

Evangelion Japanese Font Metric for LuaTeX

<https://github.com/RadioNoiseE/Evangelion-JFM>

西历 2023 年 黄京

概要

本文档将介绍名为 Evangelion Japanese Font Metric（下简称为“Eva-JFM”）的 JFM 文件。其适用于简体中文（以下简称为「简中」、繁体中文（以下简称为「繁中」）及日文字体的横直排。旨在提供一个充分利用 LuaTeX-japanese 的 priority 特性，基于标准 [1] 的同时，支持一些罕用特性的 JFM 文件。文档暂仅使用中文撰写。

1 背景及略介

TeX 是高德纳教授于 20 世纪末开发的强大排版引擎，能够完全满足西文排版的需求。然因时代局限性^{*1}以及客观原因^{*2}对中日排版支持十分有限。为达成中日排版需求，在宏扩展（如 CJK 等）之外出现了引擎扩展。影响力较大的是 pTeX 系列。

pTeX 系列采用虚拟字体的理念，使用 TFM/VF 映射 TrueType 或 OpenType 字体完成排版。其不支持宏配置字体，也不支持直接生成 PDF 格式文件。但可以满足日本的传统横纵排版需求（工业标准）。

pdfTeX 则是当时另一个 TeX 的引擎扩展，支持不经 DVI 格式直接输出 PDF 格式的文件。然对 Unicode（字符编码）及 TrueType、OpenType（「现代」矢量字体格式）的支持繁琐或有限。

LuaTeX 便是基于 pdfTeX 的引擎扩展，在原生支持 Unicode 下提供 Lua 语言扩展（使能够使用 fontloader 等模块）支持现代字体。宏配置字体特性由 luaotfload 宏集提供。它支持直接生成 PDF 文件。

LuaTeX-japanese 可看作是对两者的合并。这是一个由日本开发者北川弘典首倡的 LuaTeX 下的日文支持项目，即将 pTeX（大部分）移植到 LuaTeX 下。由于 LuaTeX 支持宏配置字体，故不需要 VF 文件为字体提供映射，但为标点挤压等需求保留并扩展^{*3}了 JFM 文件。

本项目就是一个 JFM 文件。使用 LuaTeX 的 callback，将简中、繁中、日文及横纵方向、行间标点、悬挂标点、压缩字体等特性集中于 jfm-eva.lua 单个文件中。用户可按需调用特性来完成高质量的中日排版。

2 安裝及本地配置

本项目将源文件托管于 GitHub 平台，且已上传至 Comprehensive TeX Archive Net (CTAN)。用户可使用

```
tlmgr install evangelion-jfm
```

或使用其他包管理器安装。用户也可使用

^{*1} 如没有事实上的统一字符编码等。

^{*2} 如中日字符集较大，以及书写方式的不同（纵书、横书），标点等。

^{*3} 如优先挤压（priority）特性，及一些特殊字符（如 parbdd、glue）等。

```
1 mkdir Evangelion-JFM [ && ] cd Evangelion-JFM
2 git clone https://github.com/RadioNoiseE/Evangelion-JFM
```

获取源文件，再将其放置在本地的 TEXMF 路径中，如

```
1 ~/Library/texlive/2023/texmf-dist/tex/luatex/eva-jfm
```

等。最后运行

```
1 mktexlsr
```

更新本地 T_EX 的 L_S-R 文件即可。

本文件一般情况下无需用户进行本地配置，但若有特殊需求可见第 5.3 节。

3 使用

以下是在 L^AT_EX 下使用繁中字体进行直排的示例

```
1 \usepackage{luatexja-fontspec, luatexja-adjust}
2 \setmainfont{Source Han Serif TC}[Language = Chinese Traditional, TateFeatures = {JFM = eva/{vert, trad,
   nstd}}]
3 \ltjenableadjust[priority = true]
```

(注意需要调用支持直书的文档类或使用 `\tate` 命令)。LuaT_EX-ja 的 JFM 语法为：

```
1 jfm = ⟨JFM name⟩/⟨JFM features⟩
```

而一般情况使用 `\setmainfont` 时则为：

```
1 \setmainfont{⟨font name⟩}[Language = ⟨language name⟩, ⟨dir⟩ = {JFM = ⟨JFM name⟩/⟨JFM features⟩}]
```

其中，`⟨font name⟩` 自然为需要的字体名称。`⟨language name⟩` 在使用日文字体时可忽略，而使用简中、繁中字体时为必填⁴，因 LuaT_EX-ja 会默认将其覆盖为 Japanese 选项，而这会带来灾难性的后果⁵。`⟨dir⟩` 选填 `TateFeatures`（直书）或 `YokoFeatures`（横书）。其后的 `⟨JFM name⟩` 为调用 JFM 的文件名⁶。最后的 `⟨JFM features⟩` 选项为选择使用的 JFM 特性，详细请看第 4 章。

对于进阶用户，也推荐用

```
1 \def\ltj@stdyokojfm{eva/{⟨features⟩}}
```

或配合 NFSS 来使用。

其他情况下设置 JFM 及其更多信息请看 LuaT_EX-ja 文档 [2]。

4 支持特性

本章节将介绍 Eva-JFM 的所有特性，分别为：语言特性、方向特性、扩展特性、西文特性及私有特性。

⁴ 简中填 Chinese Simplified，繁中填 Chinese Traditional 即可。

⁵ 比如错误的标点位置：日文为冒号及分号中置、其余偏靠，简中是全部偏靠，而繁中则是统统中置。

⁶ LuaT_EX-ja 会依 `jfm-⟨JFM name⟩.lua` 的格式来查找该文件。

4.1 語言特性

本区特性必填且只可填一个。不然则会报错。

`jp` → (*JaPanese*)

日语特性。当使用日文字体时需调用该特性。其与简中、繁中区别在于问号及感叹号后插入的伸缩胶量。影响特性 `lgp`，且对内部分组有影响。

`trad` → (*TRADitional chinese*)

繁体特性。当使用繁体中文字体时需调用。与简中、日语特性的区别源于中置的标点。故，对于全部标点左右插入的伸缩胶的量都与简中、日语不同。针对句点紧挨闭括号、标点位于句末时等皆有优化。

`smpl` → (*SiMPLified chinese*)

简体特性，使用简体中文字体排版时调用。与日语、繁体特性区别源于分号及冒号等全部偏靠从而影响其左右插入伸缩胶的量。Eva-JFM 对一些（不该出现的）神奇情况（如两个句号同时出现、开括号后出现问号等）进行优化。对问号、感叹号等作了特殊处理。

4.2 方向特性

本分区特性与全部其他特性兼容，可同时调用。

`vert` → (*VERTical writing*)

直书特性。对标点挤压、分组有影响。直书时必须调用。

4.3 擴展特性

本区特性 `hgp` 不依赖 `vert` 特性，其余需同 `vert` 特性同时调用。否则报错。

`extd` → (*EXTenDed font*)

压缩字体特性。默认为横比纵为 100 比 80 的字体压缩⁷。可用 `extd=<ratio>` 设置横方向拉伸比例（默认即为 1.25。需同 `extend` (`luaotfload`) 或 `FakeStretch` (`fontspec`) 同时使用。

`lgp` → (*LineGap Punctuations*)

行间标点特性。该特性将部分标点「悬挂」至行间。日文字体时与繁、简中字体时会有区别。详见第 5 章。

`hgp` → (*HanGing Punctuations*)

悬挂标点特性。该特性将部分标点「悬挂」于行末。仅简中、日文字体拥有该特性。

4.4 西文特性

本区特性使用时需先使用 `\ltjsetParameter` 设置 `jacharrange` 从而调整 `JAchar` 的范围。同时推荐与对应 `OpenType` 特性同时使用。

⁷ 日本新闻字体，如每日新闻明朝体。

hwid → (*Half WIDTH*)

半宽西文特性。使用此特性（且按上述设置完成后）西文字母排布为严格半宽。本特性不会压缩或拉伸西文字母，故当未使用对应半宽字体特性时，只会简单的重叠，此时不推荐使用。同时也会失去所有 kern 以及 italic correction 的数据，同时忽略 xkanjiskip 参数。请务必谨慎调用。

fwid → (*Full WIDTH*)

全宽西文特性。描述同上。但，若不调用全宽特性，西文间距会被简单撑开。

4.5 私有特性

使用本区特性前请先确保你清楚地知道你在做什么。

nstd → (*Non Standard*)

忽略标准特性。字体排印标准 [1] 认为逗号的压缩权重应比句号要低。本特性将句号的压缩优先级与逗号交换，使逗号被优先压缩^{*8}。仅在使用 luatexja-adjust 宏集时有效。

5 行間標點特性

本章节将提供更多详细的关于行间标点特性的信息，以及可能出现的问题及其解决方案。

5.1 關於「懸掛」

行间标点可见于古籍之中，是将标点符号与直书结合妥协的产物。

传统上悬挂句号与逗号。而 Eva-JFM 特性在繁中、简中特性下会悬挂句号、逗号、顿号、冒号及分号，日文字体下则不悬挂冒号及分号。原因在于日本习惯上将冒号与分号看作「中点类」，直书时横置处理。

本 JFM 将全部标点悬挂于字体右下位置。详见下一节。

5.2 懸掛的位置

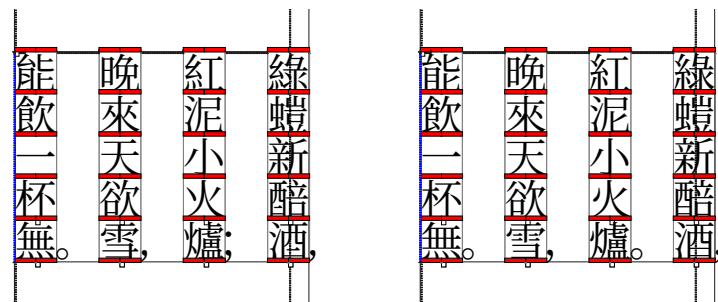


图 1 行间标点特性示意图

标点悬挂的位置有以下考量，可参照图 1。若有特殊需求请看第 5.3 节。优先级由上至下。

^{*8} 考虑逗号、句号在文字系统中占的重量，以及「开明」压缩风格。

- 三种字体风格统一，位置原则上一致（故，繁体字体也悬挂于右下、而非居中）；
- 不同标点中的相同（似）元素位置相同；
- 繁体、简体、日文字体标点触字框右边线；
- 不同标点符号因形状不同可于字框底线略下沉或上浮；
- 不同标点符号因大小不同可靠近或远离字框右边线；
- 三种字体可分别因字符设计的差异而位置略微区别。

5.3 用戶配置

本特性是以三套思源字体为基准设计的。而由于各字体的标点符号位置不可避免会有不同，故在某些特殊情况下会出现错位影响视觉效果的情况。或是单纯对原设定而言更偏好其他设定等原因，本节提供自定义及调整的两种方法。第一种较简单但可移植性较差，而第二种虽繁琐但一劳永逸。

5.3.1 修改原程式碼

在 Eva-JFM 中，控制行间标点的分区域别为

```
1 [101,2] ==> [1]; [201,2] ==> [2]; [301,2] ==> [3].
```

只需调整其中 `left` 和 `down` 键的值即可。其中 `left` 为向右移动，`down` 为向下移动。具体可参照终章。

5.3.2 使用外掛符號字體

该方法的原理就是使用特殊的仅包含（标点）符号的字体来替换原有字体中的标点符号，从而稳定其表现。符号字体可使用 <https://github.com/Buernia/Zhudou-Sans> 处的开源字体。其基于思源黑体，还添加了许多其他符号（但这里我们只会用到六个符号）及标点挤压等特性。

将其放入 TEXMF 并更新 Ls-R 文件后即可使用 Lua_T_EX-j_a 提供的 `AltFont` 键进行替换，例元：

```
1 \setmainfont[
2   Language = <language>,
3   TateFeatures = {
4     JFM = eva/{vert, lgp, <language>},
5     AltFont = {
6       {Range = "<utf-8 code>", Font = <symbol font>}
7     }
8   }
9 ]{<main font>}
```

其中首个 `<language>` 可选填 `Japanese`、`Chinese Traditional` 或 `Chinese Simplified`，第二个则填语言特性分区的对应 `jp`、`trad` 及 `smp1` 特性。`<utf-8 code>` 则为需要替换的标点符号的 Unicode 编码，如需替换句号（ideographic full stop, U+3002）则填 `3002`⁹。`<symbol font>` 以及 `<main font>` 填符号字体名称、正文字体名称即可。

对于开发者，也建议使用 NFSS 的

⁹ 编码可至 <https://www.unicode.org/charts/unihanrsindex.html> 查询。

```

1 \DeclareAlternateKanjiFont{<base encoding>}{<base family>}{<base series>}{<base shape>}{<alt encoding>}{<alt
family>}{<alt series>}{<alt shape>}{range}

```

进行替换。其中 $\langle base \rangle$ 为正文字体， $\langle alt \rangle$ 则为替换符号字体。

具体语法及示例可看 LuaTeX-ja 文档 [2]。

6 后發

Eva-JFM 的内部分组受 min10.tfm [5] 的启发，支持的 `priority` 特性则取自阿部紀行氏的 `jlreq.lua` [6] 文件。其余可见参考文献。

参考文献

- [1] W3C Japanese Layout Task Force (ed). Requirements for Japanese Text Layout (W3C Working Group Note), 2022, 2023. <https://www.w3.org/TR/jlreq/>.
- [2] LuaTeX-ja プロジェクトチーム. LuaTeX-ja パッケージ, 2022, 2023.
- [3] The Unicode Consortium. The Unicode Standard Version 15.0 - Core Specification, 2022.
- [4] Victor Eijkhout. T_EX by Topic, A T_EXnician's Reference, Addison-Wesley, 1992.
- [5] 乙部巖己. min10 フォントについて. <http://argent.shinshu-u.ac.jp/~otobe/tex/files/min10.pdf>.
- [6] Noriyuki Abe. Jlreq Document Class, 2022. <https://github.com/abenori/jlreq>.

程式碼

以下为 `jfm-eva.lua` 文件内容，供参考。

```

1 ---- Evangelion Japanese Font Metric for LuaTeX
2 ---- Current Version: 1.0.1 (a)
3 ---- Dev URL: https://github.com/RadioNoiseE/Evangelion-JFM
4 ---- © Copyright 2023, RadioNoiseE
5
6
7 -- 初始化
8 local lang_jp, lang_tc, lang_sc, dir_vt, font_extd, punc_lg, punc_hg, std_nil, al_hw, al_fw
9
10 if luatexja.jfont.jfm_feature then
11     lang_jp = luatexja.jfont.jfm_feature.jp
12     lang_tc = luatexja.jfont.jfm_feature.trad
13     lang_sc = luatexja.jfont.jfm_feature.smpl
14     dir_vt = luatexja.jfont.jfm_feature.vert
15     font_extd = luatexja.jfont.jfm_feature.extd
16     punc_lg = luatexja.jfont.jfm_feature.lgp
17     punc_hg = luatexja.jfont.jfm_feature.hgp
18     std_nil = luatexja.jfont.jfm_feature.nstd
19     al_hw = luatexja.jfont.jfm_feature.hwid

```

```

20     al_fw = luatexja.jfont.jfm_feature.fwid
21 end
22
23 -- 預處理及容錯
24 if font_extd = true and dir_vt = false then
25     tex.error('JFM feature "extd" only works with feature "vert".\n' ..
26         'For now I\'ll ignore it.')
27 end
28
29 if punc_lg = true and dir_vt = false then
30     tex.error('JFM feature "lgp" only works with feature "vert".\n' ..
31         'For now I\'ll ignore it.')
32 end
33
34 if al_hw = true and al_fw = true then
35     tex.error('JFM feature "hwid" cannot be used with "fwid".')
36 end
37
38 if not ((lang_jp and not (lang_tc or lang_sc)) or
39     (lang_tc and not (lang_jp or lang_sc)) or
40     (lang_sc and not (lang_jp or lang_tc))) then
41     tex.error('Specify one and only one feature from three language specific features\n' ..
42         '"jp", "trad" or "smp1"\n' ..
43         'is required.\n' ..
44         'For now I\'ll use "lang_jp" for japanese by default.')
45 end
46
47 -- 壓縮比例設定
48 if font_extd = true then
49     local extd_ratio = (type(luatexja.jfont.jfm_feature and font_extd) == 'string') and tonumber(font_extd)
50     or 1.25
51 end
52
53 -- 定義函數宏
54 local function logic_anif(f1, f2, r1, r2)
55     local rta = f1 and (f2 and r1) or r2
56     return rta
57 end
58
59 local function logic_if(f1, r1, r2)
60     local rti = f1 and r1 or r2
61     return rti
62 end
63
64 local function context_height()
65     local rth = dir_vt and (font_extd and extd_ratio/2 or 0.5) or 0.88
66     return rth
67 end

```

```

68 local function context_depth()
69     local rtd = dir_vt and (font_extd and extd_ratio/2 or 0.5) or 0.12
70     return rtd
71 end
72
73 -- 主體
74 local eva = {
75     version = 3,
76     dir = logic_if(dir_vt, 'tate', 'yoko'),
77     zw = 1,
78     zh = logic_anif(dir_vt, font_extd, 1.25, 1),
79     kanjiskip = {0, 0.25, 0},
80     xkanjiskip = {0.25, 0.125, 0.125},
81
82     [0] = { -- 缺省類
83         width = 1,
84         height = context_height(),
85         depth = context_depth(),
86         italic = 0,
87         left = 0,
88         down = 0,
89         align = 'middle',
90         glue = {
91             [1] = logic_if(lang_tc, {0.25, 0, 0.125, ratio = 1, priority = logic_if(std_nil, {-1, 0}, {-1,
92                 -2})}, {priority = logic_if(std_nil, {-1, 0}, {-1, -2})}),
93             [2] = logic_if(lang_tc, {0.25, 0, 0.125, ratio = 1, priority = logic_if(std_nil, {-1, -2}, {-1,
94                 0})}, {priority = logic_if(std_nil, {-1, -2}, {-1, 0})}),
95             [3] = logic_if(dir_vt, {priority = {0, -1}}, logic_if(lang_tc, {0.25, 0, 0.125, ratio = 1,
96                 priority = {-1, -1}}, {priority = {-1, -1}})),
97             [7] = {0.5, 0, 0.25, ratio = 1, priority = {-1, -2}},
98             [9] = {0.25, 0, 0.125, ratio = 1, priority = {-1, -1}}
99         },
100         round_threshold = 0.01
101     },
102
103     [1] = { -- 読点類
104         chars = logic_anif(dir_vt, punc_lg, {}, {'\ ', ' ', '}),
105         width = 0.5,
106         height = context_height(),
107         depth = context_depth(),
108         italic = 0,
109         left = 0,
110         down = 0,
111         align = logic_if(lang_tc, 'middle', 'left'),
112         glue = {
113             [0] = logic_if(lang_tc, {0.25, 0, 0.125, ratio = 0, priority = logic_if(std_nil, {-1, 0}, {-1,
114                 -2})}, {0.5, 0, 0.25, ratio = 0, priority = logic_if(std_nil, {-1, 0}, {-1, -2})}),
115             [1] = logic_if(lang_tc, {0.5, 0, 0.25}, {0.5, 0, 0.25}),
116             [2] = logic_if(lang_tc, {0.5, 0, 0.25}, {0.5, 0, 0.25}),

```



```

113     [3] = logic_if(dir_vt, logic_if(lang_tc, {0.25, 0, 0.125, ratio = 0, priority = {0, -1}}, {0.5,
        0, 0.25, priority = {0, -1}}), logic_if(lang_tc, {0.5, 0, 0.25, priority = {0, -1}}, {0.5,
        0, 0.25, ratio = 0, priority = {0, -1}})),
114     [4] = logic_if(lang_tc, {0.25, 0, 0.125, ratio = 0, priority = logic_if(std_nil, {0, 0}, {0,
        -2}}), {0.5, 0, 0.25, ratio = 0, priority = logic_if(std_nil, {0, 0}, {0, -2}})),
115     [5] = logic_if(lang_tc, {0.25, 0, 0.125, ratio = 0, priority = logic_if(std_nil, {0, 0}, {0,
        -2}}), {0.5, 0, 0.25, ratio = 0, priority = logic_if(std_nil, {0, 0}, {0, -2}})),
116     [6] = logic_if(lang_tc, {0.25, 0, 0.125, ratio = 0, priority = logic_if(std_nil, {-1, 0}, {-1,
        -2}}), {0.5, 0, 0.25, ratio = 0, priority = logic_if(std_nil, {-1, 0}, {-1, -2}})),
117     [7] = logic_if(lang_tc, {0.25, 0, 0.125, ratio = 0, priority = logic_if(std_nil, {-1, 0}, {-1,
        -2}}), {0.5, 0, 0.25, ratio = 0, priority = logic_if(std_nil, {-1, 0}, {-1, -2}})),
118     [8] = logic_if(lang_tc, {0.25, 0, 0.125}, {}),
119     [9] = logic_if(lang_tc, {0.5, 0, 0.25, priority = {0, -1}}, {0.75, 0, 0.25, ratio = 1/3,
        priority = {0, -1}})
120 },
121 end_adjust = logic_if(lang_tc, {0.25, 0}, logic_if(punc_hg, {-0.5, 0}, {0, 0}))
122 },
123
124 [101] = { -- 読点類 (行間a)
125     chars = logic_anif(dir_vt, punc_lg, {'\'}, {}),
126     width = 0,
127     height = context_height(),
128     depth = context_depth(),
129     italic = 0,
130     left = 0.38,
131     down = -0.34,
132     align = 'left'
133 },
134
135 [102] = { -- 読点類 (行間b)
136     chars = logic_anif(dir_vt, punc_lg, {'\'}, {}),
137     width = 0,
138     height = context_height(),
139     depth = context_depth(),
140     italic = 0,
141     left = logic_if(lang_tc, 0.62, 0.40),
142     down = logic_if(lang_tc, -0.58, -0.26),
143     align = 'left'
144 },
145
146 [2] = { -- 句点類
147     chars = logic_anif(dir_vt, punc_lg, {}, {'.', '。'}),
148     width = 0.5,
149     height = context_height(),
150     depth = context_depth(),
151     italic = 0,
152     left = 0,
153     down = 0,
154     align = logic_if(lang_tc, 'middle', 'left'),

```

```

155 glue = {
156     [0] = logic_if(lang_tc, {0.25, 0, 0.125, ratio = 0, priority = logic_if(std_nil, {-1, -2}, {-1,
157         0}}), {0.5, 0, 0.25, ratio = 0, priority = logic_if(std_nil, {-1, -2}, {-1, 0}})),
158     [1] = logic_if(lang_tc, {0.5, 0, 0.25}, {0.5, 0, 0.25, ratio = 0}),
159     [2] = logic_if(lang_tc, {0.5, 0, 0.25}, {0.5, 0, 0.25, ratio = 0}),
160     [3] = logic_if(dir_vt, logic_if(lang_tc, {0.25, 0, 0.125, ratio = 0, priority = {0, -1}}, {0.5,
161         0, 0.25, priority = {0, -1}}), logic_if(lang_tc, {0.5, 0, 0.25, priority = {0, -1}}, {0.5,
162         0, 0.25, ratio = 0, priority = {0, -1}})),
163     [4] = logic_if(lang_tc, {0.25, 0, 0.125, ratio = 0, priority = logic_if(std_nil, {0, -2}, {0,
164         0}}), {0.5, 0, 0.25, ratio = 0, priority = logic_if(std_nil, {0, -2}, {0, 0}})),
165     [5] = logic_if(lang_tc, {0.25, 0, 0.125, ratio = 0, priority = logic_if(std_nil, {0, -2}, {0,
166         0}}), {0.5, 0, 0.25, ratio = 0, priority = logic_if(std_nil, {0, -2}, {0, 0}})),
167     [6] = logic_if(lang_tc, {0.25, 0, 0.125, ratio = 0, priority = logic_if(std_nil, {-1, -2}, {-1,
168         0}}), {0.5, 0, 0.25, ratio = 0, priority = logic_if(std_nil, {-1, -2}, {-1, 0}})),
169     [7] = logic_if(lang_tc, {0.25, 0, 0.125, ratio = 0, priority = logic_if(std_nil, {-1, -2}, {-1,
170         0}}), {0.5, 0, 0.25, ratio = 0, priority = logic_if(std_nil, {-1, -2}, {-1, 0}})),
171     [8] = logic_if(lang_tc, {0.25, 0, 0.125, ratio = 0}, {}),
172     [9] = logic_if(lang_tc, {0.5, 0, 0.25, priority = {0, -1}}, {0.75, 0, 0.25, ratio = 1/3,
173         priority = {0, -1}})
174 },
175 end_adjust = logic_if(lang_tc, {0.25, 0}, logic_if(punc_hg, {-0.5, 0}, {0, 0}))
176 },
177
178 [201] = { -- 句點類 (行間a)
179     chars = logic_anif(dir_vt, punc_lg, {'.'}, {}),
180     width = 0,
181     height = context_height(),
182     depth = context_height(),
183     italic = 0,
184     left = logic_if(lang_tc, 0.68, 0.34),
185     down = logic_if(lang_tc, -0.58, -0.28),
186     align = 'left'
187 },
188
189 [202] = { -- 句點類 (行間b)
190     chars = logic_anif(dir_vt, punc_lg, {'.'}, {}),
191     width = 0,
192     height = context_height(),
193     depth = context_height(),
194     italic = 0,
195     left = 0.42,
196     down = -0.35,
197     align = 'left'
198 },
199
200 [3] = { -- 兩點類
201     chars = logic_if(lang_jp, {}, (logic_anif(dir_vt, punc_lg, {}, {':', ';'}))),
202     width = logic_if(dir_vt, 1, 0.5),
203     height = context_height(),

```

```

196     depth = context_depth(),
197     italic = 0,
198     left = 0,
199     down = 0,
200     align = logic_if(lang_tc, 'middle', 'left'),
201     glue = {
202         [0] = logic_if(dir_vt, {priority = {-1, -1}}, logic_if(lang_tc, {0.25, 0, 0.125, ratio = 0,
203             priority = {-1, -1}}, {0.5, 0, 0.25, ratio = 0, priority = {-1, -1}})),
204         [1] = logic_if(dir_vt, logic_if(lang_tc, {0.25, 0, 0.125, ratio = 1, priority = {0, -1}}, {
205             priority = {0, -1}}, logic_if(lang_tc, {0.5, 0, 0.25, priority = {0, -1}}, {0.5, 0, 0.25,
206             ratio = 0, priority = {0, -1}})),
207         [2] = logic_if(dir_vt, logic_if(lang_tc, {0.25, 0, 0.125, ratio = 1, priority = {0, -1}}, {
208             priority = {0, -1}}, logic_if(lang_tc, {0.5, 0, 0.25, priority = {0, -1}}, {0.5, 0, 0.25,
209             ratio = 0, priority = {0, -1}})),
210         [3] = logic_if(dir_vt, {priority = {0, -1}}, logic_if(lang_tc, {0.5, 0, 0.25, priority = {0,
211             -1}}, {0.5, 0, 0.25, ratio = 0, priority = {0, -1}})),
212         [4] = logic_if(dir_vt, {priority = {0, -1}}, logic_if(lang_tc, {0.25, 0, 0.125, ratio = 0,
213             priority = {0, -1}}, {0.5, 0, 0.25, ratio = 0, priority = {0, -1}})),
214         [5] = logic_if(dir_vt, {priority = {0, -1}}, logic_if(lang_tc, {0.25, 0, 0.125, ratio = 0,
215             priority = {0, -1}}, {0.5, 0, 0.25, ratio = 0, priority = {0, -1}})),
216         [6] = logic_if(dir_vt, {priority = {-1, -1}}, logic_if(lang_tc, {0.25, 0, 0.125, ratio = 0,
217             priority = {-1, -1}}, {0.5, 0, 0.25, ratio = 0, priority = {-1, -1}})),
218         [7] = logic_if(dir_vt, {priority = {-1, -1}}, logic_if(lang_tc, {0.25, 0, 0.125, ratio = 0,
219             priority = {-1, -1}}, {0.5, 0, 0.25, ratio = 0, priority = {-1, -1}})),
220         [8] = logic_if(lang_tc, {0.25, 0, 0.125, ratio = 0, priority = {0, -1}}, {0.5, 0, 0.25, ratio =
221             0, priority = {0, -1}}),
222         [9] = logic_if(lang_tc, {0.5, 0, 0.25, priority = {0, -1}}, {0.75, 0, 0.25, ratio = 1/3,
223             priority = {0, -1}})
224     },
225 },
226
227 [301] = { -- 兩點類 (行間a)
228     chars = logic_if(lang_jp, {}, logic_anif(dir_vt, punc_lg, {'.'}, {})),
229     width = 0,
230     height = context_height(),
231     depth = context_depth(),
232     italic = 0,
233     left = logic_if(lang_tc, 0.94, 0.72),
234     down = logic_if(lang_tc, -0.58, -0.34),
235     align = 'left'
236 },
237
238 [302] = { -- 兩點類 (行間b)
239     chars = logic_if(lang_jp, {}, logic_anif(dir_vt, punc_lg, {';'}, {})),
240     width = 0,
241     height = context_height(),
242     depth = context_depth(),
243     italic = 0,
244     left = logic_if(lang_tc, 0.96, 0.78),

```

```

233     down = logic_if(lang_tc, -0.58, -0.34),
234     align = 'left'
235 },
236
237 [4] = { -- 小書きの仮名類
238     chars = {
239         'あ', 'い', 'う', 'え', 'お', 'っ', 'ゃ', 'ゅ', 'ょ', 'わ', 'か',
240         'け', 'く', 'ぐ', 'ア', 'イ', 'ウ', 'エ', 'オ', 'ッ', 'ヤ', 'ユ',
241         'ヨ', 'ワ', 'カ', 'ケ', 'ハ', 'ベ', 'ク', 'シ', 'ス', 'ト', 'ヌ',
242         'ハ', 'ヒ', 'フ', 'ヘ', 'ホ', 'ム', 'ラ', 'リ', 'ル', 'レ', 'ロ'
243     },
244     width = 1,
245     height = context_height(),
246     depth = context_depth(),
247     italic = 0,
248     left = 0,
249     down = 0,
250     align = 'middle',
251     glue = {
252         [1] = logic_if(lang_tc, {0.25, 0, 0.125, ratio = 1}, {}),
253         [2] = logic_if(lang_tc, {0.25, 0, 0.125, ratio = 1}, {}),
254         [3] = logic_if(lang_tc, {0.25, 0, 0.125, ratio = 1, priority = {0, -1}}, {priority = {0, -1}}),
255         [7] = {0.5, 0, 0.25, ratio = 1, priority = {-1, -2}},
256         [9] = {0.25, 0, 0.125, ratio = 1, priority = {0, -1}}
257     }
258 },
259
260 [5] = { -- 疑問感嘆類
261     chars = {'!', '?', '!!', 'ㄱ', '!', 'ㄴ', '!', '??'},
262     width = logic_if(dir_vt, 1, logic_if(lang_sc, 0.5, 1)),
263     height = context_height(),
264     depth = context_depth(),
265     italic = 0,
266     left = 0,
267     down = 0,
268     align = logic_if(dir_vt, 'middle', logic_if(lang_sc, 'left', 'middle')),
269     glue = {
270         [0] = logic_if(dir_vt, logic_if(lang_jp, {1, 0, 0.5, ratio = 0, priority = {-1, 0}}, {priority =
                {-1, 0}}), logic_if(lang_tc, {priority = {-1, 0}}, {0.5, 0, 0.25, ratio = 0, priority =
                {-1, 0}})),
271         [1] = logic_if(lang_tc, {0.25, 0, 0.125, ratio = 1}, logic_anif(not dir_vt, lang_sc, {0.5, 0,
                0.25, ratio = 0}, {})),
272         [2] = logic_if(lang_tc, {0.25, 0, 0.125, ratio = 1}, logic_anif(not dir_vt, lang_sc, {0.5, 0,
                0.25, ratio = 0}, {})),
273         [3] = logic_if(dir_vt, {priority = {-1, -1}}, logic_if(lang_tc, {0.25, 0, 0.125, ratio = 1,
                priority = {-1, -1}}, {0.75, 0, 0.25, ratio = 1/3, priority = {-1, -1}})),
274         [4] = logic_if(dir_vt, logic_if(lang_jp, {1, 0, 0.5, ratio = 0}, {}), logic_if(lang_tc, {,
                0.5, 0, 0.25, ratio = 0})),
275         [7] = {0.5, 0, 0.25, ratio = 1, priority = {-1, -2}},

```



```

322     width = 0.5,
323     height = context_height(),
324     depth = context_depth(),
325     italic = 0,
326     left = 0,
327     down = 0,
328     align = 'left',
329     glue = {
330         [0] = {0.5, 0, 0.25, ratio = 0, priority = {-1, -2}},
331         [1] = logic_if(lang_tc, {0.25, 0, 0.125, ratio = 1}, {}),
332         [2] = logic_if(lang_tc, {0.25, 0, 0.125, ratio = 1}, {}),
333         [3] = logic_if(lang_sc, {priority = {0, -1}}, {0.25, 0, 0.125, ratio = 1, priority = {0, -1}}),
334         [4] = {0.5, 0, 0.25, ratio = 0, priority = {0, -2}},
335         [5] = {0.5, 0, 0.25, ratio = 0, priority = {0, -2}},
336         [6] = {0.5, 0, 0.25, ratio = 0, priority = {-1, -2}},
337         [7] = {0.5, 0, 0.25, ratio = 0, priority = {-1, -2}, kanjiskip_stretch = 1},
338         [8] = {0, 0, 0, priority = {0, 0}, kanjiskip_shrink = 1},
339         [9] = {0.25, 0, 0.125, ratio = 1, priority = {0, -1}}
340     }
341 },
342
343 [9] = { -- 中點類
344     chars = logic_if(lang_jp, {'・', '・', '・'}, {'・', '・'}),
345     width = 0.5,
346     height = context_height(),
347     depth = context_depth(),
348     italic = 0,
349     left = 0,
350     down = 0,
351     align = 'middle',
352     glue = {
353         [0] = {0.25, 0, 0.125, ratio = 0, priority = {-1, -1}},
354         [1] = logic_if(lang_tc, {0.5, 0, 0.25, priority = {0, -1}}, {0.25, 0, 0.125, ratio = 0, priority
= {0, -1}}),
355         [2] = logic_if(lang_tc, {0.5, 0, 0.25, priority = {0, -1}}, {0.25, 0, 0.125, ratio = 0, priority
= {0, -1}}),
356         [3] = logic_if(dir_vt, {0.25, 0, 0.125, ratio = 0, priority = {0, -1}}, logic_if(lang_tc, {0.5,
0, 0.25, priority = {0, -1}}, {0.25, 0, 0.125, priority = {0, -1}})),
357         [4] = {0.25, 0, 0.125, ratio = 0, priority = {0, -1}},
358         [5] = {0.25, 0, 0.125, ratio = 0, priority = {0, -1}},
359         [6] = {0.25, 0, 0.125, ratio = 0, priority = {-1, -1}},
360         [7] = {0.25, 0, 0.125, ratio = 0, priority = {-1, -1}},
361         [8] = {0.25, 0, 0.125, ratio = 0, priority = {0, -1}},
362         [9] = {0.5, 0, 0.25, priority = {0, -1}}
363     }
364 },
365
366 [10] = { -- 西文
367     chars = {},

```

```

368     width = 0,
369     height = context_height(),
370     depth = context_depth(),
371     italic = 0,
372     left = 0,
373     down = 0,
374     align = 'middle',
375     glue = {}
376 }
377 }
378
379 if al_hw == true or al_fw == true then
380     eva[10].chars = {'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k', 'l', 'm',
381                     'n', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z',
382                     'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M',
383                     'N', 'O', 'P', 'Q', 'R', 'S', 'T', 'U', 'V', 'W', 'X', 'Y', 'Z'}
384     eva[10].glue = table.fastcopy(eva[0].glue)
385     eva[10].glue[0] = {0.25, 0.125, 0.125, ratio = 0, priority = {0, -1}}
386     eva[0].glue[10] = {0.25, 0.125, 0.125, ratio = 1, priority = {0, -1}}
387     for _, catnum in ipairs({1, 2, 3, 5, 8, 9}) do
388         eva[catnum].glue[10] = table.fastcopy(eva[catnum].glue[0])
389     end
390 end
391
392 if al_hw == true and al_fw == false then
393     eva[10].width = 0.5
394 end
395
396 if al_fw == false and al_fw == true then
397     eva[10].width = 1
398 end
399
400 luatexja.jfont.define_jfm(eva)

```