

年文檔類 Nian Class

黃 京

西曆 2023 年 5 月 13 日

概要

为在 LuaTeX 下排印中日文本而作的文档类。基于 expl3 构建。

1 初始化

1.1 載入 \LaTeX 3 並檢驗依賴

```
1 \NeedsTeXFormat{LaTeX2e}
2 \RequirePackage{expl3}
3 \ProvidesExplClass{niancls}{2023-04-15}{1.0.0}{Nian Document Class}
```

定义载入文档类信息等。

```
4 %<@@=ncls>
5 \prop_gput:Nnn \g_msg_module_name_prop { ncls } { niancls }
```

申明结束。接下来检查依赖，首先为 xparse、l3keys2e 及 etoolbox 宏包。

```
6 \cs_if_exist:NF \NewDocumentCommand
7 { \RequirePackage { xparse } }
8 \cs_if_exist:NTF \ProcessKeyOptions
9 { \cs_new:Nn \@@_keyoptions_process:n { \ProcessKeyOptions { #1 } } }
10 {
11 \RequirePackage { l3keys2e }
12 \cs_new:Nn \@@_keyoptions_process:n { \ProcessKeysOptions { #1 } }
13 }
14 \cs_if_exist:NF \AtEndPreamble
15 { \RequirePackage { etoolbox } }
```

接下来检查 expl3 的版本。

```
16 \ifpackagelater { expl3 } { 2021-02-10 } { }
17 {
18 \msg_new:nnnn { ncls } { latex3-too-old }
19 { Package~`l3kernel'~and~`l3packages'~too~old. }
20 {
21 You~need~to~update~your~installation~of~the~bundles~
22 `l3kernel'~and~`l3packages'. \\\
23 Loading~niancls~will~abort!
24 }
25 \msg_critical:nn { ncls } { latex3-too-old }
26 }
```

后进行 \LaTeX 2_ϵ 格式之版本检查。

```
27 \ifl@t@r \fmtversion { 2021-06-01 } { }  
28 {  
29 \msg_new:nnnn { ncls } { latex-too-old }  
30 { Format~LaTeX2e~version~too~old. }  
31 {  
32 You~need~to~update~your~LaTeX2e~to~the~latest~release. \\  
33 Loading~nianscls~will~abort!  
34 }  
35 \msg_critical:nn { ncls } { latex-too-old }  
36 }
```

最后检查 \LaTeX 引擎，仅支持使用 \LuaTeX 编译。

```
37 \sys_if_engine luatex:F  
38 {  
39 \msg_new:nnnn { ncls } { unsupported-engine }  
40 { LuaTeX~is~the~only~supported~engine~for~nianscls. }  
41 {  
42 You~should~switch~to~LuaTeX~to~use~nianscls. \\  
43 Loading~nianscls~will~abort!  
44 }  
45 \msg_fatal:nn { ncls } { unsupported-engine }  
46 }
```

1.2 私有定义

定义 $\backslash\text{AtEndPreamble}$ 钩子。

```
47 \cs_new_protected:Npn \@@_preamble_end:n { \AtEndPreamble }
```

「疑」定义 $\backslash\text{AtEndOfClass}$ 钩子。

```
48 % \cs_new_protected:Npn \@@_at_end:n { \AtEndOfClass }
```

定义 $\backslash\text{AtBeginDocument}$ 钩子。

```
49 \cs_new_protected:Npn \@@_doc_beg:n { \AtBeginDocument }
```

定义用于在读取结束后释放缓存的宏。

```
50 \seq_new:N \g_@@_aftercls_del_seq  
51 \cs_set:Nn \@@_aftercls_addtodel:N  
52 { \seq_gput_right:Nn \g_@@_aftercls_del_seq { #1 } }  
53 \@@_preamble_end:n  
54 {  
55 \ExplSyntaxOn  
56 \cs_undefine:N \g_@@_aftercls_del_seq  
57 \ExplSyntaxOff  
58 }
```

封装 \LuaTeX 提供的 Lua 接口。

```
59 \cs_new:Npn \@@_luafunc_new:N { \newluafunction }  
60 \cs_new:Npn \@@_luafunc_use:N { \luafunction }
```

提供键对值的统一错误调试处理模版。

```

61 \msg_new:nnnn { ncls } { unknown-choice }
62 { Unknown~choice~given~to~key~`#1' }
63 {
64     Valid~choices~are:~#2; \\
65     while~you~gave:~#3.
66 }

```

1.3 内存清理

在导言区末尾清除所有非必要宏。

```

67 \@@_aftercls_addtodel:N \@@_keyoptions_process:n
68 \@@_aftercls_addtodel:N \@@_preamble_end:n
69 % \@@_aftercls_addtodel:N \@@_at_end:n
70 \@@_aftercls_addtodel:N \@@_at_doc_begin:n
71 \@@_aftercls_addtodel:N \@@_aftercls_addtodel:N
72 \@@_aftercls_addtodel:N \@@_luafunc_new:N
73 \@@_aftercls_addtodel:N \@@_luafunc_use:N

```

2 鍵對直之「預處理」

2.1 紙張配置用

2.1.1 尺寸信息

处理纸张尺寸信息。

```

74 \tl_new:N \g_@@_paper_sizeinfo_tl
75 \keys_define:nn { ncls }
76 {
77     paper .tl_gset:N = \g_@@_paper_sizeinfo_tl,
78     paper .value_required:n = true,
79     peper .initial:n = { a4 }
80 }

```

2.1.2 頁面方向

設置页面方向。

```

81 \bool_new:N \g_@@_paper_portrait_bool
82 \keys_define:nn { ncls }
83 {
84     orientation .choice:,
85     orientation / portrait .code:n = { \bool_gset_true:N \g_@@_paper_portrait_bool },
86     orientation / landscape .code:n = { \bool_gset_false:N \g_@@_paper_portrait_bool },
87     orientation / unknown .code:n =
88     {
89         \msg_error:nnxxx { ncls } { unknown-choice }
90         { orientation }
91         { portrait,~landscape }
92         { \exp_not:n { #1 } }

```

```

93     },
94     orientation .value_required:n = true,
95     orientation .initial:n = { portrait }
96 }

```

2.2 組版方向

确定使用竖书或是横排。

```

97 \bool_new:N \g_@@_dir_tate_bool
98 \keys_define:nn { ncls }
99 {
100     direction .choice:,
101     direction / yoko .code:n = { \bool_gset_false:N \g_@@_dir_tate_bool },
102     direction / tate .code:n = { \bool_gset_true:N \g_@@_dir_tate_bool },
103     direction / unknown .code:n =
104     {
105         \msg_new:nnxxx { ncls } { unknown-choice }
106         { direction }
107         { yoko,~tate }
108         { \exp_not:n { #1 } }
109     },
110     direction .value_required:n = true,
111     direction .initial:n = { yoko }
112 }

```

2.3 字體相關配置用

2.3.1 全局基準字體參數設定

处理用户所需的全局基准字体设置。

```

113 \tl_new:N \g_@@_font_magscale_tl
114 \keys_define:nn { ncls }
115 {
116     fontsize .choice:,
117     fontsize / 7pt .code:n = { \tl_gset:Nn \g_@@_font_magscale_tl { 0.6940 } },
118     fontsize / 8pt .code:n = { \tl_gset:Nn \g_@@_font_magscale_tl { 0.8330 } },
119     fontsize / 9pt .code:n = { \tl_gset:Nn \g_@@_font_magscale_tl { 0.9130 } },
120     fontsize / 10pt .code:n = { \tl_gset:Nn \g_@@_font_magscale_tl { 1.0000 } },
121     fontsize / 11pt .code:n = { \tl_gset:Nn \g_@@_font_magscale_tl { 1.0953 } },
122     fontsize / 12pt .code:n = { \tl_gset:Nn \g_@@_font_magscale_tl { 1.2000 } },
123     fontsize / 13pt .code:n = { \tl_gset:Nn \g_@@_font_magscale_tl { 1.3000 } },
124     fontsize / 14pt .code:n = { \tl_gset:Nn \g_@@_font_magscale_tl { 1.4400 } },
125     fontsize / 15pt .code:n = { \tl_gset:Nn \g_@@_font_magscale_tl { 1.5000 } },
126     fontsize / 16pt .code:n = { \tl_gset:Nn \g_@@_font_magscale_tl { 1.6000 } },
127     fontsize / 17pt .code:n = { \tl_gset:Nn \g_@@_font_magscale_tl { 1.7280 } },
128     fontsize / 20pt .code:n = { \tl_gset:Nn \g_@@_font_magscale_tl { 2.0000 } },
129     fontsize / 21pt .code:n = { \tl_gset:Nn \g_@@_font_magscale_tl { 2.0740 } },
130     fontsize / 24pt .code:n = { \tl_gset:Nn \g_@@_font_magscale_tl { 2.4000 } },

```

```

131     fontsize / 25pt .code:n = { \tl_gset:Nn \g_@@_font_magscale_tl { 2.4880 } },
132     fontsize / 30pt .code:n = { \tl_gset:Nn \g_@@_font_magscale_tl { 2.9860 } },
133     fontsize / 36pt .code:n = { \tl_gset:Nn \g_@@_font_magscale_tl { 3.5830 } },
134     fontsize / 43pt .code:n = { \tl_gset:Nn \g_@@_font_magscale_tl { 4.3000 } },
135     fontsize / unknown .code:n =
136     {
137         \msg_error:nnxxx { ncls } { unknown-choice }
138         { fontsize }
139         {
140             7pt,~ 8pt,~ 9pt,~ 10pt,~ 11pt,~ 12pt,~ 13pt,~ 14pt,~ 15pt,~
141             17pt,~ 20pt,~ 21pt,~ 24pt,~ 25pt,~ 30pt,~ 36pt,~ 43pt
142         }
143         { \exp_not:n { #1 } }
144     },
145     fontsize .value_required:n = true,
146     fontsize .initial:n = { 10pt }
147 }

```

2.3.2 語言設定

設置文档类之全局语言。

```

148 \str_new:N \g_@@_lang_cj_str
149 \keys_define:nn { ncls }
150 {
151     language .choice:,
152     language / trad .code:n = { \str_gset:Nn \g_@@_lang_cj_str { t } },
153     language / smpl .code:n = { \str_gset:Nn \g_@@_lang_cj_str { s } },
154     language / jp .code:n = { \str_gset:Nn \g_@@_lang_cj_str { j } },
155     language / unknown .code:n =
156     {
157         \msg_error:nnxxx { ncls } { unknown-choice }
158         { language }
159         { trad,~smpl,~jp }
160         { \exp_not:n { #1 } }
161     },
162     language .value_required:n = true,
163     language .initial:n = { jp }
164 }

```

2.3.3 字體設定

設定全局明朝及哥特字体。

```

165 \tl_new:N \g_@@_font_mincho_tl
166 \tl_new:N \g_@@_font_gothic_tl
167 \keys_define:nn { ncls }
168 {
169     mincho .tl_gset:N = \g_@@_font_mincho_tl,
170     gothic .tl_gset:N = \g_@@_font_gothic_tl,
171     mincho .value_required:n = true,

```

```

172     gothic .value_required:n = true,
173     mincho .initial:n = { HaranoAji Mincho },
174     gothic .initial:n = { HaranoAji Gothic }
175 }

```

2.3.4 字體縮放率設定

设置全局中日字符缩放率的值。

```

176 \tl_new:N \g_@@_font_cjscale_tl
177 \keys_define:nn { ncls }
178 {
179     scale .tl_gset:N = \g_@@_font_cjsacle_tl,
180     scale .value_required:n = true,
181     scale .initial:n = { 0.924715 }
182 }

```

2.3.5 字體矩陣高級設定

设置全局标点特性。

```

183 \bool_new:N \g_@@_jfm_hanging_bool
184 \bool_new:N \g_@@_jfm_linegap_bool
185 \keys_define:nn { ncls }
186 {
187     punct .multichoice:,
188     punct / hanging .code:n = { \bool_gset_true:N \g_@@_jfm_hanging_bool },
189     punct / linegap .code:n = { \bool_gset_true:N \g_@@_jfm_linegap_bool },
190     punct .value_required:n = true
191 }

```

2.3.6 視覺字號補正

判断是否对 NFSS 视觉字号进行补正。

```

192 \bool_new:N \g_@@_font_xreal_bool
193 \keys_define:nn { ncls }
194 {
195     magstyle .choice:,
196     magstyle / real .code:n = { \bool_gset_false:N \g_@@_font_xreal_bool },
197     magstyle / xreal .code:n = { \bool_gset_true:N \g_@@_font_xreal_bool },
198     magstyle .value_required:n = true,
199     magstyle .initial:n = { xreal }
200 }

```

2.3.7 回滾字體特性

设置（可选）的回滚字体特性。其依赖 luaotfload 宏集的实验特性，危险呐。

```

201 \bool_new:N \g_@@_font_fallback_bool
202 \tl_new:N \g_@@_font_fallback_tl
203 \keys_define:nn { ncls }

```

```

204     {
205         fallback .code:n =
206         {
207             \bool_gset_true:N \g_@@_font_fallback_bool
208             \tl_gset:Nn \g_@@_font_fallback_tl { #1 }
209         }
210         fallback .value_required:N = true
211     }

```

2.4 初始化處理

使用封裝的宏處理用戶設置。

```

212 \@@_keyoptions_process:n { ncls }

```

3 主要特性

3.1 紙張尺寸配置

3.1.1 全局宏申明

申明存储纸张尺寸信息的特性列表。

```

213 \prop_new:N \g_@@_paper_sizelist_prop

```

用户指定、暂时存储的字列表已定义，此处存储最终数据的逗号列表及纸长度及宽度的全局申明。

```

214 \clist_new:N \g_@@_paper_sizeconf_clist
215 \dim_new:N \g_@@_paper_width_dim
216 \dim_new:N \g_@@_paper_height_dim

```

以及两个存储长、宽的局部宏。

```

217 \tl_new:N \l_@@_paper_widthaux_tl
218 \tl_new:N \l_@@_paper_heightaux_tl

```

3.1.2 主要功能宏

随后定义用于添加尺寸信息的宏。

```

219 \cs_new:Nn \@@_paper_addsize:nnn
220 {
221     \prop_put_if_new:Nnn \g_@@_paper_sizelist_prop
222     { #1 }
223     { { #2 }, { #3 } }
224 }

```

3.1.3 尺寸參數設定

通过__ncls_addpapersize:nnn 设置具体参数。

```

225 \@@_paper_addsize:nnn { a0 } { { 841 mm } { 1189 mm } }
226 \@@_paper_addsize:nnn { a1 } { { 594 mm } { 841 mm } }
227 \@@_paper_addsize:nnn { a2 } { { 420 mm } { 594 mm } }

```

```

228 \@@_paper_addsize:nnn { a3 } { 297 mm } { 420 mm }
229 \@@_paper_addsize:nnn { a4 } { 210 mm } { 297 mm }
230 \@@_paper_addsize:nnn { a5 } { 148 mm } { 210 mm }
231 \@@_paper_addsize:nnn { a6 } { 105 mm } { 148 mm }
232 \@@_paper_addsize:nnn { b0 } { 1000 mm } { 1414 mm }
233 \@@_paper_addsize:nnn { b1 } { 707 mm } { 1000 mm }
234 \@@_paper_addsize:nnn { b2 } { 500 mm } { 707 mm }
235 \@@_paper_addsize:nnn { b3 } { 353 mm } { 500 mm }
236 \@@_paper_addsize:nnn { b4 } { 250 mm } { 353 mm }
237 \@@_paper_addsize:nnn { b5 } { 176 mm } { 250 mm }
238 \@@_paper_addsize:nnn { b6 } { 125 mm } { 176 mm }
239 \@@_paper_addsize:nnn { c0 } { 917 mm } { 1297 mm }
240 \@@_paper_addsize:nnn { c1 } { 648 mm } { 917 mm }
241 \@@_paper_addsize:nnn { c2 } { 458 mm } { 648 mm }
242 \@@_paper_addsize:nnn { c3 } { 324 mm } { 458 mm }
243 \@@_paper_addsize:nnn { c4 } { 229 mm } { 324 mm }
244 \@@_paper_addsize:nnn { c5 } { 162 mm } { 229 mm }
245 \@@_paper_addsize:nnn { c6 } { 114 mm } { 162 mm }
246 \@@_paper_addsize:nnn { b0j } { 1030 mm } { 1456 mm }
247 \@@_paper_addsize:nnn { b1j } { 728 mm } { 1030 mm }
248 \@@_paper_addsize:nnn { b2j } { 515 mm } { 728 mm }
249 \@@_paper_addsize:nnn { b3j } { 364 mm } { 515 mm }
250 \@@_paper_addsize:nnn { b4j } { 257 mm } { 364 mm }
251 \@@_paper_addsize:nnn { b5j } { 182 mm } { 257 mm }
252 \@@_paper_addsize:nnn { b6j } { 128 mm } { 182 mm }
253 \@@_paper_addsize:nnn { screen } { 225 mm } { 180 mm }

```

3.1.4 内部参数处理

处理用户设定「一」：处理键对值列表的两种分支情况。

```

254 \prop_if_in:NoT \g_@@_paper_sizelist_prop
255 { \tl_to_str:N \g_@@_paper_sizeinfo_tl }
256 {
257   \prop_get:NoN \g_@@_paper_sizelist_prop
258   { \tl_to_str:N \g_@@_paper_sizeinfo_tl }
259   \g_@@_paper_sizeinfo_tl
260 }

```

处理用户设定「二」：处理字列表，使用逗号列表将长、宽分离。

```

261 \clist_gset:Nx \g_@@_paper_sizeconf_clist
262 { \g_@@_paper_sizeinfo_tl }
263 \clist_gpop:NN \g_@@_paper_sizeconf_clist \l_@@_paper_widthaux_tl
264 \clist_gpop:NN \g_@@_paper_sizeconf_clist \l_@@_paper_heightaux_tl

```

3.1.5 页面方向

处理页面方向选项。

```

265 \bool_if:NTF \g_@@_paper_portrait_bool
266 {

```



```

267     \dim_gset:Nn \g_@@_paper_width_dim
268     { \tl_use:N \l_@@_paper_widthaux_tl }
269     \dim_gset:Nn \g_@@_paper_height_dim
270     { \tl_use:N \l_@@_paper_heightaux_tl }
271   }
272   {
273     \dim_gset:Nn \g_@@_paper_width_dim
274     { \tl_use:N \l_@@_paper_heightaux_tl }
275     \dim_gset:Nn \g_@@_paper_height_dim
276     { \tl_use:N \l_@@_paper_widthaux_tl }
277   }

```

3.1.6 完成設置

完成纸张给配置。

```

278   \pdf_pagesize_gset:nn
279   { \dim_use:N \g_@@_paper_width_dim }
280   { \dim_use:N \g_@@_paper_height_dim }

```

3.1.7 清除内存

并做好内存管理。

```

281   \@@_aftercls_addtodel:N \@@_paper_addsize:nnn
282   \@@_aftercls_addtodel:N \g_@@_paper_sizelist_prop
283   \@@_aftercls_addtodel:N \g_@@_paper_sizeinfo_tl
284   \@@_aftercls_addtodel:N \g_@@_paper_sizeaux_tl
285   \@@_aftercls_addtodel:N \g_@@_paper_sizeconf_clist
286   \@@_aftercls_addtodel:N \g_@@_paper_width_dim
287   \@@_aftercls_addtodel:N \g_@@_paper_height_dim
288   \@@_aftercls_addtodel:N \g_@@_paper_portrait_bool
289   \@@_aftercls_addtodel:N \l_@@_paper_widthaux_tl
290   \@@_aftercls_addtodel:N \l_@@_paper_heightaux_tl

```

3.2 字體矩陣配置

3.2.1 全局宏申明

全局逗号列表申明。

```

291   \clist_new:N \g_@@_jfm_feats_clist

```

3.2.2 主要

其定义及作用域分散于后二节中，此章仅为占位。初始化其。

```

292   \clist_gset:Nn \g_@@_jfm_feats_clist { nstd }

```

3.2.3 内存管理

于最后清除之。

```

293   \@@_aftercls_addtodel:N \g_@@_jfm_feats_clist

```

3.3 組版方向配置

3.3.1 鈎子及定義

支持纵排组版，使用钩子进行处理。

```
294 \bool_if:NT \g_@@_dir_tate_bool
295 {
296   \RequirePackage { lltxtext } \tate
297   \@@_doc_beg:n
298   {
299     \message { 《縦組モード》 } \adjustbaseline
300   }
301 }
```

3.3.2 字體矩陣配置

「疑」同时配置对应的字体矩阵特性。（其逗号列表的宏定义将在「字体矩阵配置・内存管理」中被定义与清除。）

```
302 % \bool_if:NT \g_@@_dir_tate_bool
303 % { \clist_gput_left:Nn \g_@@_jfm_feats_clist { vert } }
```

3.3.3 内存管理

清除存储对方向之判断的宏。

```
304 \@@_aftercls_addtodel \g_@@_dir_tate_bool
```

3.4 字體相關配置

3.4.1 全局宏申明

申明 OpenType 字体特性的字列表。

```
305 \tl_new:N \g_@@_font_langfeat_tl
```

3.4.2 主要設定

对各语言分别设置字体矩阵特性及 OpenType 字体特性。

```
306 \str_if_eq:VnT \g_@@_lang_cj_str { t }
307 {
308   \clist_gput_left:Nn \g_@@_jfm_feats_clist { trad }
309   \tl_gset:Nn \g_@@_font_langfeat_tl { Chinese~Traditional }
310 }
311 \str_if_eq:VnT \g_@@_lang_cj_str { s }
312 {
313   \clist_gput_left:Nn \g_@@_jfm_feats_clist { simpl }
314   \tl_gset:Nn \g_@@_font_langfeat_tl { Chinese~Simplified }
315 }
316 \str_if_eq:VnT \g_@@_lang_cj_str { j }
317 {
318   \clist_gput_left:Nn \g_@@_jfm_feats_clist { jp }
```

```

319     \tl_gset:Nn \g_@@_font_langfeat_tl { Japanese }
320 }

```

3.4.3 字體矩陣高級設置

设置字体矩阵的标点高级特性。

```

321     \bool_if:NT \g_@@_jfm_hanging_bool
322     { \clist_gput_left:Nn \g_@@_jfm_feats_clist { hgp } }
323     \bool_if:NT \g_@@_jfm_linegap_bool
324     { \clist_gput_left:Nn \g_@@_jfm_feats_clist { lgp } }

```

3.4.4 載入中日文支持宏集

载入 Lua_T_EX-ja 宏集。准备好预定义。

```

325     \tl_gset:Nx \Cjascale { \tl_use:N \g_@@_font_cjscale_tl }
326     \tl_gset:Nx \ltj@stdmcfont { \tl_use:N \g_@@_font_mincho_tl }
327     \tl_gset:Nx \ltj@stdgtfont { \tl_use:N \g_@@_font_gothic_tl }
328     \tl_gset:Nx \ltj@stdyokojfm
329     { eva / { \clist_use:Nn \g_@@_jfm_feats_clist { , } } }
330     \tl_gset:Nx \ltj@stdtatejfm
331     { eva / { \clist_use:Nn \g_@@_jfm_feats_clist { , } , vert } }
332     \RequirePackage { luatexja }

```

3.4.5 補正用單位

设置补正用 point 单位，依据缩放率。

```

333     \dim_gset:Nn \mpt { \g_@@_font_magscale_tl \p@ }

```

同时对\@ptsize 采取同 bxjs 及 ltj 文档类相同之策略。「餘」

```

334     \dim_compare:nNnT
335     { \mpt } < { 1 \p@ }
336     { \tl_gset:Nn \@ptsize { -20 } }
337     \dim_compare:nNnT
338     { \mpt } = { 1 \p@ }
339     { \tl_gset:Nn \@ptsize { 0 } }
340     \dim_compare:nNnT
341     { \mpt } = { 1.095 \p@ }
342     { \tl_gset:Nn \@ptsize { 1 } }
343     \dim_compare:nNnT
344     { \mpt } = { 1.200 \p@ }
345     { \tl_gset:Nn \@ptsize { 2 } }
346     \dim_compare:nNnT
347     { \mpt } > { 1.200 \p@ }
348     { \tl_gset:Nn \@ptsize { -20 } }

```

3.4.6 視覺字號補正

对是否补正时统一单位\mpt 进行处理，而当字号本就无需\mag 时使补正失效。

```

349     \bool_if:NTF \g_@@_font_xreal_bool

```

```

350 {
351   \dim_compare:nNnT
352     { \mpt } = { \p@ }
353     { \bool_gset_false:N \g_@@_font_xreal_bool }
354 }
355 { \dim_gset:Nn \mpt { \p@ } }

```

实际补正。注意编码等，以及 expl3 与 $\text{\LaTeX} 2_{\epsilon}$ 的兼容性（需小心维护）。

```

356 \bool_if:NT \g_@@_font_xreal_bool
357 {
358   \exp_after:wN \cs_set_eq:NN \cs:w TU/lmr/m/n/10 \cs_end: \scan_stop:
359   \exp_after:wN \cs_set_eq:NN \cs:w TU/lmss/m/n/10 \cs_end: \scan_stop:
360   \exp_after:wN \cs_set_eq:NN \cs:w TU/lmtt/m/n/10 \cs_end: \scan_stop:
361   \@@_luafunc_new:N \@@_magnify_font_calc
362   \group_begin:
363     % \char_set_catcode_other:N \$
364     \char_set_catcode_other:N \%
365     \char_set_catcode_space:n { 32 }
366     \lua_now:e
367     {
368       local mpt = tex.getdimen('mpt')/65536
369       lua.get_functions_table()[\the\@@_magnify_font_calc] = function()
370         tex.sprint(-2, math.floor(0.5 + mpt * tex.getdimen(luatexbase.registernumber 'dimen@')))
371       end
372       function luatexja.ncls_unmagnify_fsize(e)
373         local s = luatexja.print_scaled(floor(0.5 + e / mpt * 65536))
374         tex.sprint(-2, (s:match('%.\0\$')) and s:sub(1, -3) or s)
375       end
376     }
377   \group_end:
378   \cs_new:Npn \@@_magnify_external_font:w #1~at #2~at #3 \@nil
379   {
380     \tl_set:Nn \l_tempa_tl { #1 }
381     \tl_set:Nn \l_tempb_tl { #2 }
382     \tl_if_empty:NTF \l_tempb_tl
383     {
384       \tl_set:Nx \l_tempb_tl
385       {
386         scaled \lua_now:e { tex.sprint(-2, math.floor(0.5 + \g_@@_font_magscale_tl * 1000)) }
387       }
388     }
389     {
390       \dim_set:Nn \dimen@ { \tl_use:N \l_tempb_tl }
391       \tl_set:Nx \l_tempb_tl
392       { at \@@_luafunc_use:N \@@_magnify_font_calc~sp }
393     }
394     \tl_set:Nx \l_tempa_tl
395     {
396       \tl_set:Nn \exp_not:N \external@font

```

```

397         { \tl_use:N \l_tempa_tl \tl_use:N \l_tempb_tl }
398     }
399 }
400 \cs_new_eq:NN \@@_get_externalfont_orig: { \get@external@font }
401 \cs_gset:Nn \get@external@font
402 {
403     \tl_gset:Nx \f@size
404     { \lua_now:e { luatexja.ncls_unmagnify_fsize(\f@size) } }
405     \@@_get_externalfont_orig:
406     \group_begin:
407         \tl_set:Nx \l_tempa_tl
408         { \external@font \tl_use:N \c_space_tl~at \tl_use:N \c_space_tl~at }
409         \exp_after:wN \@@_magnify_external_font:w \tl_use:N \l_tempa_tl \@nil
410         \exp_after:wN
411     \group_end:
412     \tl_use:N \l_tempa_tl
413 }
414 }

```

NFSS 魔改结束，注意其内部宏的局部及全局命名空间。此部分不进行优化。

3.4.7 回滾字體可選特性

处理前面键对值取到的用户设定。有对是否激活的判断。

```

415 \bool_if:NT \g_@@_font_fallback_bool
416 {
417     \group_begin:
418     \char_set_catcode_space:n { 32 }
419     \lua_now:e
420     {
421         luaotfload.add_fallback
422         (
423             " nclsfallback ",
424             { " \tl_use:N \g_@@_font_fallback_tl : mode = node ; script = cjk ; language = \tl_use:N \g_@@_font_langfeat_t
425         )
426     }
427     \group_end:
428 }

```

3.4.8 中日 NFSS 設定

先防止标记这些字体的记号被展开。

```

429 \exp_after:wN \cs_set_eq:NN \cs:w JY3/mc/m/n/10 \cs_end: \scan_stop:
430 \exp_after:wN \cs_set_eq:NN \cs:w JY3/gt/m/n/10 \cs_end: \scan_stop:
431 \exp_after:wN \cs_set_eq:NN \cs:w JT3/mc/m/n/10 \cs_end: \scan_stop:
432 \exp_after:wN \cs_set_eq:NN \cs:w JT3/gt/m/n/10 \cs_end: \scan_stop:

```

然后封装定义字体参数的命令（ $\text{\LaTeX 2}_{\epsilon}$ 提供）。如此可以简单地在将来挂接更多稀奇古怪的特性。然目前只有两个分支。

```

433 \cs_new:Nn \g_@@_font_declareshape:nnnn
434 {
435   \bool_if:NTF \g_@@_font_fallback_bool
436   {
437     \DeclareFontShape { #1 } { #2 } { m } { n }
438     {
439       <-> s * [ \tl_use:N \g_@@_font_cjsacale_tl ] #3 :
440       - kern ; script = cjk ; language = \tl_use:N \g_@@_font_langfeat_tl ;
441       jfm = { eva / { \clist_use:Nn \g_@@_jfm_feats_clist { , } #4 } } ;
442       fallback = nclsfallback
443     }
444   }
445   {
446     \DeclareFontShape { #1 } { #2 } { m } { n }
447     {
448       <-> s * [ \tl_use:N \g_@@_font_cjsacale_tl ] #3 :
449       - kern ; script = cjk ; language = \tl_use:N \g_@@_font_langfeat_tl ;
450       jfm = { eva / { \clist_use:Nn \g_@@_jfm_feats_clist { , } #4 } }
451     }
452   }
453 }

```

终于，定义四个源字体。

```

454 \g_@@_font_declareshape:nnnn { JY3 } { mc } { \tl_use:N \g_@@_font_mincho_tl } { }
455 \g_@@_font_declareshape:nnnn { JY3 } { gt } { \tl_use:N \g_@@_font_gothic_tl } { }
456 \g_@@_font_declareshape:nnnn { JT3 } { mc } { \tl_use:N \g_@@_font_mincho_tl } { ,vert }
457 \g_@@_font_declareshape:nnnn { JT3 } { gt } { \tl_use:N \g_@@_font_gothic_tl } { ,vert }

```

然后偷懒，用递归定义其余分支字体。至于 delux 就以后再加吧。

```

458 \clist_map_inline:nn { JY3, JT3 }
459 {
460   \clist_map_inline:nn { n, it, sl, sc }
461   {
462     \clist_map_inline:nn { m, b, bx, sb }
463     {
464       \bool_if:nF { \str_if_eq_p:nn { ##1 } { n } && \str_if_eq_p:nn { #####1 } { m } }
465       { \DeclareFontShape { #1 } { gt } { #####1 } { ##1 } { <-> ssub * gt/m/n } { } }
466     }
467     \str_if_eq:nn { ##1 } { n }
468     { \DeclareFontShape { #1 } { mc } { m } { ##1 } { <-> ssub * mc/m/n } { } }
469     \clist_map_inline:nn { b, bx, sb }
470     { \DeclareFontShape { #1 } { mc } { #####1 } { ##1 } { <-> ssub * gt/m/n } { } }
471   }
472 }

```

3.4.9 清理内存

内存管理。清除不必要的宏定义。

```

473 \@@_aftercls_addtodel:N \g_@@_font_magscale_tl

```

```

474 \@@_aftercls_addtodel:N \g_@@_lang_cj_str
475 \@@_aftercls_addtodel:N \g_@@_font_langfeat_tl
476 \@@_aftercls_addtodel:N \g_@@_font_mincho_tl
477 \@@_aftercls_addtodel:N \g_@@_font_gothic_tl
478 \@@_aftercls_addtodel:N \g_@@_font_cjscale_tl
479 \@@_aftercls_addtodel:N \g_@@_jfm_hanging_tl
480 \@@_aftercls_addtodel:N \g_@@_jfm_linegap_tl
481 \@@_aftercls_addtodel:N \g_@@_font_xreal_bool
482 \@@_aftercls_addtodel:N \g_@@_font_fallback_bool
483 \@@_aftercls_addtodel:N \g_@@_font_fallback_tl
484 \@@_aftercls_addtodel:N \@@_font_declareshape:nmmn

```

Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in *roman* refer to the code lines where the entry is used.

Symbols	\char 363–365, 418	G
\\$ 363, 374	\Cjascale 325	\g 5, 50, 52, 56, 74, 77, 81, 85, 86, 97, 101, 102, 113, 117–134, 148, 152–154, 165, 166, 169, 170, 176, 179, 183, 184, 188, 189, 192, 196, 197, 201, 202, 207, 208, 213–216, 221, 254, 255, 257–259, 261–265, 267, 269, 273, 275, 279, 280, 282–288, 291–294, 302–306, 308, 309, 311, 313, 314, 316, 318, 319, 321–327, 329, 331, 333, 349, 353, 356, 386, 415, 424, 435, 439–441, 448–450, 454–457, 473–483
\% 364	\clist 214, 261, 263, 264, 291, 292, 303, 308, 313, 318, 322, 324, 329, 331, 441, 450, 458, 460, 462, 469	\get@external@font 400, 401
\@@ 9, 12, 47–49, 51, 53, 59, 60, 67–73, 212, 219, 225–253, 281–290, 293, 297, 304, 361, 369, 378, 392, 400, 405, 409, 433, 454–457, 473–484	\cs 6, 8, 9, 12, 14, 47–49, 51, 56, 59, 60, 219, 358–360, 378, 400, 401, 429–433	\group . . 362, 377, 406, 411, 417, 427
\@ifl@t@r 27		
\@ifpackagelater 16		
\@nil 378, 409		
\@ptsize 336, 339, 342, 345, 348		
\@ 22, 32, 42, 64		
A	D	
\adjustbaseline 299	\DeclareFontShape 437, 446, 465, 468, 470	
\AtBeginDocument 49	\dim 215, 216, 267, 269, 273, 275, 279, 280, 333, 334, 337, 340, 343, 346, 351, 355, 390	
\AtEndOfClass 48	\dimen@ 390	
\AtEndPreamble 14, 47	E	
	\exp 92, 108, 143, 160, 358–360, 396, 409, 410, 429–432	K
B	\ExplSyntaxOff 57	\keys 75, 82, 98, 114, 149, 167, 177, 185, 193, 203
\bool 81, 85, 86, 97, 101, 102, 183, 184, 188, 189, 192, 196, 197, 201, 207, 265, 294, 302, 321, 323, 349, 353, 356, 415, 435, 464	\ExplSyntaxOn 55	L
	\external@font 396, 408	\l 217, 218, 263, 264, 268, 270, 274, 276, 289, 290, 380–382, 384, 390, 391, 394, 397, 407, 409, 412
C	F	\ltj@stdgtfont 327
\c 408	\f@size 403, 404	
	\fmtversion 27	

\ltj@stdmcfont 326
\ltj@stdtatejfm 330
\ltj@stdyokojfm 328
\lua 366, 386, 404, 419
\luafunction 60

M

\message 299
\mpt 333, 335,
338, 341, 344, 347, 352, 355
\msg 18, 25, 29,
35, 39, 45, 61, 89, 105, 137, 157

N

\NeedsTeXFormat 1
\NewDocumentCommand 6
\newluafunction 59

P

\p@ 333, 335,
338, 341, 344, 347, 352, 355
\pdf 278
\ProcessKeyOptions 8, 9
\ProcessKeysOptions 12
\prop 5, 213, 221, 254, 257
\ProvidesExplClass 3

R

\RequirePackage 2, 7, 11, 15, 296, 332

S

\scan 358–360, 429–432
\seq 50, 52
\str 148, 152–154,
306, 311, 316, 464, 467

\sys 37

T

\tate 296
\the 369
\tl 74, 113, 117–134,
165, 166, 176, 202, 208,
217, 218, 255, 258,
268, 270, 274, 276, 305,
309, 314, 319, 325–328,
330, 336, 339, 342, 345,
348, 380–382, 384, 390,
391, 394, 396, 397, 403,
407–409, 412, 424, 439,
440, 448, 449, 454–457