

年文檔類 Nian Class

黃 京

西曆 2023 年 5 月 31 日

概要

为在 LuaTeX 下排印中日本而作的文档类。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}{2023-05-20}{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:NIF \ProcessKeyOptions
9   { \cs_new_protected:Nn \@@_keyoptions_process:n { \ProcessKeyOptions { #1 } } }
10  {
11    \RequirePackage { l3keys2e }
12    \cs_new_protected: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 \AtEndPreamble 钩子。

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

「疑」定义 \AtEndOfClass 钩子。

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

定义 \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_new:Nn \@@_macro_release:N
52 { \seq_gput_right:Nn \g_@@_aftercls_del_seq { #1 } }
53 \@@_at_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_protected:Npn \@@_luafunc_new:N { \newluafunction }
60 \cs_new_protected: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 }

```

同时为字体缩放\mag=xreal 预定义同一单位，见「编译模式」处首次使用时的注释。

```

67 \dim_new:N \mpt
68 \dim_set:Nn \mpt { \p@ }

```

1.3 内存清理

在导言区末尾清除所有非必要宏。

```

69 \@@_macro_release:N \@@_keyoptions_process:n
70 \@@_macro_release:N \@@_at_preamble_end:n
71 % \@@_macro_release:N \@@_at_doc_end:n
72 \@@_macro_release:N \@@_at_doc_begin:n
73 \@@_macro_release:N \@@_macro_release:N
74 \@@_macro_release:N \@@_luafunc_new:N
75 \@@_macro_release: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。

```

76 \tl_new:N \l_@@_paper_sizeinfo_tl
77 \keys_define:nn { ncls }
78 {
79   paper .tl_set:N = \l_@@_paper_sizeinfo_tl,
80   paper .value_required:n = true,
81   peper .initial:n = { a4 }
82 }

```

2.1.2 輔助線

是否需要辅助线。为了方便分类，将 tombow 和 mentuke 都并入 corpmark 类，并扔掉了 tombo。「补完」

```

83 \bool_new:N \l_@@_paper_corpmark_mark_bool
84 \bool_new:N \l_@@_paper_corpmark_date_bool
85 \keys_define:nn { ncls }
86 {
87   corpmark .choice:,
88   corpmark / tombow .code:n =
89   {
90     \bool_set_true:N \l_@@_paper_corpmark_mark_bool

```

```

91     \bool_set_true:N \l_@@_paper_corpmark_date_bool
92 },
93 corpmark / mentuke .code:n =
94 {
95     \bool_set_true:N \l_@@_paper_corpmark_mark_bool
96     \bool_set_false:N \l_@@_paper_corpmark_date_bool
97 },
98 corpmark / unknown .code:n =
99 {
100     \msg_error:nnxxx { ncls } { unknown-choice }
101     { corpmark }
102     { tombow,~mentuke }
103     { \exp_not:n { #1 } }
104 },
105 corpmark .value_required:n = true
106 }

```

2.1.3 頁面方向

设置页面方向。

```

107 \bool_new:N \l_@@_paper_portrait_bool
108 \keys_define:nn { ncls }
109 {
110     orientation .choice:,
111     orientation / portrait .code:n = { \bool_set_true:N \l_@@_paper_portrait_bool },
112     orientation / landscape .code:n = { \bool_set_false:N \l_@@_paper_portrait_bool },
113     orientation / unknown .code:n =
114     {
115         \msg_error:nnxxx { ncls } { unknown-choice }
116         { orientation }
117         { portrait,~landscape }
118         { \exp_not:n { #1 } }
119     },
120     orientation .value_required:n = true,
121     orientation .initial:n = { portrait }
122 }

```

2.2 選項設定

2.2.1 組版方向

确定使用竖书或是横排。

```

123 \bool_new:N \l_@@_layout_tate_bool
124 \keys_define:nn { ncls }
125 {
126     direction .choice:,
127     direction / yoko .code:n = { \bool_set_false:N \l_@@_layout_tate_bool },
128     direction / tate .code:n = { \bool_set_true:N \l_@@_layout_tate_bool },
129     direction / unknown .code:n =

```

```

130     {
131         \msg_new:nxxxx { ncls } { unknown-choice }
132         { direction }
133         { yoko,~tate }
134         { \exp_not:n { #1 } }
135     },
136     direction .value_required:n = true,
137     direction .initial:n = { yoko }
138 }

```

2.2.2 文檔類型

设置文档类型（全局）为文章、论文报告、或书籍。依赖错综复杂，故不使用布尔类型。

```

139 \str_new:N \l_@@_layout_doctype_str
140 \keys_define:nn { ncls }
141 {
142     doctype .choice:,
143     doctype / article .code:n = { \str_set:Nn \l_@@_layout_doctype_str { a } },
144     doctype / report .code:n = { \str_set:Nn \l_@@_layout_doctype_str { r } },
145     doctype / book .code:n = { \str_set:Nn \l_@@_layout_doctype_str { b } },
146     doctype .value_required:n = true,
147     doctype .initial:n = { article }
148 }

```

2.2.3 分欄

设置全局双栏或单栏。这里只是处理，待后设置。

```

149 \bool_new:N \l_@@_layout_restonecol_bool
150 \keys_define:nn { ncls }
151 {
152     column .choice:,
153     column / one .code:n = { \bool_set_false:N \l_@@_layout_restonecol_bool },
154     column / two .code:n = { \bool_set_true:N \l_@@_layout_restonecol_bool },
155     column / unknown .code:n =
156     {
157         \msg_error:nxxxx { ncls } { unknown-choice }
158         { column }
159         { one,~two }
160         { \exp_not:n { #1 } }
161     },
162     column .value_required:n = true,
163     column .initial:n = { one }
164 }

```

然后顺便设置双栏的间距。因为其属于页面尺寸参数需要在初始时通过文档类的选项设定避免多次计算页面故在这里设置。（好长一句话）

```

165 \tl_new:N \l_@@_layout_column_gap_tl
166 \keys_define:nn { ncls }
167 {

```

```

168     column_gap .tl_set:N = \l_@@_layout_column_gap_tl,
169     column_gap .value_required:n = true,
170     column_gap .initial:n = { 2 \zw }
171 }

```

2.2.4 單雙面

设置是单面列印抑或是双面。事关边距，使用两个布尔参数小心处理。同时，此与文档类型有关（依存系）。

```

172 \bool_new:N \l_@@_layout_twoside_bool
173 \bool_new:N \l_@@_layout_mparswitch_bool
174 \keys_define:nn { ncls }
175 {
176     print .choice:,
177     print / oneside .code:n =
178     {
179         \bool_set_false:N \l_@@_layout_twoside_bool
180         \bool_set_false:N \l_@@_layout_mparswitch_bool
181     },
182     print / twoside .code:n =
183     {
184         \bool_set_true:N \l_@@_layout_twoside_bool
185         \bool_set_true:N \l_@@_layout_mparswitch_bool
186     },
187     print / vartwoside .code:n =
188     {
189         \bool_set_true:N \l_@@_layout_twoside_bool
190         \bool_set_false:N \l_@@_layout_mparswitch_bool
191     },
192     print / unknown .code:n =
193     {
194         \msg_error:nnxxx { ncls } { unknown-choice }
195         { print }
196         { oneside,~twoside,~vartwoside }
197         { \exp_not:n { #1 } }
198     },
199     print .value_required:n = true
200 }

```

2.2.5 標題頁

是否需要标题页。与文档类型有关（依存系）。

```

201 \bool_new:N \l_@@_layout_titlepage_bool
202 \keys_define:nn { ncls }
203 {
204     titlepage .bool_set:N = \l_@@_layout_titlepage_bool,
205     titlepage .value_required:n = true
206 }

```

2.2.6 章節起

如何开启新的章节呢？大概只针对论文报告及书籍。也用两个布尔类型实现判别。

```
207 \bool_new:N \l_@@_layout_openleft_bool
208 \bool_new:N \l_@@_layout_openright_bool
209 \keys_define:nn { ncls }
210 {
211   open .choice:,
212   open / left .code:n =
213   {
214     \bool_set_true:N \l_@@_layout_openleft_bool
215     \bool_set_false:N \l_@@_layout_openright_bool
216   },
217   open / right .code:n =
218   {
219     \bool_set_false:N \l_@@_layout_openleft_bool
220     \bool_set_true:N \l_@@_layout_openright_bool
221   },
222   open / any .code:n =
223   {
224     \bool_set_true:N \l_@@_layout_openleft_bool
225     \bool_set_true:N \l_@@_layout_openright_bool
226   },
227   open / unknown .code:n =
228   {
229     \msg_error:nnxxx { ncls } { unknown-choice }
230     { open }
231     { left,~right,~any }
232     { \exp_not:n { #1 } }
233   },
234   open .value_required:n = true
235 }
```

2.2.7 編譯模式

最终成品或暂时替代。其实就是坏盒子长度的区别（零或非零），当然此处也会用到一个布尔型值。由于此时`\mpt`还未被定义，故此处先将其设为`\p0`，用户指定时（希望）它会被正确定义。「需要补完」

```
236 \bool_new:N \l_@@_layout_draft_bool
237 \keys_define:nn { ncls }
238 {
239   version .choice:,
240   version / draft .code:n =
241   {
242     \bool_set_true:N \l_@@_layout_draft_bool
243     \dim_set:Nn \overfullrule { 6 \mpt }
244   },
245   version / final .code:n =
246   {
```

```

247     \bool_set_false:N \l_@@_layout_draft_bool
248     \dim_set:Nn \overfullrule { 0 \mpt }
249   },
250   version / unknown .code:n =
251   {
252     \msg_error:nnxxx { ncls } { unknown-choice }
253     { version }
254     { draft,~final }
255     { \exp_not:n { #1 } }
256   },
257   version .value_required:n = true,
258   version .initial:n = { final }
259 }

```

2.2.8 數學公式設定

此处设置数学公式的排版大方向，说人话就是对齐排列方式和编号方式。首先是对齐。

```

260 \bool_new:N \l_@@_layout_fleqn_bool
261 \keys_define:nn { ncls }
262 {
263   equalign .choice:,
264   equalign / left .code:n = { \bool_set_true:N \l_@@_layout_fleqn_bool },
265   equalign / middle .code:n = { \bool_set_false:N \l_@@_layout_fleqn_bool },
266   equalign / unknown .code:n =
267   {
268     \msg_error:nnxxx { ncls } { unknown-choice }
269     { equalign }
270     { left,~middle }
271     { \exp_not:n { #1 } }
272   },
273   equalign / value_required:n = true,
274   equalign / initial:n = { middle }
275 }

```

其次则是编号。

```

276 \bool_new:N \l_@@_layout_leqno_bool
277 \keys_define:nn { ncls }
278 {
279   eqnum .choice:,
280   eqnum / left .code:n = { \bool_set_true:N \l_@@_layout_leqno_bool },
281   eqnum / right .code:n = { \bool_set_false:N \l_@@_layout_leqno_bool },
282   eqnum / unknown .code:n =
283   {
284     \msg_error:nnxxx { ncls } { unknown-choice }
285     { eqnum }
286     { left,~right }
287     { \exp_not:n { #1 } }
288   },
289   eqnum .value_required:n = true,

```



```

290     eqnum .initial:n = { right }
291 }

```

2.2.9 參考文獻式樣設定

主要是支持 open bib 格式，虽然有些无聊。

```

292 \bool_new:N \l_@@_layout_openbib_bool
293 \keys_define:nn { ncls }
294 {
295     bibstyle .choice:,
296     bibstyle / open .code:n = { \bool_set_true:N \l_@@_layout_openbib_bool },
297     bibstyle / plain .code:n = { \bool_set_false:N \l_@@_layout_openbib_bool },
298     bibstyle / unknown .code:n =
299     {
300         \msg_error:nnxxx { ncls } { unknown-choice }
301         { bibstyle }
302         { open,~plain }
303         { \exp_not:n { #1 } }
304     },
305     bibstyle .value_required:n = ture,
306     bibstyle .initial:n = { plain }
307 }

```

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。

```

308 \tl_new:N \l_@@_font_magscale_tl
309 \keys_define:nn { ncls }
310 {
311     fontsize .choice:,
312     fontsize / 7pt .code:n = { \tl_set:Nn \l_@@_font_magscale_tl { 0.6940 } },
313     fontsize / 8pt .code:n = { \tl_set:Nn \l_@@_font_magscale_tl { 0.8330 } },
314     fontsize / 9pt .code:n = { \tl_set:Nn \l_@@_font_magscale_tl { 0.9130 } },
315     fontsize / 10pt .code:n = { \tl_set:Nn \l_@@_font_magscale_tl { 1.0000 } },
316     fontsize / 11pt .code:n = { \tl_set:Nn \l_@@_font_magscale_tl { 1.0953 } },
317     fontsize / 12pt .code:n = { \tl_set:Nn \l_@@_font_magscale_tl { 1.2000 } },
318     fontsize / 13pt .code:n = { \tl_set:Nn \l_@@_font_magscale_tl { 1.3000 } },
319     fontsize / 14pt .code:n = { \tl_set:Nn \l_@@_font_magscale_tl { 1.4400 } },
320     fontsize / 15pt .code:n = { \tl_set:Nn \l_@@_font_magscale_tl { 1.5000 } },
321     fontsize / 16pt .code:n = { \tl_set:Nn \l_@@_font_magscale_tl { 1.6000 } },
322     fontsize / 17pt .code:n = { \tl_set:Nn \l_@@_font_magscale_tl { 1.7280 } },
323     fontsize / 20pt .code:n = { \tl_set:Nn \l_@@_font_magscale_tl { 2.0000 } },
324     fontsize / 21pt .code:n = { \tl_set:Nn \l_@@_font_magscale_tl { 2.0740 } },
325     fontsize / 24pt .code:n = { \tl_set:Nn \l_@@_font_magscale_tl { 2.4000 } },
326     fontsize / 25pt .code:n = { \tl_set:Nn \l_@@_font_magscale_tl { 2.4880 } },
327     fontsize / 30pt .code:n = { \tl_set:Nn \l_@@_font_magscale_tl { 2.9860 } },
328     fontsize / 36pt .code:n = { \tl_set:Nn \l_@@_font_magscale_tl { 3.5830 } },

```

```

329     fontsize / 43pt .code:n = { \tl_set:Nn \l_@@_font_magscale_tl { 4.3000 } },
330     fontsize / unknown .code:n =
331     {
332         \msg_error:nnxxx { ncls } { unknown-choice }
333         { fontsize }
334         {
335             7pt,~ 8pt,~ 9pt,~ 10pt,~ 11pt,~ 12pt,~ 13pt,~ 14pt,~ 15pt,~
336             17pt,~ 20pt,~ 21pt,~ 24pt,~ 25pt,~ 30pt,~ 36pt,~ 43pt
337         }
338         { \exp_not:n { #1 } }
339     },
340     fontsize .value_required:n = true,
341     fontsize .initial:n = { 10pt }
342 }

```

2.3.2 語言設定

设置文档类之全局语言。

```

343 \str_new:N \l_@@_lang_cj_str
344 \keys_define:nn { ncls }
345 {
346     language .choice:,
347     language / trad .code:n = { \str_set:Nn \l_@@_lang_cj_str { t } },
348     language / smpl .code:n = { \str_set:Nn \l_@@_lang_cj_str { s } },
349     language / jp .code:n = { \str_set:Nn \l_@@_lang_cj_str { j } },
350     language / unknown .code:n =
351     {
352         \msg_error:nnxxx { ncls } { unknown-choice }
353         { language }
354         { trad,~smpl,~jp }
355         { \exp_not:n { #1 } }
356     },
357     language .value_required:n = true,
358     language .initial:n = { jp }
359 }

```

2.3.3 字體設定

设定全局明朝及哥特字体。

```

360 \tl_new:N \l_@@_font_mincho_tl
361 \tl_new:N \l_@@_font_gothic_tl
362 \keys_define:nn { ncls }
363 {
364     mincho .tl_set:N = \l_@@_font_mincho_tl,
365     gothic .tl_set:N = \l_@@_font_gothic_tl,
366     mincho .value_required:n = true,
367     gothic .value_required:n = true,
368     mincho .initial:n = { HaranoAji Mincho },
369     gothic .initial:n = { HaranoAji Gothic }

```

```
370 }
```

2.3.4 字體縮放率設定

设置全局中日字符缩放率的值。

```
371 \tl_new:N \l_@@_font_cjscale_tl
372 \keys_define:nn { ncls }
373 {
374   scale .tl_set:N = \l_@@_font_cjsacle_tl,
375   scale .value_required:n = true,
376   scale .initial:n = { 0.924715 }
377 }
```

2.3.5 字體矩陣高級設定

设置全局标点特性。

```
378 \bool_new:N \l_@@_jfm_hanging_bool
379 \bool_new:N \l_@@_jfm_linegap_bool
380 \keys_define:nn { ncls }
381 {
382   punct .multichoice:,
383   punct / hanging .code:n = { \bool_set_true:N \l_@@_jfm_hanging_bool },
384   punct / linegap .code:n = { \bool_set_true:N \l_@@_jfm_linegap_bool },
385   punct .value_required:n = true
386 }
```

2.3.6 視覺字號補正

判断是否对 NFSS 视觉字号进行补正。

```
387 \bool_new:N \l_@@_font_xreal_bool
388 \keys_define:nn { ncls }
389 {
390   magstyle .choice:,
391   magstyle / real .code:n = { \bool_set_false:N \l_@@_font_xreal_bool },
392   magstyle / xreal .code:n = { \bool_set_true:N \l_@@_font_xreal_bool },
393   magstyle .value_required:n = true,
394   magstyle .initial:n = { xreal }
395 }
```

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。

```
396 \bool_new:N \l_@@_font_fallback_bool
397 \tl_new:N \l_@@_font_fallback_tl
398 \keys_define:nn { ncls }
399 {
400   fallback .code:n =
401   {
```

```

402     \bool_set_true:N \l_@@_font_fallback_bool
403     \tl_set:Nn \l_@@_font_fallback_tl { #1 }
404   },
405   fallback .value_required:n = true
406 }

```

2.4 初始化處理

使用封装的宏处理用户设置。

```

407 \@@_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。

```

408 \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「优化」的全局申明。

```

409 \clist_new:N \g_@@_paper_sizeconf_clist
410 % \dim_new:N \g_@@_paper_width_dim
411 % \dim_new:N \g_@@_paper_height_dim

```

「优化」以及两个存储长、宽的局部宏。

```

412 % \tl_new:N \l_@@_paper_widthaux_tl
413 % \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。

```

414 \cs_new:Nn \@@_paper_addsize:nnn
415 {
416   \prop_gput_if_new:Nnn \g_@@_paper_sizelist_prop
417     { #1 }
418     { #2 , #3 }
419 }

```

3.1.3 尺寸參數設定

通过__ncls_addpapersize:nnn 设置具体参数。

```

420 \@@_paper_addsize:nnn { a0 } { 841 mm } { 1189 mm }
421 \@@_paper_addsize:nnn { a1 } { 594 mm } { 841 mm }
422 \@@_paper_addsize:nnn { a2 } { 420 mm } { 594 mm }
423 \@@_paper_addsize:nnn { a3 } { 297 mm } { 420 mm }
424 \@@_paper_addsize:nnn { a4 } { 210 mm } { 297 mm }
425 \@@_paper_addsize:nnn { a5 } { 148 mm } { 210 mm }

```

```

426 \@@_paper_addsize:nnn { a6 } { { 105 mm } { 148 mm } }
427 \@@_paper_addsize:nnn { b0 } { { 1000 mm } { 1414 mm } }
428 \@@_paper_addsize:nnn { b1 } { { 707 mm } { 1000 mm } }
429 \@@_paper_addsize:nnn { b2 } { { 500 mm } { 707 mm } }
430 \@@_paper_addsize:nnn { b3 } { { 353 mm } { 500 mm } }
431 \@@_paper_addsize:nnn { b4 } { { 250 mm } { 353 mm } }
432 \@@_paper_addsize:nnn { b5 } { { 176 mm } { 250 mm } }
433 \@@_paper_addsize:nnn { b6 } { { 125 mm } { 176 mm } }
434 \@@_paper_addsize:nnn { c0 } { { 917 mm } { 1297 mm } }
435 \@@_paper_addsize:nnn { c1 } { { 648 mm } { 917 mm } }
436 \@@_paper_addsize:nnn { c2 } { { 458 mm } { 648 mm } }
437 \@@_paper_addsize:nnn { c3 } { { 324 mm } { 458 mm } }
438 \@@_paper_addsize:nnn { c4 } { { 229 mm } { 324 mm } }
439 \@@_paper_addsize:nnn { c5 } { { 162 mm } { 229 mm } }
440 \@@_paper_addsize:nnn { c6 } { { 114 mm } { 162 mm } }
441 \@@_paper_addsize:nnn { b0j } { { 1030 mm } { 1456 mm } }
442 \@@_paper_addsize:nnn { b1j } { { 728 mm } { 1030 mm } }
443 \@@_paper_addsize:nnn { b2j } { { 515 mm } { 728 mm } }
444 \@@_paper_addsize:nnn { b3j } { { 364 mm } { 515 mm } }
445 \@@_paper_addsize:nnn { b4j } { { 257 mm } { 364 mm } }
446 \@@_paper_addsize:nnn { b5j } { { 182 mm } { 257 mm } }
447 \@@_paper_addsize:nnn { b6j } { { 128 mm } { 182 mm } }
448 \@@_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。

```

449 \prop_if_in:NoI \g_@@_paper_sizelist_prop
450 { \tl_to_str:N \l_@@_paper_sizeinfo_tl }
451 {
452   \prop_get:NoN \g_@@_paper_sizelist_prop
453   { \tl_to_str:N \l_@@_paper_sizeinfo_tl }
454   \l_@@_paper_sizeinfo_tl
455 }

```

处理用户设定「二」：处理字列表，使用逗号列表将长、宽分离。

```

456 \clist_set:No \l_tmpa_clist
457 { \l_@@_paper_sizeinfo_tl }
458 \clist_pop:NN \l_tmpa_clist \l_tmpa_tl
459 \clist_pop:NN \l_tmpa_clist \l_tmpb_tl

```

3.1.5 页面方向

处理页面方向选项。

```

460 \bool_if:NTF \l_@@_paper_portrait_bool
461 {
462   \dim_set:Nn \l_tmpa_dim
463   { \tl_use:N \l_tmpa_tl }
464   \dim_set:Nn \l_tmpb_dim

```

```

465     { \tl_use:N \l_tmpb_tl }
466   }
467   {
468     \dim_set:Nn \l_tmpa_dim
469       { \tl_use:N \l_tmpb_tl }
470     \dim_set:Nn \l_tmpb_dim
471       { \tl_use:N \l_tmpa_tl }
472   }

```

3.1.6 輔助線判定

辅助线设置。有些肮脏？

```

473 \bool_if:NT \l_@@_paper_corpmark_mark_bool
474   {
475     \legacy_if_set_true:n { tombow }
476     \bool_if:NTF \l_@@_paper_corpmark_date_bool
477       {
478         \legacy_if_set_true:n { tombowdate }
479         \dim_set:Nn \@tombowwidth { .1 \mpt }
480         \@bannertoken
481         {
482           \str_use:N \c_sys_jobname_str
483           \tl_use:N \c_space_tl (
484             \int_use:N \c_sys_year_int -
485             \exp_args:No \two@digits { \int_use:N \c_sys_month_int } -
486             \exp_args:No \two@digits { \int_use:N \c_sys_day_int }
487             \tl_use:N \c_space_tl
488             \exp_args:No \two@digits { \int_use:N \c_sys_hour_int } :
489             \exp_args:No \two@digits { \int_use:N \c_sys_minute_int } )
490         }
491         \maketombowbox
492       }
493       {
494         \legacy_if_set_false:n { tombowdate }
495         \dim_set:Nn \@tombowwidth { \z@ }
496         \maketombowbox
497       }
498   }

```

3.1.7 完成設置

完成纸张给配置。注意其中以及混入了一些奇怪的辅助线用判断，以及为与其它一些可能会操作页面的神奇的宏集兼容，会尝试统一某些长度。

```

499 \dim_set:Nn { \paperwidth } \l_tmpa_dim
500 \dim_set:Nn { \paperheight } \l_tmpb_dim
501 \bool_if:NT \l_@@_paper_corpmark_mark_bool
502   {
503     \dim_if_exist:NF \stockwidth

```

```

504     { \dim_new:N \stockwidth }
505     \dim_if_exist:NF \stockheight
506     { \dim_new:N \stockheight }
507     \dim_set:Nn \stockwidth { \l_tmpa_dim + 2 in }
508     \dim_set:Nn \stockheight { \l_tmpb_dim + 2 in }
509     \dim_set_eq:NN \l_tmpa_dim \stockwidth
510     \dim_set_eq:NN \l_tmpb_dim \stockheight
511   }
512 \pdf_pagesize_gset:nn
513   { \dim_use:N \l_tmpa_dim }
514   { \dim_use:N \l_tmpb_dim }

```

3.1.8 清除内存

并做好内存管理。

```

515 \@@_macro_release:N \@@_paper_addsize:nnn
516 \@@_macro_release:N \g_@@_paper_sizelist_prop
517 \@@_macro_release:N \l_@@_paper_sizeinfo_tl
518 % \@@_macro_release:N \g_@@_paper_sizeconf_clist
519 % \@@_macro_release:N \g_@@_paper_width_dim
520 % \@@_macro_release:N \g_@@_paper_height_dim
521 \@@_macro_release:N \l_@@_paper_corpmark_mark_bool
522 \@@_macro_release:N \l_@@_paper_corpmark_date_bool
523 \@@_macro_release:N \l_@@_paper_portrait_bool
524 % \@@_macro_release:N \l_@@_paper_widthaux_tl
525 % \@@_macro_release: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。

```

526 \clist_new:N \l_@@_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。

```

527 \clist_set:Nn \l_@@_jfm_feats_clist { nstd }

```

3.2.3 内存管理

于最后清除之。

```

528 \@@_macro_release:N \l_@@_jfm_feats_clist

```

3.3 選項設定

3.3.1 組版方向設定

支持纵排组版，使用钩子进行处理。

```

529 \bool_if:NT \l_@@_layout_tate_bool
530 {
531   \RequirePackage { lltjext } \tate
532   \@@_doc_beg:n
533   {
534     \iow_trem:n { 《縦組モード》 } \adjustbaseline
535   }
536 }

```

「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。）

```

537 % \bool_if:NT \l_@@_layout_tate_bool
538 % { \clist_put_left:Nn \l_@@_jfm_feats_clist { vert } }

```

3.3.2 内存管理

清除不必要的宏。

```

539 \@@_macro_release:N \l_@@_layout_tate_bool
540 \@@_macro_release:N \l_@@_layout_restonecol_bool
541 \@@_macro_release:N \l_@@_layout_column_gap_tl
542 \@@_macro_release:N \l_@@_layout_twoside_bool
543 \@@_macro_release:N \l_@@_layout_mparswitch_bool
544 \@@_macro_release:N \l_@@_layout_titlepage_bool
545 \@@_macro_release:N \l_@@_layout_doctype_bool
546 \@@_macro_release:N \l_@@_layout_openleft_bool
547 \@@_macro_release:N \l_@@_layout_openright_bool
548 \@@_macro_release:N \l_@@_layout_draft_bool
549 \@@_macro_release:N \l_@@_layout_fleqn_bool
550 \@@_macro_release:N \l_@@_layout_leqno_bool
551 \@@_macro_release:N \l_@@_layoit_openbib_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。

```

552 \tl_new:N \l_@@_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。

```

553 \str_if_eq:VnT \l_@@_lang_cj_str { t }
554 {
555   \clist_put_left:Nn \l_@@_jfm_feats_clist { trad }
556   \tl_set:Nn \l_@@_font_langfeat_tl { Chinese~Traditional }
557 }
558 \str_if_eq:VnT \l_@@_lang_cj_str { s }
559 {
560   \clist_put_left:Nn \l_@@_jfm_feats_clist { smpl }
561   \tl_set:Nn \l_@@_font_langfeat_tl { Chinese~Simplified }

```



```

562 }
563 \str_if_eq:VnT \l_@@_lang_cj_str { j }
564 {
565     \clist_put_left:Nn \l_@@_jfm_feats_clist { jp }
566     \tl_set:Nn \l_@@_font_langfeat_tl { Japanese }
567 }

```

3.4.3 字體矩陣高級設置

设置字体矩阵的标点高级特性。

```

568 \bool_if:NT \l_@@_jfm_hanging_bool
569 { \clist_put_left:Nn \l_@@_jfm_feats_clist { hgp } }
570 \bool_if:NT \l_@@_jfm_linegap_bool
571 { \clist_put_left:Nn \l_@@_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。

```

572 \tl_set:Nx \Cjascale { \tl_use:N \l_@@_font_cjscale_tl }
573 \tl_set:Nx \ltj@stdmcfont { \tl_use:N \l_@@_font_mincho_tl }
574 \tl_set:Nx \ltj@stdgtfont { \tl_use:N \l_@@_font_gothic_tl }
575 \tl_set:Nx \ltj@stdyokojfm
576 { eva / { \clist_use:Nn \l_@@_jfm_feats_clist { , } } }
577 \tl_set:Nx \ltj@stdtatejfm
578 { eva / { \clist_use:Nn \l_@@_jfm_feats_clist { , } , vert } }
579 \RequirePackage { luatexja }

```

并启用 luatexja-adjust 宏集。

```

580 \RequirePackage { luatexja-adjust }
581 \ltjenableadjust
582 [
583     lineend = extended,
584     priority = true
585 ]

```

3.4.5 補正用單位

设置补正用 point 单位，依据缩放率。

```

586 \dim_set:Nn \mpt { \l_@@_font_magscale_tl \p@ }

```

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

```

587 \dim_compare:nNnT
588 { \mpt } < { 1 \p@ }
589 { \tl_set:Nn \@ptsize { -20 } }
590 \dim_compare:nNnT
591 { \mpt } = { 1 \p@ }
592 { \tl_set:Nn \@ptsize { 0 } }
593 \dim_compare:nNnT
594 { \mpt } = { 1.095 \p@ }

```

```

595 { \tl_set:Nn \@ptsize { 1 } }
596 \dim_compare:nNnT
597 { \mpt } = { 1.200 \p@ }
598 { \tl_set:Nn \@ptsize { 2 } }
599 \dim_compare:nNnT
600 { \mpt } > { 1.200 \p@ }
601 { \tl_set: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。

```

602 \bool_if:NTF \l_@@_font_xreal_bool
603 {
604   \dim_compare:nNnT
605     { \mpt } = { \p@ }
606     { \bool_set_false:N \l_@@_font_xreal_bool }
607 }
608 { \dim_set:Nn \mpt { \p@ } }

```

实际补正。注意编码等，以及 expl3 与 L^AT_EX 2_ε 的兼容性（需小心维护）。

```

609 \bool_if:NT \l_@@_font_xreal_bool
610 {
611   \exp_after:wN \cs_set_eq:NN \cs:w TU/lmr/m/n/10 \cs_end: \scan_stop:
612   \exp_after:wN \cs_set_eq:NN \cs:w TU/lms/m/n/10 \cs_end: \scan_stop:
613   \exp_after:wN \cs_set_eq:NN \cs:w TU/lmtt/m/n/10 \cs_end: \scan_stop:
614   \@@_luafunc_new:N \@@_magnify_font_calc
615   \group_begin:
616     % \char_set_catcode_other:N \$
617     \char_set_catcode_other:N %
618     \char_set_catcode_space:n { 32 }
619     \lua_now:e
620     {
621       local mpt = tex.getdimen('mpt')/65536
622       lua.get_functions_table()[\the\@@_magnify_font_calc] = function()
623         tex.sprint(-2, math.floor(0.5 + mpt * tex.getdimen(luatexbase.registernumber 'dimen@')))
624       end
625       function luatexja.ncls_unmagnify_fsize(e)
626         local s = luatexja.print_scaled(floor(0.5 + e / mpt * 65536))
627         tex.sprint(-2, (s:match('%.\$')) and s:sub(1, -3) or s)
628       end
629     }
630   \group_end:
631   \cs_new:Npn \@@_magnify_external_font:w #1~at #2~at #3 \q_nil
632   {
633     \tl_set:Nn \l_tmpa_tl { #1 }
634     \tl_set:Nn \l_tmpb_tl { #2 }
635     \tl_if_empty:NTF \l_tmpb_tl
636     {
637       \tl_set:Nx \l_tmpb_tl

```

```

638         {
639             scaled \lua_now:e { tex.sprint(-2, math.floor(0.5 + \l_@@_font_magscale_tl * 1000)) }
640         }
641     }
642     {
643         \dim_set:Nn \dimen@ { \tl_use:N \l_tmpb_tl }
644         \tl_set:Nx \l_tmpb_tl
645         { at \@@_luafunc_use:N \@@_magnify_font_calc~sp }
646     }
647     \tl_set:Nx \l_tmpa_tl
648     {
649         \tl_set:Nn \exp_not:N \external@font
650         { \tl_use:N \l_tmpa_tl \tl_use:N \l_tmpb_tl }
651     }
652 }
653 \cs_new_eq:NN \@@_get_externalfont_orig: { \get@external@font }
654 \cs_set:Nn \get@external@font
655 {
656     \tl_set:Nx \f@size
657     { \lua_now:e { luatexja.ncls_unmagnify_fsize(\f@size) } }
658     \@@_get_externalfont_orig:
659     \group_begin:
660         \tl_set:Nx \l_tmpa_tl
661         { \external@font \tl_use:N \c_space_tl~at \tl_use:N \c_space_tl~at }
662         \exp_after:wN \@@_magnify_external_font:w \tl_use:N \l_tmpa_tl \q_nil
663         \exp_after:wN
664     \group_end:
665     \tl_use:N \l_tmpa_tl
666 }
667 }

```

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。

```

668 \bool_if:NT \l_@@_font_fallback_bool
669 {
670     \group_begin:
671         \char_set_catcode_space:n { 32 }
672         \lua_now:e
673         {
674             luaotfload.add_fallback
675             (
676                 " nclsfallback ",
677                 { " \tl_use:N \l_@@_font_fallback_tl : mode = node ; script = cjk ; language = \tl_use:N \l_@@_font_langfeat_tl " }
678             )
679         }
680     \group_end:

```

```
681 }
```

3.4.8 中日 NFSS 設定

先防止标记这些字体的记号被展开。

```
682 \exp_after:wN \cs_set_eq:NN \cs:w JY3/mc/m/n/10 \cs_end: \scan_stop:
683 \exp_after:wN \cs_set_eq:NN \cs:w JY3/gt/m/n/10 \cs_end: \scan_stop:
684 \exp_after:wN \cs_set_eq:NN \cs:w JT3/mc/m/n/10 \cs_end: \scan_stop:
685 \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（ \LaTeX 2_ϵ 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。

```
686 \cs_new:Nn \l_@@_font_declareshape:nnnn
687 {
688   \bool_if:NTF \l_@@_font_fallback_bool
689   {
690     \DeclareFontShape { #1 } { #2 } { m } { n }
691     {
692       <-> s * [ \tl_use:N \l_@@_font_cjsacale_tl ] #3 :
693       - kern ; script = cjk ; language = \tl_use:N \l_@@_font_langfeat_tl ;
694       jfm = { eva / { \clist_use:Nn \l_@@_jfm_feats_clist { , } #4 } } ;
695       fallback = nclsfallback
696     }
697   }
698   {
699     \DeclareFontShape { #1 } { #2 } { m } { n }
700     {
701       <-> s * [ \tl_use:N \l_@@_font_cjsacale_tl ] #3 :
702       - kern ; script = cjk ; language = \tl_use:N \l_@@_font_langfeat_tl ;
703       jfm = { eva / { \clist_use:Nn \l_@@_jfm_feats_clist { , } #4 } }
704     }
705   }
706 }
```

终于，定义四个源字体。

```
707 \l_@@_font_declareshape:nnnn { JY3 } { mc } { \tl_use:N \l_@@_font_mincho_tl } { }
708 \l_@@_font_declareshape:nnnn { JY3 } { gt } { \tl_use:N \l_@@_font_gothic_tl } { }
709 \l_@@_font_declareshape:nnnn { JT3 } { mc } { \tl_use:N \l_@@_font_mincho_tl } { , vert }
710 \l_@@_font_declareshape:nnnn { JT3 } { gt } { \tl_use:N \l_@@_font_gothic_tl } { , vert }
```

然后偷懒，用递归定义其余分支字体。至于 deluxe 就以后再加吧。

```
711 \clist_map_inline:nn { JY3, JT3 }
712 {
713   \clist_map_inline:nn { n, it, sl, sc }
714   {
715     \clist_map_inline:nn { m, b, bx, sb }
716     {
717       \bool_if:nF { \str_if_eq_p:nn { ##1 } { n } && \str_if_eq_p:nn { #####1 } { m } }
718       { \DeclareFontShape { #1 } { gt } { #####1 } { ##1 } { <-> ssub * gt/m/n } { } }
```

```

719     }
720     \str_if_eq:nnF { ##1 } { n }
721     { \DeclareFontShape { #1 } { mc } { m } { ##1 } { <-> ssub * mc/m/n } { } }
722     \clist_map_inline:nn { b, bx, sb }
723     { \DeclareFontShape { #1 } { mc } { #####1 } { ##1 } { <-> ssub * gt/m/n } { } }
724 }
725 }

```

3.4.9 清理内存

内存管理。清除不必要的宏定义。

```

726 \@@_macro_release:N \l_@@_font_magscale_tl
727 \@@_macro_release:N \l_@@_lang_cj_str
728 \@@_macro_release:N \l_@@_font_langfeat_tl
729 \@@_macro_release:N \l_@@_font_mincho_tl
730 \@@_macro_release:N \l_@@_font_gothic_tl
731 \@@_macro_release:N \l_@@_font_cjscale_tl
732 \@@_macro_release:N \l_@@_jfm_hanging_tl
733 \@@_macro_release:N \l_@@_jfm_linegap_tl
734 \@@_macro_release:N \l_@@_font_xreal_bool
735 \@@_macro_release:N \l_@@_font_fallback_bool
736 \@@_macro_release:N \l_@@_font_fallback_tl
737 \@@_macro_release:N \@@_font_declareshape:nnnn

```

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		
\\$ 616, 627	\@_magnify_font_calc . 614, 622, 645	\bool_if:NT 473, 501,
\% 617	\@_paper_addsize:nnn	529, 537, 568, 570, 609, 668
\@@_at_doc_begin:n 72 414, 420–448, 515	\bool_if:NTF 460, 476, 602, 688
\@@_at_doc_end:n 48, 71	\@bannertoken 480	\bool_new:N 83,
\@@_at_preamble_end:n 47, 53, 70	\@ifl@t@r 27	84, 107, 123, 149, 172, 173,
\@@_doc_beg:n 49, 532	\@ifpackagelater 16	201, 207, 208, 236, 260,
\@@_font_declareshape:nnnn . . .	\@ptsize 589, 592, 595, 598, 601	276, 292, 378, 379, 387, 396
. 686, 707–710, 737	\@tombowwidth 479, 495	\bool_set_false:N
\@@_get_external_font_orig: 653, 658	\\ 22, 32, 42, 64 96, 112, 127, 153,
\@@_keyoptions_process:n 9, 12, 69, 407	A	179, 180, 190, 215, 219,
\@@_luafunc_new:N 59, 74, 614	\adjustbaseline 534	247, 265, 281, 297, 391, 606
\@@_luafunc_use:N 60, 75, 645	\AtBeginDocument 49	\bool_set_true:N
\@@_macro_release:N	\AtEndOfClass 48 90, 91, 95, 111, 128,
. 51, 69–75, 515–525,	\AtEndPreamble 14, 47	154, 184, 185, 189, 214,
528, 539–551, 726–737	B	220, 224, 225, 242, 264,
\@@_magnify_external_font:w 631, 662	\bool_if:nF 717	280, 296, 383, 384, 392, 402

C

\c_space_tl 483, 487, 661

\c_sys_day_int 486

\c_sys_hour_int 488

\c_sys_jobname_str 482

\c_sys_minute_int 489

\c_sys_month_int 485

\c_sys_year_int 484

\char_set_catcode_other:N . 616, 617

\char_set_catcode_space:n . 618, 671

\Cjascale 572

\clist_map_inline:nn

. 711, 713, 715, 722

\clist_new:N 409, 526

\clist_pop:NN 458, 459

\clist_put_left:Nn

. 538, 555, 560, 565, 569, 571

\clist_set:Nn 527

\clist_set:No 456

\clist_use:Nn . . . 576, 578, 694, 703

\cs:w 611–613, 682–685

\cs_end: 611–613, 682–685

\cs_if_exist:Nf 6, 14

\cs_if_exist:Ntf 8

\cs_new:Nn 51, 414, 686

\cs_new:Npn 631

\cs_new_eq:NN 653

\cs_new_protected:Nn 9, 12

\cs_new_protected:Npn . 47–49, 59, 60

\cs_set:Nn 654

\cs_set_eq:NN . . . 611–613, 682–685

\cs_undefine:N 56

D

\DeclareFontShape

. . . . 690, 699, 718, 721, 723

\dim_compare:nNt

. 587, 590, 593, 596, 599, 604

\dim_if_exist:Nf 503, 505

\dim_new:N . . . 67, 410, 411, 504, 506

\dim_set:Nn 68, 243, 248, 462, 464,

468, 470, 479, 495, 499,

500, 507, 508, 586, 608, 643

\dim_set_eq:NN 509, 510

\dim_use:N 513, 514

\dimen@ 643

E

\exp_after:wN

. 611–613, 662, 663, 682–685

\exp_args:No 485, 486, 488, 489

\exp_not:N 649

\exp_not:n 103,

118, 134, 160, 197, 232,

255, 271, 287, 303, 338, 355

\ExplSyntaxOff 57

\ExplSyntaxOn 55

\external@font 649, 661

F

\f@size 656, 657

\fmtversion 27

G

\g_@@_aftercls_del_seq . . . 50, 52, 56

\g_@@_paper_height_dim . . . 411, 520

\g_@@_paper_sizeconf_clist 409, 518

\g_@@_paper_sizelist_prop . . .

. . . . 408, 416, 449, 452, 516

\g_@@_paper_width_dim . . . 410, 519

\g_msg_module_name_prop 5

\get@external@font 653, 654

\group_begin: 615, 659, 670

\group_end: 630, 664, 680

I

\int_use:N 484–486, 488, 489

\iow_trem:n 534

K

\keys_define:nn

. . . 77, 85, 108, 124, 140,

150, 166, 174, 202, 209,

237, 261, 277, 293, 309,

344, 362, 372, 380, 388, 398

L

\l_@@_font_cjsacale_tl . . . 692, 701

\l_@@_font_cjsacle_tl 374

\l_@@_font_cjscale_tl . 371, 572, 731

\l_@@_font_fallback_bool . . .

. . . . 396, 402, 668, 688, 735

\l_@@_font_fallback_tl

. 397, 403, 677, 736

\l_@@_font_gothic_tl

. 361, 365, 574, 708, 710, 730

\l_@@_font_langfeat_tl 552, 556,

561, 566, 677, 693, 702, 728

\l_@@_font_magscale_tl

. 308, 312–329, 586, 639, 726

\l_@@_font_mincho_tl

. 360, 364, 573, 707, 709, 729

\l_@@_font_xreal_bool . . . 387,

391, 392, 602, 606, 609, 734

\l_@@_jfm_feats_clist . 526–528,

538, 555, 560, 565, 569,

571, 576, 578, 694, 703

\l_@@_jfm_hanging_bool 378, 383, 568

\l_@@_jfm_hanging_tl 732

\l_@@_jfm_linegap_bool 379, 384, 570

\l_@@_jfm_linegap_tl 733

\l_@@_lang_cj_str 343,

347–349, 553, 558, 563, 727

\l_@@_layoit_openbib_bool . . . 551

\l_@@_layout_cloumn_gap_tl . . . 168

\l_@@_layout_column_gap_tl 165, 541

\l_@@_layout_doctype_bool . . . 545

\l_@@_layout_doctype_str . . .

. 139, 143–145

\l_@@_layout_draft_bool

. 236, 242, 247, 548

\l_@@_layout_fleqn_bool

. 260, 264, 265, 549

\l_@@_layout_leqno_bool

. 276, 280, 281, 550

\l_@@_layout_mparswitch_bool . .

. . . . 173, 180, 185, 190, 543

\l_@@_layout_openbib_bool . . .

. 292, 296, 297

\l_@@_layout_openleft_bool . . .

. . . . 207, 214, 219, 224, 546

\l_@@_layout_openright_bool . .

. . . . 208, 215, 220, 225, 547

\l_@@_layout_restonecol_bool . .

. 149, 153, 154, 540

\l_@@_layout_tate_bool

. 123, 127, 128, 529, 537, 539

\l_@@_layout_titlepage_bool . .

. 201, 204, 544

\l_@@_layout_twoside_bool . . .

. . . . 172, 179, 184, 189, 542

\l_@@_paper_corpmark_date_bool

. 84, 91, 96, 476, 522

<code>\l_@@_paper_corpmark_mark_bool</code>	<code>\msg_fatell:nn</code> 45	<code>\seq_gput_right:Nn</code> 52
. . . . 83, 90, 95, 473, 501, 521	<code>\msg_new:nnnn</code> 18, 29, 39, 61	<code>\seq_new:N</code> 50
<code>\l_@@_paper_heightaux_tl</code> . . 413, 525	<code>\msg_new:nnxxx</code> 131	<code>\stockheight</code> 505, 506, 508, 510
<code>\l_@@_paper_portrait_bool</code>		<code>\stockwidth</code> 503, 504, 507, 509
. . . . 107, 111, 112, 460, 523	N	<code>\str_if_eq:nnF</code> 720
<code>\l_@@_paper_sizeinfo_tl</code>	<code>\NeedsTeXFormat</code> 1	<code>\str_if_eq:VnT</code> 553, 558, 563
76, 79, 450, 453, 454, 457, 517	<code>\NewDocumentCommand</code> 6	<code>\str_if_eq_p:nn</code> 717
<code>\l_@@_paper_widthaux_tl</code> . . 412, 524	<code>\newluafunction</code> 59	<code>\str_new:N</code> 139, 343
<code>\l_tmpa_clist</code> 456, 458, 459	O	<code>\str_set:Nn</code> 143–145, 347–349
<code>\l_tmpa_dim</code> 462, 468, 499, 507, 509, 513	<code>\overfullrule</code> 243, 248	<code>\str_use:N</code> 482
<code>\l_tmpa_tl</code> 458, 463, 471,	P	<code>\sys_if_engine luatex:F</code> 37
633, 647, 650, 660, 662, 665	<code>\p@</code> 68, 586, 588,	
<code>\l_tmpb_dim</code> 464, 470, 500, 508, 510, 514	591, 594, 597, 600, 605, 608	T
<code>\l_tmpb_tl</code> 459, 465, 469,	<code>\paperheight</code> 500	<code>\tate</code> 531
634, 635, 637, 643, 644, 650	<code>\paperwidth</code> 499	<code>\the</code> 622
<code>\legacy_if_set_false:n</code> 494	<code>\pdf_pagesize_gset:nn</code> 512	<code>\tl_if_empty:NIF</code> 635
<code>\legacy_if_set_true:n</code> 475, 478	<code>\ProcessKeyOptions</code> 8, 9	<code>\tl_new:N</code> 76, 165, 308, 360,
<code>\ltj@stdgtfont</code> 574	<code>\ProcessKeysOptions</code> 12	361, 371, 397, 412, 413, 552
<code>\ltj@stdmcfont</code> 573	<code>\prop_get:NoN</code> 452	<code>\tl_set:Nn</code> 312–329, 403,
<code>\ltj@stdtatejfm</code> 577	<code>\prop_gput:Nnn</code> 5	556, 561, 566, 589, 592,
<code>\ltj@stdyokojfm</code> 575	<code>\prop_gput_if_new:Nnn</code> 416	595, 598, 601, 633, 634, 649
<code>\ltjenableadjust</code> 581	<code>\prop_if_in:NoT</code> 449	<code>\tl_set:Nx</code> 572–575,
<code>\lua_now:e</code> 619, 639, 657, 672	<code>\prop_new:N</code> 408	577, 637, 644, 647, 656, 660
<code>\luafunction</code> 60	<code>\ProvidesExplClass</code> 3	<code>\tl_to_str:N</code> 450, 453
M	Q	<code>\tl_use:N</code> . . . 463, 465, 469, 471,
<code>\maketombowbox</code> 491, 496	<code>\q_nil</code> 631, 662	483, 487, 572–574, 643,
<code>\mpt</code> 67,	R	650, 661, 662, 665, 677,
68, 243, 248, 479, 586, 588,	<code>\RequirePackage</code>	692, 693, 701, 702, 707–710
591, 594, 597, 600, 605, 608	. . . 2, 7, 11, 15, 531, 579, 580	<code>\two@digits</code> 485, 486, 488, 489
<code>\msg_critical:nn</code> 25, 35	S	Z
<code>\msg_error:nnxxx</code>	<code>\scan_stop:</code> 611–613, 682–685	<code>\z@</code> 495
. 100, 115, 157, 194, 229,		<code>\zw</code> 170
252, 268, 284, 300, 332, 352		