 392
X	
\XePersianHM	23
xephm commands:	
\c_xephm_a_charclass	
.	349, 458, 502, 504, 528, 530
\l_xephm_a_clist	455, 456
\l_xephm_active_ligs_tl	
.	59, 284, 285, 286, 287,
288, 289, 290, 291, 292, 293, 296, 553	
\l_xephm_col_default_tl	128, 303
\l_xephm_color_bool	73, 308, 354
\l_xephm_color_rgb_tl	65
\l_xephm_color_tl	64, 303, 306, 356
\l_xephm_counter_int	37, 372, 374, 375
\c_xephm_d_charclass	346,
464, 491, 496, 497, 498, 499, 501,	
502, 517, 522, 523, 524, 525, 527, 528	
\l_xephm_d_clist	461, 462
\l_xephm_diff_pos_dim	67, 400, 404, 406
\l_xephm_fnt_default_tl	115, 273
\l_xephm_fnt_kayhan_tl	87, 245
\l_xephm_fnt_kayhannavaar_tl	88, 246
\l_xephm_fnt_kayhanpook_tl	89, 247
\l_xephm_fnt_kayhansayeh_tl	90, 248
\l_xephm_fnt_khoramshahr_tl	91, 249
\l_xephm_fnt_khorramshahr_tl	92, 250
\l_xephm_fnt_niloofar_tl	93, 251
\l_xephm_fnt_noskip_tl	116, 274
\l_xephm_fnt_paatch_tl	94, 252
\l_xephm_fnt_riyaz_tl	95, 253
\l_xephm_fnt_roya_tl	96, 254
\l_xephm_fnt_shafigh_tl	97, 255
\l_xephm_fnt_shafighKurd_tl	98, 256
\l_xephm_fnt_shafighUzbek_tl	99, 257
\l_xephm_fnt_shiraz_tl	100, 258
\l_xephm_fnt_sols_tl	101, 259
\l_xephm_fnt_tabriz_tl	102, 260
\l_xephm_fnt_titr_tl	103, 261
\l_xephm_fnt_titre_tl	104, 262
\l_xephm_fnt_traffic_tl	105, 263
\l_xephm_fnt_vahid_tl	106, 264
\l_xephm_fnt_vosta_tl	107, 265
\l_xephm_fnt_yaghut_tl	108, 266
\l_xephm_fnt_yagut_tl	109, 267
\l_xephm_fnt_yas_tl	110, 268
\l_xephm_fnt_yekan_tl	111, 269
\l_xephm_fnt_yermook_tl	112, 270
\l_xephm_fnt_zar_tl	113, 271
\l_xephm_fnt_ziba_tl	114, 272
\l_xephm_font_full_tl	54, 417, 418
\l_xephm_font_init_tl	419, 420, 423
\l_xephm_font_name_tl	55, 418, 419, 431
\c_xephm_four_int	32, 445
\l_xephm_gap_filler_tl	
.	61, 193, 204, 209, 214,
219, 222, 223, 415, 485, 508, 509, 533	
\l_xephm_high_penalty_int	46, 78, 234
\l_xephm_kashida_fix_bool	
.	69, 200, 205, 210, 215, 220, 315,
319, 320, 368, 413, 487, 492, 513, 518	
\xephm_kashida_glyph	
.	366, 488, 489, 493, 494,
496, 497, 498, 499, 500, 501, 502, 503	
\xephm_kashida_leaders	
.	411, 514, 515, 519, 520,
522, 523, 524, 525, 526, 527, 528, 529	
\l_xephm_kashida_slot_int	39
\l_xephm_ksh_box	25, 425, 427
\c_xephm_ksh_int	27, 406, 445, 446
\c_xephm_l_charclass	
.	347, 470, 498, 499,
500, 503, 504, 524, 525, 526, 529, 530	
\l_xephm_l_clist	467, 468
\l_xephm_lig_aalt_clist	130, 142
\l_xephm_lig_aalt_tl	118, 142, 284
\l_xephm_lig_ccmp_clist	131, 143
\l_xephm_lig_ccmp_tl	119, 143, 285
\l_xephm_lig_default_clist	138
\l_xephm_lig_default_tl	
.	126, 292, 296, 553
\l_xephm_lig_dlig_clist	132, 144
\l_xephm_lig_dlig_tl	120, 144, 286
\l_xephm_lig_fina_clist	133, 145
\l_xephm_lig_fina_tl	121, 145, 287
\l_xephm_lig_init_clist	134, 146
\l_xephm_lig_init_tl	122, 146, 288

<code>\l_xephm_lig_locl_clist</code>	135, 147	247, 248, 249, 250, 251, 252, 253,
<code>\l_xephm_lig_locl_tl</code>	123, 147, 289	254, 255, 256, 257, 258, 259, 260,
<code>\l_xephm_lig_medi_clist</code>	136, 148	261, 262, 263, 264, 265, 266, 267,
<code>\l_xephm_lig_medi_tl</code>	124, 148, 290	268, 269, 270, 271, 272, 273, 274,
<code>\l_xephm_lig_names_clist</code>		275, 277, 325, 328, 488, 493, 514, 519
.	140, 546, 551, 552	
<code>\l_xephm_lig_names_len_int</code>		<code>\l_xephm_stretch_glyph_tl</code>
.	545, 546, 547 81, 190, 193, 485
<code>\l_xephm_lig_rlig_clist</code>	137, 149	<code>\l_xephm_stretch_leaders_glyph_</code>
<code>\l_xephm_lig_rlig_tl</code>	125, 149, 291	<code>tl</code> 82, 202, 204, 219, 415, 508
<code>\l_xephm_ligature_bool</code>	71, 294, 542	<code>\l_xephm_stretch_leaders_hrule_</code>
<code>\l_xephm_ligatures_clist</code> 544, 555, 558		<code>tl</code> 83, 207, 209, 509
<code>\xephm_line_break:</code>	570	<code>\l_xephm_stretch_off_tl</code>
<code>\l_xephm_line_break_penalty_int</code>		<code>\l_xephm_stretch_on_tl</code> 85, 217
.	41, 231, 232, 233, 234, 235, 236, 586	<code>\c_xephm_two_int</code> 31, 444
<code>\l_xephm_line_break_tl</code>	49	<code>\l_xephm_use_color_tl</code>
<code>\l_xephm_linebreakpenalty_bool</code> 63, 352, 406, 425, 443
.	72, 237, 567	<code>\c_xephm_y_charclass</code>
<code>\l_xephm_low_penalty_int</code>	44, 76, 232 350, 482, 486, 491, 496, 512, 517, 522
<code>\c_xephm_lrm_int</code>	28, 370, 442	<code>\l_xephm_y_clist</code> 479, 480
<code>\l_xephm_main_font_full_tl</code>		<code>\l_xephm_zref_tl</code> 375, 377, 382, 387, 392
.	51, 196, 197	<code>\c_xephm_zwj_int</code> 29, 426,
<code>\l_xephm_main_font_name_tl</code> 428, 442, 448, 488, 489, 493, 494,
.	52, 197, 198 496, 497, 498, 499, 500, 501, 502, 503
<code>\l_xephm_max_penalty_int</code>	47, 79, 235	<code>\XeTeXcharclass</code> 458, 464, 470, 476, 482
<code>\l_xephm_med_penalty_int</code>	45, 77, 233	<code>\XeTeXcharglyph</code> 445, 446
<code>\l_xephm_min_penalty_int</code>	43, 75, 231	<code>\XeTeXglyph</code> 425
<code>\l_xephm_pos_tl</code> 374, 382, 392, 398, 402		<code>\XeTeXglyphbounds</code> 444, 445
<code>\c_xephm_r_charclass</code>		<code>\XeTeXglyphindex</code> 425
.	348, 476, 501, 503, 527, 529	<code>\XeTeXinterchartokenstate</code> 453
<code>\l_xephm_r_clist</code>	473, 474	<code>\XeTeXinterchartoks</code>
<code>\c_xephm_skip_a_tl</code> 486, 491, 496, 497, 498, 499, 500,
.	34, 328, 489, 494, 496, 497, 498, 501, 502, 503, 504, 512, 517, 522,
.	499, 500, 501, 502, 503, 515, 520, 523, 524, 525, 526, 527, 528, 529, 530
.	522, 523, 524, 525, 526, 527, 528, 529	
<code>\c_xephm_skip_b_tl</code>	35	
<code>\l_xephm_skip_default_tl</code>		
.	57, 245, 246,	
		Z
		<code>\zpspx</code> 382, 392
		<code>\zsaveposx</code> 377, 387